

---

# Enterprise PeopleTools 8.50 PeopleBook: PeopleTools Portal Technology

---

**September 2009**

Copyright © 1988, 2009, Oracle and/or its affiliates. All rights reserved.

## **Trademark Notice**

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

## **License Restrictions Warranty/Consequential Damages Disclaimer**

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

## **Warranty Disclaimer**

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

## **Restricted Rights Notice**

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

### *U.S. GOVERNMENT RIGHTS*

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

## **Hazardous Applications Notice**

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

## **Third Party Content, Products, and Services Disclaimer**

This software and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third party content, products and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third party content, products or services.

# Contents

## Preface

<b>PeopleTools Portal Technologies Preface .....</b>	<b>xix</b>
PeopleTools Portal Technologies .....	xix

## Part 1

### Getting Started with PeopleTools Portal Technologies

#### Chapter 1

<b>Getting Started with PeopleTools Portal Technologies .....</b>	<b>3</b>
PeopleTools Portal Technologies Overview .....	3
PeopleTools Portal Technologies Implementation .....	3
Installing PeopleSoft Pure Internet Architecture .....	4
Configuring Your Web Profile .....	4
Configuring Application Security .....	5
Setting Up the Portal .....	5
(Optional) Setting Up JSR 168 .....	6
Configuring Performance Features .....	6

#### Chapter 2

<b>Understanding PeopleSoft Pure Internet Architecture .....</b>	<b>7</b>
PeopleSoft Pure Internet Architecture Fundamentals .....	7
Database Server .....	8
Application Servers .....	9
Application Servers .....	9
Domains .....	9
Listeners, Handlers, and Queues .....	10
PeopleSoft Server Processes .....	13
Services .....	14
Oracle Middleware .....	15
Database Connectivity .....	15

Batch Server Environment .....	15
Batch Server Environment .....	15
Batch Server Support .....	16
PeopleSoft Process Scheduler and the Application Server .....	16
Web Server .....	16
Server Software Elements .....	17
PeopleSoft Servlets .....	17
Oracle Jolt .....	17
Web Browser .....	18
Configuration and Deployment Options .....	19
Configuration Options .....	20
Deployment Options .....	21

## Chapter 3

<b>Understanding Portal Technology .....</b>	<b>23</b>
Portal Architecture .....	23
Node Definitions .....	25
Portal Templates and Template Pagelets .....	25
Navigation .....	26
Portal Registry .....	27
Portal Servlets .....	29
Portal Servlets .....	29
Page Assembly for Page-Based Templates .....	30
Portal URL Formats .....	31
Basic Portal URL Format .....	32
URL Format for PeopleSoft Pure Internet Architecture Content Types .....	34
URL Format for Unwrapped PeopleSoft Pure Internet Architecture Content .....	35
Pagelet URLs .....	36
System URLs .....	36
Proxy Architecture and Relative URLs .....	37
Page-Based Template Proxying .....	38
Proxying and URL Conversion .....	38
Anchor Tag Conversions .....	39
Form Tag Conversions .....	39
JavaScript Conversions .....	40

## Part 2

### Using Navigation Pages

#### Chapter 4

<b>Working With Navigation Pages .....</b>	<b>43</b>
Understanding Navigation Pages .....	43
Standard Navigation Pages .....	44
Custom Navigation Pages .....	46
Navigation Page Display Caching .....	47
Understanding Navigation Collections .....	48
Navigation Collections .....	48
Navigation Collection Images .....	50
Common Elements Used on Navigation Pages .....	52
Setting Up Navigation Collection Options .....	52
Understanding Navigation Collection Options .....	53
Common Elements Used in This Section .....	54
Pages Used to Define Portal Utilities Options .....	56
Defining Portal Utilities System Options .....	56
Defining Portal Utilities Registry Options .....	59
Creating and Maintaining Navigation Collections .....	61
Pages Used to Create and Maintain Navigation Collections .....	62
Searching for a Navigation Collection .....	63
Creating or Editing a Navigation Collection .....	64
Adding a Folder .....	68
Selecting a Source Folder .....	71
Selecting Navigation Collection Images .....	72
Editing a Folder .....	73
Adding a Link .....	74
Selecting a Source Link .....	76
Editing a Link .....	78
Publishing Navigation Collections .....	80
Pages Used to Publish Navigation Collections .....	80
Publishing a Navigation Collection .....	80
Publishing Pagelets .....	83
Pages Used to Publish Pagelets to Other Portals .....	84
Publishing a Pagelet to Multiple Portals .....	84
Publishing Multiple Pagelets to a Portal .....	86
Running Portal Utility Processes .....	88
Understanding Portal Utility Processes .....	88

Pages Used to Run Portal Utility Processes .....	90
Running the Synchronize Collection Security Process .....	90
Running the Delete Empty Folders Process .....	91
Running the Clean Portal Project Process .....	92

## Part 3

### Working with PeopleTools Portals

#### Chapter 5

<b>Administering Portals .....</b>	<b>99</b>
Understanding Portal Administration .....	99
Common Elements used in This Chapter .....	101
Administering Portal Definitions .....	101
Common Elements Used in This Section .....	101
Pages Used to Administer Portal Definitions .....	102
Defining Portals .....	102
Adding and Editing Portal Definitions .....	103
Changing Default Portals .....	104
Administering Folders .....	105
Pages Used to Administer Folders .....	105
Defining Folders .....	105
Deleting Folders .....	107
Setting Folder Security .....	108
Administering Content References .....	109
Pages Used to Administer Content References .....	110
Defining Content References .....	112
Creating Related Links .....	119
Setting Content Reference Security .....	122
Reviewing Content Reference Security .....	123
Testing Content References .....	124
Deleting Content References .....	124
Defining Content Reference Links .....	125
Setting Content Reference Link Security .....	128
Managing Folders and Content References .....	129
Pages Used to Manage Folders and Content References .....	129
Copying Folders and Content References .....	130
Moving Folders and Content References .....	131
Synchronizing Portal Object Security .....	132
Registering Components in Multiple Portals .....	134
Registering Components Multiple Times .....	134

Managing General Portal Settings .....	135
Page Used to Manage General Portal Settings .....	135
Managing General Portal Settings .....	136
Managing Portal Objects .....	139
Pages Used to Manage Portal Objects .....	139
Viewing Portal Objects Folder Settings .....	139
Managing Templates .....	140
Adding and Editing Templates .....	141
Deleting Templates .....	142
Reviewing Menu Item Information .....	142
Common Elements Used in This Section .....	142
Pages Used to Review Menu Item Information .....	143
Viewing Menu Item Details .....	143
Viewing Menu Folder Structure Information .....	145
Viewing Menu Security Details .....	147
Building Registry Search Indexes .....	149
Page Used to Build Registry Search Indexes .....	149
Associating Keywords to Content References .....	150
Building Search Indexes For PeopleSoft Applications .....	150

## Chapter 6

<b>Administering Portal Homepages and Pagelets .....</b>	<b>153</b>
Understanding Portal Homepage and Pagelet Administration .....	153
Managing Tabbed Homepages .....	153
Pages Used to Manage Tabbed Homepages .....	154
Adding Homepage Tabs .....	154
Configuring the Homepage Tab .....	154
Selecting the Tab Content .....	157
Arranging the Tab Layout .....	158
Rearranging the Tab Order .....	160
Troubleshooting Homepages .....	160
Managing Pagelet Attributes .....	161
Page Used to Manage Pagelet Attributes .....	161
Managing Pagelet Attributes .....	161
Configuring Pagelet Help Links .....	164
Configuring Pagelet Time-outs .....	164
Configuring Automatic Pagelet Refresh Attributes .....	165

## Chapter 7

<b>Modifying the Portal Interface .....</b>	<b>167</b>
---	------------

Using Predefined Template HTML Objects .....	167
Using Predefined Homepage HTML Objects .....	169
Constructing Homepages .....	169
Constructing Headers .....	171
Constructing Pagelets .....	172
Adapting Homepage and Pagelet Objects .....	173
Using Custom Portal Headers .....	177
Changing Link Colors .....	178
Applying Changes .....	179

## Chapter 8

<b>Working with Portal Templates .....</b>	<b>181</b>
Understanding Portal Templates .....	181
Applying Template Types .....	181
Understanding Template Types .....	182
Using Page-Based Static Templates .....	183
Using Frame-Based Static Templates .....	183
Using Frame-Based Dynamic Templates .....	184
Using iFrame-Based Dynamic Templates .....	188
Developing Portal Templates .....	193
Template Pagelets Based on Pages .....	193
Inheritance of Style Classes .....	194
Template Pagelet Tags .....	194
Considerations for Non-PeopleSoft Content Providers .....	195
Considerations for Frame-Based Templates .....	195
Partial Path Nodes .....	196
Pagelet Names in Templates .....	196
Considerations for Forms and Templates .....	196
Error Message Display .....	197

## Chapter 9

<b>Using Portal Caching Features .....</b>	<b>199</b>
Understanding Portal Caching .....	199
Administering Server-Based Caching .....	200
Understanding Caching Scope .....	200
Implementing Target Content Caching .....	201
Implementing Pagelet Caching .....	202
Implementing Metadata Caching .....	205
Implementing Proxied JavaScript Caching .....	207
Administering Browser-Based Caching .....	207

Understanding Browser-Based Caching .....	207
Implementing Homepage Caching .....	208
Controlling Navigation Pagelet Caching .....	209
Implementing PeopleSoft Page Caching .....	210
Administering Web Server-Based Navigation Caching .....	211
Understanding Web Server-Based Navigation Caching .....	211
Implementing Web Server-Based Navigation Caching .....	211

## Chapter 10

<b>Configuring the Portal Environment .....</b>	<b>213</b>
Understanding the Authentication Domain .....	214
Configuring Web Profiles .....	216
Understanding Web Profiles .....	216
Pages Used to Configure Web Profiles .....	219
Configuring General Portal Properties .....	220
Configuring Portal Security .....	225
Configuring Virtual Addressing .....	232
Configuring Cookie Rules .....	234
Configuring Caching .....	235
Configuring Trace and Debug Options .....	240
Configuring Look and Feel .....	242
Configuring Custom Properties .....	249
Viewing Web Profile Histories .....	253
Defining Portal Nodes .....	254
Understanding Portal Nodes .....	255
Pages Used to Define Portal Nodes .....	255
Setting Portal Nodes .....	255
Specifying the Node Type .....	257
Implementing Single Signon Functionality .....	257
Understanding Single Signon Functionality .....	257
Signing In to a Non-Default Portal .....	258
Redirecting Pages with Sensitive Data .....	259
Overriding Page Encoding .....	259
Importing Menu Groups into the Portal Registry .....	260
Understanding Menu Group Importing .....	260
Page Used to Import Menu Groups .....	261
Importing Menu Groups .....	261
Selecting Components to Include in Portal Navigation .....	263
Improving Same-Server Performance Under SSL .....	264
Using SSL Accelerators .....	265
Using Reverse Proxy Servers .....	266
Understanding Reverse Proxy Servers .....	266

Making Reverse Proxy Settings .....	267
Issuing Requests by Using a Proxy Server .....	268
Specifying Hosts That Should Not Be Proxied .....	269
Using Firewalls with Reverse Proxy Servers .....	269
Using Reverse Proxy Servers with SSL Accelerators .....	271
Applying Load Balancing .....	272

## Part 4

### Working with Pagelets

#### Chapter 11

<b>Using Pagelet Wizard .....</b>	<b>277</b>
Overview of Pagelet Wizard .....	277
Overview of Pagelet Wizard Data Sources .....	278
Overview of Pagelet Wizard Display Formats .....	280
Overview of Pagelet Wizard Display Sizes .....	286
Overview of Pagelet Wizard Pagelet Types .....	286
Defining Pagelet Wizard Setup Values .....	289
Pages Used to Set Up Pagelet Wizard Setup Values .....	289
Setting Up the PeopleSoft Integration Broker Gateway for Pagelet Wizard .....	290
Setting Up the Default Gateway for Use with Pagelet Wizard Data Sources .....	290
Setting Up the Pagelet Wizard IB Connector Data Source Security .....	290
Setting Up the Pagelet Wizard Integration Broker Data Source .....	291
Defining Pagelet Wizard Transformation Types, Display Formats, and Data Types .....	292
Pages Used to Define Pagelet Wizard Transformation Types, Display Formats, and Data Types .....	292
Defining Transformation Types .....	292
Defining Display Formats .....	294
Defining Data Types .....	296
Defining Pagelet Wizard XSL Prototypes .....	300
Page Used to Define Pagelet Wizard XSL Prototypes .....	300
Defining XSL Prototypes .....	300
Defining Pagelet Wizard Headers and Footers .....	302
Pages Used to Define Pagelet Wizard Headers and Footers .....	303
Defining Headers .....	303
Defining Header Images .....	305
Defining Footers .....	306
Defining Footer Images .....	307
Defining Pagelet Wizard Pagelet Categories .....	308
Page Used to Define Pagelet Wizard Pagelet Categories .....	308
Defining Pagelet Categories .....	308

Step 1: Entering Pagelet Identifying Information .....	310
Page Used to Enter Pagelet Identifying Information .....	310
Specifying Pagelet Information .....	310
Step 2: Selecting a Pagelet Data Source .....	312
Page Used to Select a Pagelet Data Source .....	313
Selecting an HTML Data Source .....	313
Selecting an IB Connector Data Source .....	315
Selecting an Integration Broker Data Source .....	316
Selecting a Navigation Collection Data Source .....	317
Selecting an OBIEE Report Data Source .....	318
Selecting a PeopleSoft Query Data Source .....	319
Selecting a Rowset Data Source .....	320
Selecting a Search Record Data Source .....	321
Selecting a URL Data Source .....	323
Step 3: Specifying Pagelet Data Source Parameters .....	324
Understanding System Variables Supported as Data Source Parameters .....	325
Common Elements Used in This Section .....	326
Page Used to Specify Pagelet Data Source Parameters .....	329
Specifying HTML Data Source Parameters .....	329
Specifying IB Connector Data Source Parameters .....	330
Specifying Integration Broker Data Source Parameters .....	331
Specifying Navigation Collection Data Source Parameters .....	332
Specifying OBIEE Data Source Parameters .....	333
Specifying Query Data Source Parameters .....	334
Specifying Rowset Data Source Parameters .....	336
Specifying Search Record Data Source Parameters .....	336
Specifying URL Data Source Parameters .....	337
Specifying User-Defined Data Source Parameters .....	339
Step 4: Selecting a Pagelet Display Format .....	340
Page Used to Select a Pagelet Display Format .....	340
Selecting a Display Format .....	340
Step 5: Specifying Pagelet Display Options .....	342
Common Elements Used in This Section .....	342
Pages Used to Specify Pagelet Display Options .....	345
Specifying Table Display Options .....	346
Specifying Passthru Display Options .....	349
Specifying Search List Display Options .....	351
Specifying Menu Display Options .....	352
Specifying List Display Options .....	355
Specifying Chart Display Options .....	356
Specifying Custom Display Options .....	360
Specifying Data Link Details .....	362
Step 6: Specifying Pagelet Publication Options .....	365
Common Elements Used in This Section .....	365
Pages Used to Specify Pagelet Publication Options .....	367

Specifying Publication Options .....	367
Using Pagelet Wizard Post-Transformation Processing .....	371
Understanding Pagelet Wizard Post-Transformation Processing .....	372
Inserting Message Catalog Text .....	373
Formatting Date, Time, DateTime, Currency, and Number Elements .....	374
Disabling WSRP Style Substitution .....	375
Inserting a Content-Reference-Based Link .....	375
Inserting a Content Reference URL as Text .....	376
Generating Charts .....	376
Inserting Pagelet Headers and Footers in Specific Locations .....	376
Administering Pagelets .....	377
Common Elements Used in This Chapter .....	377
Pages Used to Administer Pagelet Wizard Pagelets .....	379
Understanding Administering Pagelets .....	379
Deleting and Copying Pagelet Wizard Pagelets .....	379
Generating Export and Import Scripts for Pagelet Wizard Pagelets .....	382
Reviewing Pagelet Wizard Pagelet Details .....	384

## Chapter 12

<b>Building Pagelets .....</b>	<b>387</b>
Prerequisites .....	387
Understanding Pagelets .....	387
Describing Pagelets .....	388
Describing Pagelet Development .....	388
Describing Pagelet Extensions .....	389
Describing Pagelet Personalizations .....	391
Understanding PeopleSoft Portal Architecture .....	392
Describing PeopleSoft Portal Architecture .....	393
Describing Client Access .....	394
Describing the Web Server .....	394
Describing the Application Server .....	395
Describing the Database Server .....	396
Describing the Portal Registry .....	396
Describing Portal Components .....	397
Describing Page Assembly .....	398
Understanding Pagelet Development .....	400
Describing Pagelet Development Guidelines .....	400
Describing Pagelet Development Considerations .....	401
Describing Pagelet Development Options .....	410
Describing Size Considerations .....	412
Describing HTML Design Considerations .....	414
Describing Branding Considerations .....	415

Understanding Single Signon and Pagelets .....	417
Describing Single Signon .....	417
Describing the PeopleSoft Authentication Process .....	418
Describing the PeopleSoft Authentication API .....	420
Developing Pagelets Using PeopleSoft Pure Internet Architecture .....	425
Developing Pagelets Based on PeopleSoft Pure Internet Architecture Pages .....	425
Developing Pagelets Using iScripts .....	426
Developing Pagelets Based on iScripts .....	426
Developing Pagelets Using Java .....	428
Understanding Developing Pagelets in Java .....	428
Describing Java Pagelet Development Considerations .....	429
Writing Business Logic in Java .....	432
Using Java to Write Business Logic for a PeopleSoft or Non-PeopleSoft System .....	436
Invoking PeopleSoft Components from Java .....	437
Developing Contextual Embeddable Pagelets .....	440
PeopleCode Sample for Rendering the Context-Based Embeddable Pagelet .....	442
Administering Pagelets in the Portal Registry .....	442
Registering Homepage and Template Pagelets .....	443
Modifying Pagelet Attributes and Security .....	443
Registering URL-Based Pagelets .....	446
Registering Pagelet Extensions .....	448
Handling Special Situations .....	449
Pagelet Running Location .....	449
Using Refresh Tags in Pagelets .....	449

## Chapter 13

<b>Working with Related Content Services .....</b>	<b>451</b>
Understanding Related Content Services .....	451
Understanding Delivered Related Content Services .....	454
Related Discussion Service .....	454
Related Links Service .....	455
Related Tags Service .....	455
Common Elements in This Chapter .....	455
Defining Related Content Services .....	459
Understanding Service URL Types .....	459
Pages Used to Define Related Content Services .....	460
Creating the Non-PeopleSoft URL Type .....	460
Creating the PeopleSoft Component URL Type .....	461
Creating the PeopleSoft Embeddable Pagelet URL Type .....	462
Creating the PeopleSoft OBIEE Pagelet URL Type .....	463
Creating the PeopleSoft Query URL Type .....	464
Creating the PeopleSoft Script URL Type .....	465

Testing the Related Content Service .....	466
Writing Help Text for Related Content Service Definitions .....	467
Copying Related Content Service Definitions .....	469
Assigning and Managing Related Content Services .....	469
Pages Used to Assign and Manage Related Content Services .....	470
Assigning Related Content Services to Application Pages .....	470
Mapping Fields to Service Parameters .....	473
Managing Related Content Configuration .....	475
Creating Security Handlers .....	476
Page Used to Create Related Content Security Handlers .....	476
Understanding Security Handlers .....	476
Defining Security Handlers .....	477

## Chapter 14

<b>Configuring OBIEE Report Pagelets .....</b>	<b>479</b>
Understanding OBIEE Integration Within the PeopleSoft Framework .....	479
Installing and Configuring the OBIEE Plug In .....	480
Understanding the OBIEE Plug In .....	480
Configuring the web.xml File .....	481
Installing the OBIEE Plug In .....	481
Configuring OBIEE Server Setup .....	481
Loading OBIEE Report Definitions .....	482
Loading OBIEE Reports into PeopleSoft Applications .....	484
Deleting OBIEE Reports from PeopleSoft Applications .....	485
Viewing OBIEE Report Details .....	485
Creating OBIEE Report Pagelets .....	487
Using Pagelet Wizard to Create OBIEE Report Pagelets .....	487

## Part 5

### Working with Portlet Technologies

## Chapter 15

<b>Understanding Web Services For Remote Portlets (WSRP) .....</b>	<b>491</b>
Understanding WSRP .....	491
Terminology .....	493
Implementing WSRP Protocol Scenario .....	494
WSRP and Server Cluster Configuration Considerations .....	495

## Chapter 16

<b>Configuring WS-Security For WSRP Consumption and Production .....</b>	<b>497</b>
Determining Security Requirements .....	497
Understanding the PeopleSoft WS-Security for WSRP .....	498
Setting Up WSRP Security Options .....	506
Using HTTPS Protocol to Communicate Between Producer and Consumer .....	506
Installing Certificates For Encryption and Signature .....	508
Setting Up SAML .....	510
Creating the SAML Administrator .....	510
Importing Digital Certificates .....	511
Configuring the SAML Inbound Setup .....	511
Running the RedeployWSRP.cmd Executable .....	512
Configuring WS-Security For PeopleSoft as a WSRP Producer .....	512
Defining Nodes With WS-Security .....	513
Pages Used to Define Nodes With WS-Security .....	513
Configuring WS-Security For WSRP Consumers .....	514

## Chapter 17

<b>Using WSRP to Consume and Produce Remote Portlets .....</b>	<b>517</b>
Setting Up the Local Gateway URL (Consumer) .....	517
Page Used to Set Up the Local Gateway URL .....	517
Setting Up the Local Gateway URL .....	518
Using the WSRP Consumption Wizard (Consumer) .....	518
Understanding the WSRP Consumption Wizard Data .....	519
Common Elements Used in This Section .....	519
Pages Used to Navigate WSRP Consumption Wizard .....	520
Importing Producer Information .....	520
Registering Producers .....	522
Creating Content References .....	524
Displaying Remote Portlets on a Homepage (Consumer) .....	525
Pages Used to Display Remote Portlets .....	526
Reviewing Content References .....	526
Displaying Remote Portlets on a Homepage .....	528
Producing Remote Portlets (Producer) .....	530
Pages Used to Produce Remote Portlets .....	530
Specifying PeopleSoft Content for WSRP Production .....	531
Offering WSRP Portlets .....	533
Working With WSRP Content References .....	535
Setting WSRP Display Mode .....	535

Mapping PeopleSoft Styles to WSRP Styles .....	536
Validating WSRP-Compliance .....	536
Using PeopleSoft WSRP-Produced Content .....	538

## Chapter 18

<b>Working with JSR 168 Portlets .....</b>	<b>539</b>
Understanding JSR 168 .....	539
Understanding PeopleSoft Compliance with JSR 168 .....	539
Java Portlet API .....	540
Window States .....	540
Content Caching .....	540
Portlet Modes .....	540
Configuring the Portlet Container .....	540
Describing the Portlet Entity Registry .....	541
Choosing Portlets to Produce For WSRP .....	541

## Appendix A

<b>Consuming PeopleSoft Components as a Web Service in OBIEE .....</b>	<b>543</b>
Understanding the OBIEE and the PeopleSoftComponents Web Service Interaction .....	543
Setting Up OBIEE to Consume PeopleSoft Components as a Web Service .....	544

## Appendix B

<b>Language Support For Consuming and Producing Remote Portlets .....</b>	<b>545</b>
Language Support For Consuming Remote Portlets .....	545
Language Support Rules For Producing Remote Portlets .....	546

## Appendix C

<b>Understanding Changes in Portal Configuration Settings .....</b>	<b>547</b>
General Properties .....	547
Security Properties .....	548
Virtual Addressing Properties .....	549
Cookie Rules .....	550
Caching Settings .....	550
Trace and Debug Properties .....	552
Look and Feel Settings .....	553

**Appendix D**

<b>PeopleTools Web Libraries .....</b>	<b>555</b>
Understanding Web Libraries .....	555
PeopleTools Web Libraries .....	555

**Appendix E**

<b>Configuring the Application for 8.4x Navigation .....</b>	<b>559</b>
Reconfiguring the Default Template .....	559
Reconfiguring the Navigation Pagelet .....	562

**Appendix F**

<b>Converting Frames-Based Templates to iFrame-Based Templates .....</b>	<b>565</b>
Converting the eProcurement Shopping Cart Template .....	565
Gathering Information About the Current Template .....	565
Creating the iFrame Template iScript .....	566
Creating the iFrame Template HTML Definition .....	566
Setting Security for the Template iScript .....	570
Converting the Template Content Reference .....	571
Troubleshooting Errors .....	571

**Appendix G**

<b>Configuring OBIEE Related Content Services .....</b>	<b>573</b>
OBIEE Configuration Prerequisites .....	573

**Appendix H**

<b>Related Content Reserved Words and Service Parameters .....</b>	<b>575</b>
Reserved Words Default Mapping .....	575

<b>Index .....</b>	<b>577</b>
--------------------	------------



# PeopleTools Portal Technologies Preface

This preface discusses:

- PeopleTools portal technologies.
- PeopleBooks and the Online PeopleSoft Library

---

## PeopleTools Portal Technologies

This book describes PeopleTools portal technologies, which consists of PeopleSoft Pure Internet Architecture and the PeopleTools portal technology used for creating and managing portals.

The terms *PeopleTools portal technology*, *PeopleTools base portal*, *portal technology*, and *portal* are used interchangeably throughout this PeopleBook to refer to the core portal technology that's part of PeopleTools.

---

**Note.** Portal packs and PeopleSoft Enterprise Portal are separate applications and provide their own documentation.

---

---

## PeopleBooks and the Online PeopleSoft Library

A companion PeopleBook called PeopleBooks and the Online PeopleSoft Library contains general information, including:

- Understanding the PeopleSoft online library and related documentation.
- How to send PeopleSoft documentation comments and suggestions to Oracle.
- How to access hosted PeopleBooks, downloadable HTML PeopleBooks, and downloadable PDF PeopleBooks as well as documentation updates.
- Understanding PeopleBook structure.
- Typographical conventions and visual cues used in PeopleBooks.
- ISO country codes and currency codes.
- PeopleBooks that are common across multiple applications.
- Common elements used in PeopleBooks.
- Navigating the PeopleBooks interface and searching the PeopleSoft online library.
- Displaying and printing screen shots and graphics in PeopleBooks.
- How to manage the PeopleSoft online library including full-text searching and configuring a reverse proxy server.

- Understanding documentation integration and how to integrate customized documentation into the library.
- Glossary of useful PeopleSoft terms that are used in PeopleBooks.

You can find this companion PeopleBook in your PeopleSoft online library.

## **Part 1**

# **Getting Started with PeopleTools Portal Technologies**

### **Chapter 1**

#### **Getting Started with PeopleTools Portal Technologies**

### **Chapter 2**

#### **Understanding PeopleSoft Pure Internet Architecture**

### **Chapter 3**

#### **Understanding Portal Technology**



## Chapter 1

# Getting Started with PeopleTools Portal Technologies

This chapter discusses:

- PeopleTools portal technologies overview.
- PeopleTools portal technologies implementation.

---

## PeopleTools Portal Technologies Overview

PeopleTools portal technology consists of PeopleSoft Pure Internet Architecture and the various technologies used for creating and managing portals.

PeopleSoft Pure Internet Architecture enables internet application deployment through a browser, and enables you to take advantage of PeopleSoft intranet solutions, internet solutions, and integration technologies. PeopleSoft Pure Internet Architecture runs seamlessly in portals created and managed by PeopleTools portal technology.

PeopleTools portal technology is built on top of PeopleSoft Pure Internet Architecture and enables you to easily access and administer multiple content providers, including PeopleSoft databases such as PeopleSoft CRM and HRMS, as well as non-PeopleSoft content. It enables you to combine content from these multiple sources and deliver the result to users in a unified, simple-to-use interface.

The main elements of the PeopleTools portal technology are a *portal servlet* and an *application server*. These two elements work together to provide common portal processing features such as page assembly, search ability, content management, navigation, and homepage personalization.

---

## PeopleTools Portal Technologies Implementation

To implement PeopleSoft portal technologies, you need to:

- Design and configure the portal environment.
- Install PeopleSoft Pure Internet Architecture.
- Configure your web profile.
- Configure application security.

- Set up the portal.
- (Optional) Set up JSR 168.
- Configure performance features, such as caching.

In the planning phase of your implementation, take advantage of all PeopleSoft sources of information, including installation guides, hardware and software guides, red papers, and PeopleBooks.

Build the physical network and server architecture necessary to support your development. Ensure that you consider all the scenarios for your network and security, including creating subnets with appropriate routers and switches, and creating firewalls, proxy servers, reverse proxy servers, and Secure Sockets Layer.

During implementation, perform the following tasks within your portal environment:

- Size servers appropriately.
- Configure software to optimize portal performance.
- Scale load balancing up to load requirements.

---

**Note.** These implementation activities assume that you have installed your required databases, application servers, and web servers.

---

### See Also

*Enterprise PeopleTools 8.50 PeopleBook: Getting Started with Enterprise PeopleTools*

The PeopleTools installation guide for your database platform and supplemental information about third-party application integration on the PeopleSoft Customer Connection web site.

## Installing PeopleSoft Pure Internet Architecture

To install one or more PeopleSoft portal sites, perform this step:

Step	Reference
Run the PeopleSoft Pure Internet Architecture setup program for your database.	See The PeopleTools 8.50 installation guide for your database platform.

## Configuring Your Web Profile

During PeopleSoft Pure Internet Architecture setup, you'll be prompted to select one of the web profiles delivered with your PeopleSoft application, or to specify a new web profile if none of the delivered profiles meet your needs. To configure your selected web profile, you perform the following step:

<b>Step</b>	<b>Reference</b>
Configure your selected web profile.	<p>See <a href="#">Chapter 10, "Configuring the Portal Environment," Configuring Web Profiles, page 216.</a></p> <p>See <i>Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration</i>, "Working with Oracle WebLogic."</p> <p>See <i>Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration</i>, "Working with IBM WebSphere."</p>

## Configuring Application Security

Applications comes with roles and permissions that you can configure or use as delivered. Roles define the permission lists assigned to groups of people. Permission lists define a group of securable objects. A comprehensive role and permission list design is necessary to best manage which users can access each piece of content in the portal.

A powerful feature of PeopleTools security is the dynamic role. User membership to these roles is defined programmatically instead of by manually updating a membership list. Dynamic roles can simplify user role assignments significantly.

To secure access to content, you perform the following steps:

<b>Step</b>	<b>Reference</b>
Configure permission lists, roles, and user profiles.	See <i>Enterprise PeopleTools 8.50 PeopleBook: Security Administration</i> .
(Optional) Set up single sign-on.	<p>See <i>Enterprise PeopleTools 8.50 PeopleBook: Security Administration</i>, "Implementing Single Signon."</p> <p>See <a href="#">Chapter 10, "Configuring the Portal Environment," Implementing Single Signon Functionality, page 257.</a></p> <p>See <i>Enterprise PeopleTools 8.50 PeopleBook: Security Administration</i>, "Employing LDAP Directory Services."</p>

## Setting Up the Portal

To set up the portal, you perform the following steps:

<b>Step</b>	<b>Reference</b>
Administer portal definitions.	See <a href="#">Chapter 5, "Administering Portals," Administering Portal Definitions, page 101.</a>
Define folders and set folder security.	See <a href="#">Chapter 5, "Administering Portals," Administering Folders, page 105.</a>
Define content references, create related links, and set content reference security.	See <a href="#">Chapter 5, "Administering Portals," Administering Content References, page 109.</a>

<b>Step</b>	<b>Reference</b>
Add templates and set template security.	See <a href="#">Chapter 5, "Administering Portals," Managing Portal Objects, page 139.</a>
Create tabbed homepages and specify tab layouts.	See <a href="#">Chapter 6, "Administering Portal Homepages and Pagelets," page 153.</a>

## (Optional) Setting Up JSR 168

JSR 168 is an emerging standard for portals running in a Java 2 Platform, Enterprise Edition architecture. It specifies content definitions much like pagelets, which present portal content defined according to the JSR 168 standard. Portlets are Java-based web components that you can register in the PeopleSoft portal. To set up JSR 168, perform the following step:

<b>Step</b>	<b>Reference</b>
Implement JSR 168 portlets.	See <a href="#">Chapter 18, "Working with JSR 168 Portlets," Describing the Portlet Entity Registry, page 541.</a>

## Configuring Performance Features

When properly configured, caching significantly boosts portal performance. To configure performance features for the portal, perform the following steps:

<b>Step</b>	<b>Reference</b>
Administer server-based caching.	See <a href="#">Chapter 9, "Using Portal Caching Features," Administering Server-Based Caching, page 200.</a>
Administer browser-based caching.	See <a href="#">Chapter 9, "Using Portal Caching Features," Administering Browser-Based Caching, page 207.</a>
Administer web server-based navigation caching.	See <a href="#">Chapter 9, "Using Portal Caching Features," Administering Web Server-Based Navigation Caching, page 211.</a>

## Chapter 2

# Understanding PeopleSoft Pure Internet Architecture

This chapter discusses:

- PeopleSoft Pure Internet Architecture fundamentals.
- Database server.
- Application servers.
- Batch server environment.
- Web server.
- Web browser.
- Configuration and deployment options.

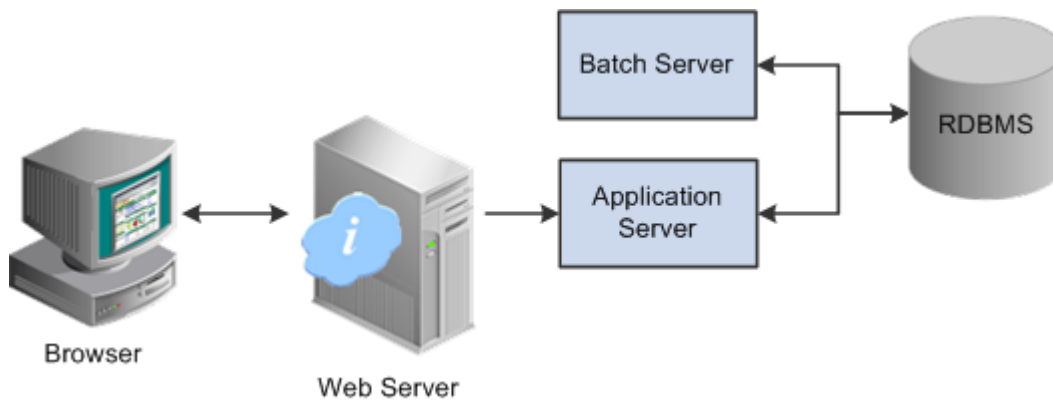
---

## PeopleSoft Pure Internet Architecture Fundamentals

Your PeopleSoft application runs on PeopleSoft Pure Internet Architecture, which requires a variety of software and hardware elements:

- A relational database management system (RDBMS) server.
- An application server.
- A batch server.
- A web server.
- A web browser.

It's important to understand the role of each element before you can decide which configuration will work best for your implementation of PeopleSoft Pure Internet Architecture. The following diagram illustrates, at a high level, the physical relationship between PeopleSoft Pure Internet Architecture elements:



High-level PeopleSoft Pure Internet Architecture relationships

Configuring PeopleSoft Pure Internet Architecture is not just about enabling internet application deployment through a browser. PeopleSoft Pure Internet Architecture enables you to take advantage of all PeopleSoft intranet and internet solutions, as well as PeopleSoft integration technologies such as PeopleSoft Integration Broker.

See the PeopleSoft platforms database on the PeopleSoft Customer Connection website.

---

## Database Server

The database server houses a database engine and your PeopleSoft application database, which includes all the application's object definitions, system tables, application tables, and data. The database server must be running one of the PeopleSoft-supported RDBMS and operating system combinations.

A single database server can have multiple application servers connecting to it. The database server simultaneously handles the application server connections, development environment connections, and batch programs running against it.

---

**Note.** Using the PeopleTools development environment in Microsoft Windows, you can connect directly to the database, or indirectly through an application server.

---

The PeopleSoft database is the repository for all information managed by your PeopleSoft application. Both application data and PeopleSoft metadata are stored and maintained in the database. PeopleSoft Application Designer enables you to define and maintain this metadata, which the system uses to drive the runtime architecture. The application server runs business logic based on the metadata.

Use PeopleSoft Application Designer to define, configure, and modify PeopleSoft applications. You can create dozens of different types of application objects, such as fields, records, pages, and messages. When an application developer saves an application object, PeopleSoft Application Designer saves this definition to the metadata repository in the PeopleSoft database.

At runtime, the application server fetches the most recent application object definitions from the metadata repository, compiles and caches the application object into memory, and runs the business rules based on the definitions.

---

## Application Servers

This section discusses:

- Application servers.
- Domains.
- Listeners, handlers, and queues.
- PeopleSoft server processes.
- Services.
- Oracle products.
- Database connectivity.

## Application Servers

The application server is the core of PeopleSoft Pure Internet Architecture; it runs business logic and issues SQL to the database server. An application server consists of numerous PeopleSoft services and server processes. Just as different elements make up the physical environment in which an application server operates, such as database servers and web servers, a variety of elements operate on the application server, enabling it to respond effectively to a multitude of transaction requests and handle transaction processing, system scaling, browser requests, and so on.

An application server maintains the SQL connection to the database for browser requests and the PeopleTools development environment in Microsoft Windows.

You should have a general knowledge of how an application server operates before you attempt to configure and tune it.

### See Also

*Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*

## Domains

An application server domain is the collection of server processes, supporting processes, and resource managers that enable connections to the database. You manage each domain with a separate configuration file, and you configure each application server domain to connect to a single database. A single application server machine can support multiple application server domains running on it. You configure an application server domain using the PSADMIN utility located in the *PS\_HOME*\appserv directory on the application server.

There can be a one-to-one or a many-to-one relationship between application server domains and a database. In the simplest case, you configure a single application server domain to connect to a single PeopleSoft database. In a more sophisticated environment, you may configure multiple application server domains, with each domain connecting to the same PeopleSoft database. The opposite is not valid; a single application server domain cannot be used to connect to multiple PeopleSoft databases.

For example, suppose you have installed three databases, HRDMO1, HRDMO2, and HRDMO3, and you want to enable browser requests to each database. In this case, you must configure at least three application server domains, one for each database. As demand increases, you may need to configure multiple application server domains per database, for redundancy, fail-over, and performance reasons.

You can configure multiple application server domains under a single PeopleSoft configuration home directory, or *PS\_CFG\_HOME*. In this context, *PS\_CFG\_HOME* refers to the PeopleSoft high-level directory on the application server, not the file server. *PS\_CFG\_HOME* is the directory to which you installed the PeopleSoft application server configuration files when using the Server Transfer program.

PSADMIN creates a directory beneath *PS\_CFG\_HOME*\appserv for each application server domain that you configure. Using the previous HRDMO example, suppose you decided to name the application server domains the same name as the database to which they connect. In this case, PSADMIN creates subdirectories \HRDMO1, \HRDMO2, and \HRDMO3 beneath the *PS\_CFG\_HOME*\appserv directory on the application server.

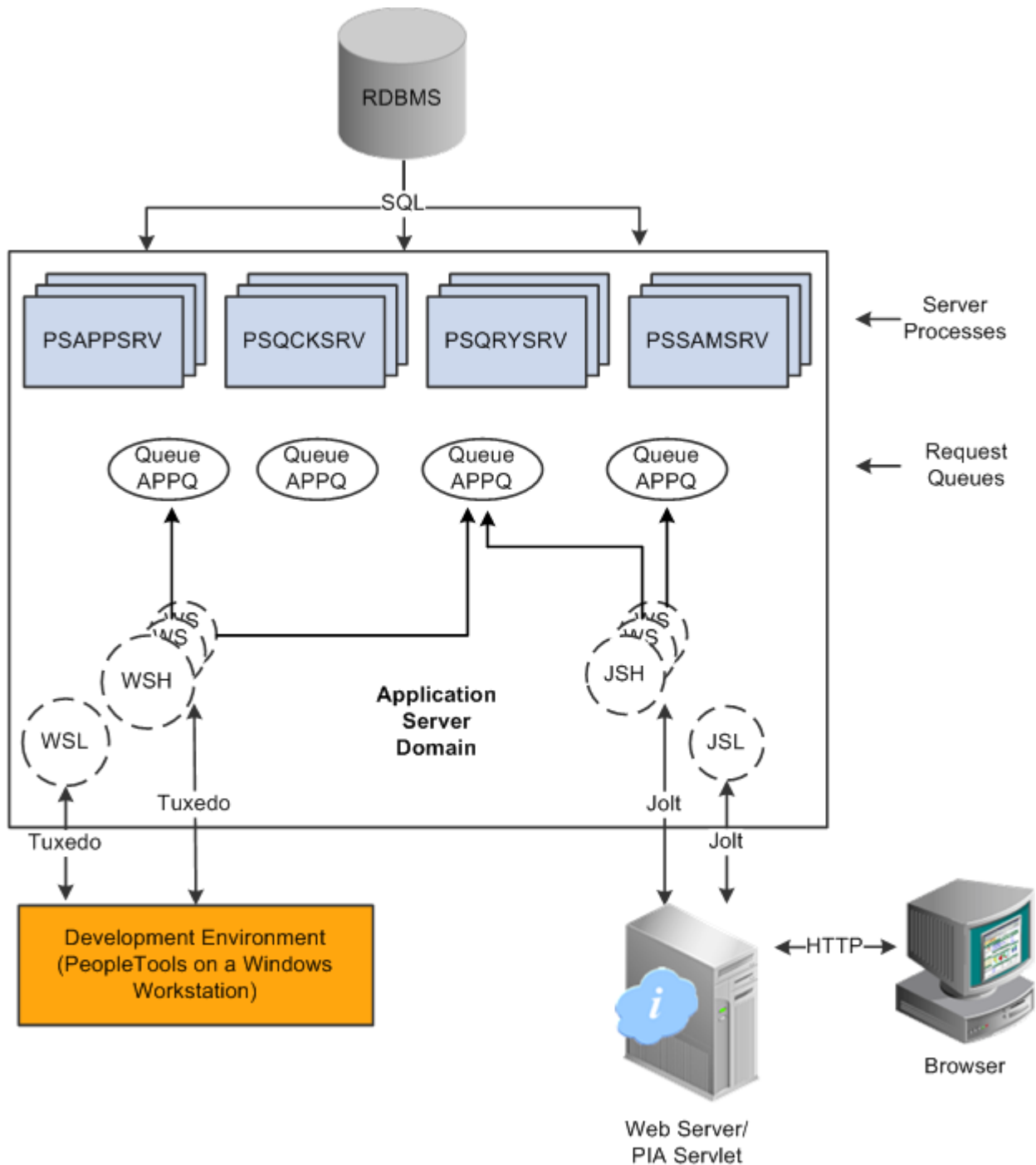
When you boot an application server domain, it starts the set of server processes associated with that domain, such as PSAPPSRV, PSQCKSRV, and PSSAMSRV. Each server process establishes a persistent connection to a PeopleSoft database, and this connection acts as generic SQL pipeline that the server process uses to send and receive SQL.

Each server process uses the same SQL connection to facilitate requests from multiple sources. From the RDBMS perspective, each server process within a domain represents a connected user.

## Listeners, Handlers, and Queues

Listeners, handlers, and queues provide the basis of the application server functionality. Understanding the role of each element is essential when you configure and tune your application server. For instance, although it is important to know how to configure multiple Java server handlers, it is equally important to know why and when to perform this task.

The following diagram illustrates application server processes:



### Application server components and server processes

For simplicity, the diagram does not depict every server process that runs on the application server. For example, the optimization server process, **PSOPTENG**, is not pictured, and integration server processes were also left out.

**Note.** When discussing PeopleSoft architecture mechanics, the term *service* becomes overused. The following statement may help to clarify this term: An application server domain calls server processes, such as **PSAPPSRV**, which in turn invoke services, such as **MgrGetObject**, on the database.

The following table describes each component depicted in the previous diagram:

<b><i>Item</i></b>	<b><i>Description</i></b>
Workstation listener (WSL)	The workstation listener monitors Oracle Tuxedo ports for initial connection requests sent from the PeopleTools development environment. After the workstation listener accepts a connection from a workstation, it directs the request to a workstation handler. From that point, the Microsoft Windows workstation interacts with the workstation handler to which it is assigned.
Workstation handler (WSH)	The workstation handler processes the requests it receives from the workstation listener. A unique port number identifies a workstation handler. The port numbers for the workstation handler are selected (internally by Oracle Tuxedo) from a specified range of numbers. You can configure multiple workstation handlers to take care of demand increases; new processes are created as other processes become overloaded.
Oracle Jolt server listener (JSL)	The Oracle Jolt server listener applies only to browser requests. The Oracle Jolt server listener monitors the Oracle Jolt port for connection requests sent from the browser through the web server. After the Oracle Jolt server listener accepts a connection, it directs the request to a Oracle Jolt server handler. From that point, the browser interacts with the Oracle Jolt server handler. This is analogous to the relationship between the workstation server listener and workstation server handler.
Oracle Jolt server handler (JSH)	The Oracle Jolt server handler applies only to browser requests. The Oracle Jolt server handler processes the requests it receives from the Oracle Java server listener. The port numbers for the Oracle Jolt server handler are selected internally by Oracle Tuxedo in sequential order.
Request queues	Each type of server process has a service request queue that it shares with other servers of the same type (as in PSAPPSRV on APPQ, PSQCKSRV on CCKQ). The workstation handler and Oracle Jolt server handler insert requests into the appropriate queue, and then the individual server processes complete each request in the order that it appears.

<i>Item</i>	<i>Description</i>
Server processes	The server processes act as the heart of the application server domain. They maintain the SQL connection and make sure that each transaction request gets processed on the database and that the results are returned to the appropriate origin.

## PeopleSoft Server Processes

Multiple server processes run in an application server domain. A server process is executable code that receives incoming transaction requests. The server process carries out a request by making calls to a service, such as MgrGetObject.

Server processes invoke services to perform application logic and issue SQL to the RDBMS. Each application server process, such as PSAPPSRV, PSQCKSRV, PSQRYSRV, PSSAMSRV, or PSOPTENG, establishes and maintains its own connection to the database.

The server process waits for the service to complete, then returns information to the device that initiated the request, such as a browser. While a server process waits for a service to complete, other transaction requests wait in a queue until the current service completes. A service may take a fraction of a second to complete or several seconds, depending on the type and complexity of the service. When the service completes, the server process is then available to process the next request in the corresponding queue.

You need to configure only those server processes that your implementation requires per domain. The minimum server processes that a domain requires are PSAPPSRV and PSSAMSRV.

You can configure multiple instances of the same server processes to start when you boot the application server domain. This helps you to handle predicted workloads. Furthermore, Oracle Tuxedo can dynamically spawn incremental server processes to handle increasing numbers of transaction requests. The capability to configure multiple server processes and spawn incremental server processes contributes to the application server's scalability.

The following list describes the possible server processes included in an application server domain. Depending on the configuration options that you choose, not all of the server processes will necessarily be a part of every domain.

The basic PeopleSoft server processes are:

- PSAPPSRV

This process performs functional requests, such as building and loading components (which were known as panel groups in previous releases). It also provides the memory and disk-caching feature for PeopleTools objects on the application server. PSAPPSRV is required to be running in any domain.

- PSQCKSRV

This process performs quick, read-only SQL requests. This is an optional process designed to improve performance by reducing the workload of PSAPPSRV.

- PSQRYSRV

This process is designed to handle any query run by PeopleSoft Query. This is an optional process designed to improve performance by reducing the workload of PSAPPSRV.

- PSSAMSRV

This SQL application manager process handles the conversational SQL that is mainly associated with PeopleSoft Application Designer. This process is required to be running on any domain.

- PSOPTENG

This optimization engine process provides optimization services in PeopleSoft Optimization Framework. You need to configure this process in a server domain only if you want to use the optimization plug-in delivered with PeopleSoft applications.

The following set of server processes is used for application messaging. (Your messaging domain must also contain PSAPPSRV and PSSAMSRV, the required server processes.)

- PSMSGDSP

- PSMSGHND

- PSPUBDSP

- PSPUBHND

- PSSUBDSP

- PSSUBHND

---

**Note.** You can examine servers by using the `ps -ef` command in UNIX or Task Manager in Microsoft Windows. The PeopleSoft configuration utility, PSADMIN, also offers a monitoring utility.

---

## Services

When a PeopleSoft application sends a request to the application server, it sends a service name and a set of parameters, such as `MgrGetObject` and its parameters. Oracle Tuxedo then queues the transaction request to a specific server process that is designed to handle certain services.

When a server process boots, it advertises to the system the predefined services it handles. You can see the association between the many services and server processes by reviewing the `PSAPPSRV.UBB` file.

### See Also

*Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*, "Using the PSADMIN Utility," Using PSADMIN Executables and Configuration Files

## Oracle Middleware

PeopleSoft uses Oracle Tuxedo, a middleware framework and transaction monitor, to manage database transactions. PeopleSoft also uses Oracle Jolt, a Java API and class library, as the layer that facilitates communication between the PeopleSoft servlets on the web server and the application server. Both Oracle Tuxedo and Jolt are required.

Although we document the Oracle components with respect to their function within the context of a PeopleSoft environment, we do not duplicate the documentation provided by Oracle. You should become familiar with the Oracle documentation that we ship along with our products. The Oracle documentation provides an extensive error catalog that serves as an invaluable source of information when troubleshooting.

---

**Note.** Oracle Tuxedo doesn't actually perform processing on the application server; it schedules PeopleSoft server processes to perform the transactions.

---

See Enterprise PeopleTools 8.50 installation guide for your database platform.

## Database Connectivity

Application servers require database connectivity software installed locally to maintain the SQL connection with the RDBMS. You must install the required connectivity software and associated utilities for your RDBMS.

After the application server establishes a connection to the database, any device that initiates a transaction request through the application server takes advantage of the application server's direct connection to the database and therefore requires no connectivity software.

---

## Batch Server Environment

This section provides discusses:

- Batch server environment.
- Batch server support.
- PeopleSoft Process Scheduler and the application server.

## Batch Server Environment

The batch server environment is where you have PeopleSoft Process Scheduler installed and configured, and it is the location where many of your batch programs, such as Application Engine programs, run. In most situations, this is also where you have COBOL and SQR executables installed.

In a multiserver environment, you can decide where your sites' batch server environment resides. In PeopleSoft, you can install the batch server on a separate server, or it can run on either the application server or the database server.

## See Also

*Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Process Scheduler*, "Understanding PeopleSoft Process Scheduler"

## Batch Server Support

You can install PeopleSoft Process Scheduler on any supported application server, database server, or batch server, but it's important that you choose a location that's supported in the PeopleSoft environment. There are at least two options for each database environment.

If you install PeopleSoft Process Scheduler on a machine that is a supported database machine, but not a supported application server machine, you can still configure and administer PeopleSoft Process Scheduler using PSADMIN, the PeopleTools server administration utility. However, you won't be able to access application server options from PSADMIN.

## PeopleSoft Process Scheduler and the Application Server

PeopleSoft uses PSADMIN to configure and administer both the application server and PeopleSoft Process Scheduler server. The PeopleSoft Process Scheduler setup procedure in PSADMIN provides a menu-driven interface to configure PeopleSoft Process Scheduler parameters and administer the Process Scheduler server agent.

Even though the application server and PeopleSoft Process Scheduler have PSADMIN as a common interface and share the directories under *PS\_HOME* on the application server, they are separate entities. For instance, you boot, configure, and shut down the application server and the PeopleSoft Process Scheduler server separately.

The application server uses Oracle Tuxedo to schedule transaction requests and send transaction results, and it issues SQL requests to the database server using the persistent connections of a collection of PeopleSoft application server processes.

In contrast, PeopleSoft Process Scheduler is a separate facility that's designed to poll the PeopleSoft database table, PSPRCSRQST, for inserted requests and initiate Application Engine, COBOL, SQR, and other batch processes.

---

## Web Server

A Java-enabled web server is required to support browser transaction requests and PeopleSoft application messaging technology. You install on the web server a collection of PeopleSoft Java servlets designed to handle a wide range of PeopleSoft transactions.

This section discusses:

- Server software elements.
- PeopleSoft servlets.
- Oracle Jolt.

## Server Software Elements

During the PeopleSoft installation, a variety of PeopleSoft Java servlets are installed on the web server, so, you must have a supported servlet engine installed.

The following software runs on the PeopleSoft Pure Internet Architecture web server:

- Web services.

Web services software manages the web server, such as WebLogic or WebSphere.

- Servlet engine.

The servlet engine is the environment in which servlets run. This component is tied to the web services software, but in some cases you install it separately.

- Java servlets.

Java is a platform-independent programming language used widely for web-based programs. Servlets are Java programs that run on the web server. The Java executables are required for the servlet engine to operate.

## PeopleSoft Servlets

The following PeopleSoft servlets reside on the web server:

- Portal servlet.

The portal servlet handles all of the requests and formatting for the users accessing PeopleSoft through PeopleSoft portal technologies. It manages search, content management, and homepage personalization.

- Integration gateway servlet.

This servlet transmits publish/subscribe messages between message nodes. The gateway handles PeopleSoft-to-PeopleSoft messages, PeopleSoft-to-third party messages, and third party-to-PeopleSoft messages.

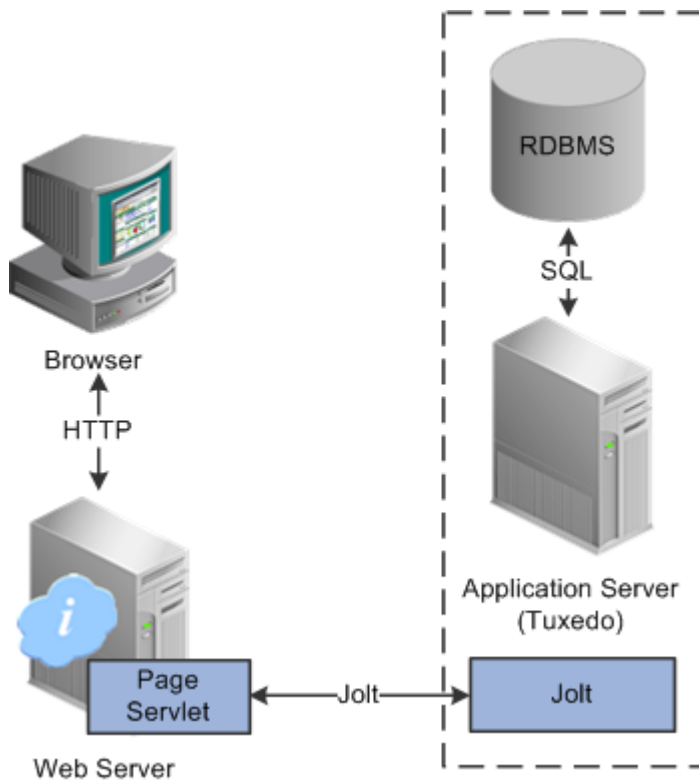
- Report repository servlet.

This servlet enables users to easily access and distribute the output of batch reports, such as Crystal and SQR, run through PeopleSoft Process Scheduler over the internet. This servlet retrieves the report output in the report repository and serves it to the browser.

## Oracle Jolt

The PeopleSoft servlets on the web server transmit requests and data through a connection to Oracle Jolt, which runs on the application server. Oracle Jolt extends Oracle Tuxedo capabilities to the communication layer between the web-based environment and the C++ environments. You configure the servlets to direct requests from the web server to a predefined Oracle Jolt port on the application server.

Oracle Jolt must coexist with Tuxedo on the same application server machine. Oracle Jolt can't function without Tuxedo. The following diagram shows the relationship between PeopleSoft Pure Internet Architecture components:



Oracle Jolt in relation to other components

Web browsers don't connect directly to the application server. Instead, they send HTTP requests to the portal servlet running on the web server. The web server translates the HTTP request into a Oracle Jolt request that is sent to a specified Jolt port. Then the application server itself, leveraging Oracle Tuxedo, runs the appropriate SQL against the database.

---

## Web Browser

The web browser is the primary means by which users and administrators access PeopleSoft applications and administrative tools.

You must make sure that a supported browser is installed on each workstation. You do not need to install other software on the workstation running the browser, such as applets or connectivity software. The system sends pure HTML to the browser.

A web browser uses the HTTP protocol. The browser sends a request to the web server, which forwards the request to the application server. A servlet installed on the web server facilitates all browser connections.

The browser does not download any applets to complete a transaction. The application server sends only the following to the browser:

- HTML

- XML
- JavaScript
- Cookies

Because the browser processes only this basic internet content, the client workstation is not burdened with unnecessary processing responsibility.

PeopleSoft Pure Internet Architecture leverages web browser cookies to store a unique access token for each user when the user is initially authenticated. When the user connects to another PeopleSoft system, the token in the browser cookie is used to reauthenticate the user and bypass the sign-in process. The browser cookie is an in-memory cookie and is never written to disk. The cookie is also encrypted to prevent snooping and check-summed to prevent tampering.

See *Troubleshooting Browser Limitations* document available from Customer Connection.

---

**Note.** With PeopleSoft Pure Internet Architecture, there is no traditional client software installation. Most processing occurs at the server level. PeopleSoft still supports the PeopleTools development environment in Microsoft Windows, which is intended for application developers and system administrators who need access to PeopleTools. These users require workstations running Microsoft Windows.

---

### ***Web Browser Cookies Setting***

To use PeopleSoft Pure Internet Architecture, the Internet option to allow session cookies must be enabled.

To enable session cookies:

1. From your browser menu, select Tools, Internet Options, Security, Custom Level.
2. Scroll down to the Cookies – Allow per-session cookies (not stored) option.
3. Select Enable.
4. Click OK to save the setting.
5. Click OK again, to close the dialog window.

---

**Note.** The Internet option to allow cookies that are stored on your computer is optional.

---

---

## **Configuration and Deployment Options**

This section discusses:

- Configuration options.
- Deployment options.

## Configuration Options

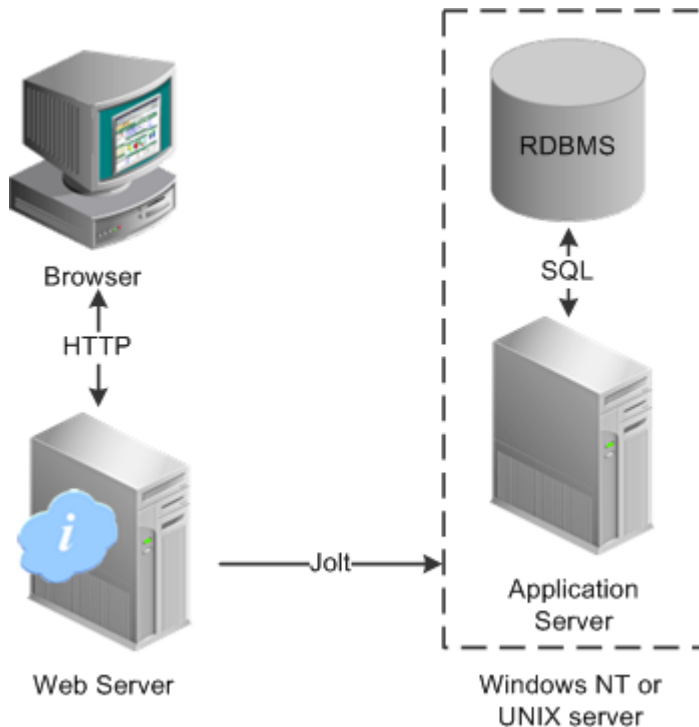
You can configure your environment to support either a physical or a logical application server configuration. In some cases, the PeopleSoft standard installation procedure recommends one or the other depending on the combination of database and operating system at your site.

Generally, your application server should be on the same physical machine as the database server. This produces a logical separation between the application server and database server. If the application server and database server don't reside on the same machine, then the application server and the database server should be connected to the same high-performance backbone network. This ensures optimum performance.

### ***Logical Application Server Configuration***

A logical application server environment means that one or more servers share the same physical machine. The servers are logically, but not physically, separate.

The following diagram depicts a logical configuration with two server machines—one for the web server, and the other for the application and database servers:



Logical application server configuration

The solid line surrounding the application server and the database server represents one physical machine. In this case, a logical application server configuration is possible only when both the database server and the application server are supported on a particular operating system. This shows that certain PeopleSoft Pure Internet Architecture elements can share the same machine.

Although this diagram depicts the application server and the database server sharing the same machine, the web server could also reside on the same machine with both the application server and the database server. The only requirement is that each component be supported by the underlying operating system. If all servers are located on the same machine, however, you should consider security issues. If you're deploying PeopleSoft applications to the internet, you will most likely want your web server outside of your network firewall and not on the same machine as the database server.

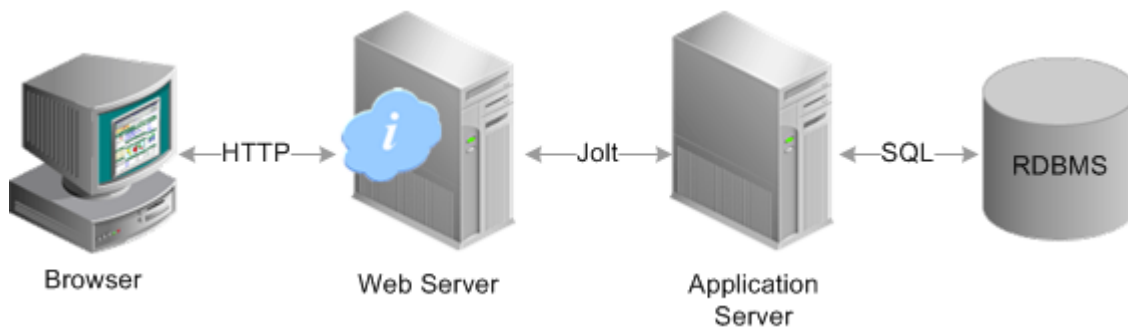
---

**Note.** For development, testing, or training purposes, you might want to have all PeopleSoft Pure Internet Architecture elements on the same Microsoft Windows or UNIX machine.

---

### ***Physical Application Server Configuration***

A physical application server configuration means that each component resides on a separate machine. The following diagram depicts a physical application server configuration:



Physical application server configuration

## **Deployment Options**

There are a variety of user deployment options.

### ***PeopleTools Base Portal***

You can use the PeopleTools base portal to deploy PeopleSoft applications to a browser. The portal enables you to integrate PeopleSoft content with content from other data sources. The PeopleSoft portal can stand on its own, or you can integrate it with any enterprise portal that you may already be using.

PeopleSoft portal technology consists of the portal servlet and an application server. These two components work together to provide common portal processing, such as page assembly, search ability, content management, navigation, and homepage personalization. You can combine content from a wide variety of data sources and deliver the result to users in a unified, simple-to-use interface.

### ***Development Environment***

Although the majority of users connect using browsers, application developers and system administrators, who require access to PeopleTools, must use Microsoft Windows workstations. PeopleSoft Application Designer and various other PeopleTools applications are not accessible through a browser.

With the PeopleTools development environment in Microsoft Windows, you can connect directly to the database (two-tier), or you can connect through an application server (three-tier).

## ***Integration Technologies***

Although PeopleSoft integration solutions don't deploy a user interface, they do enable you to share information with third-party systems and other PeopleSoft databases. To take advantage of these integration solutions, you must configure PeopleSoft Pure Internet Architecture. PeopleSoft provides the following integration technologies:

- PeopleSoft Integration Broker.

This middleware technology facilitates synchronous and asynchronous messaging among internal systems and with trading partners, while managing message structure, message format, and transport disparities.

- PeopleSoft Component Interfaces.

This object-oriented, request/reply, component architecture enables third-party applications to synchronously invoke PeopleSoft business logic.

- PeopleSoft Business Interlinks.

This plug-in framework enables PeopleSoft applications to easily invoke third-party application programming interfaces over the internet.

## ***See Also***

*Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Integration Broker*, "Understanding PeopleSoft Integration Broker"

*Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Component Interfaces*, "Understanding Component Interfaces"

## Chapter 3

# Understanding Portal Technology

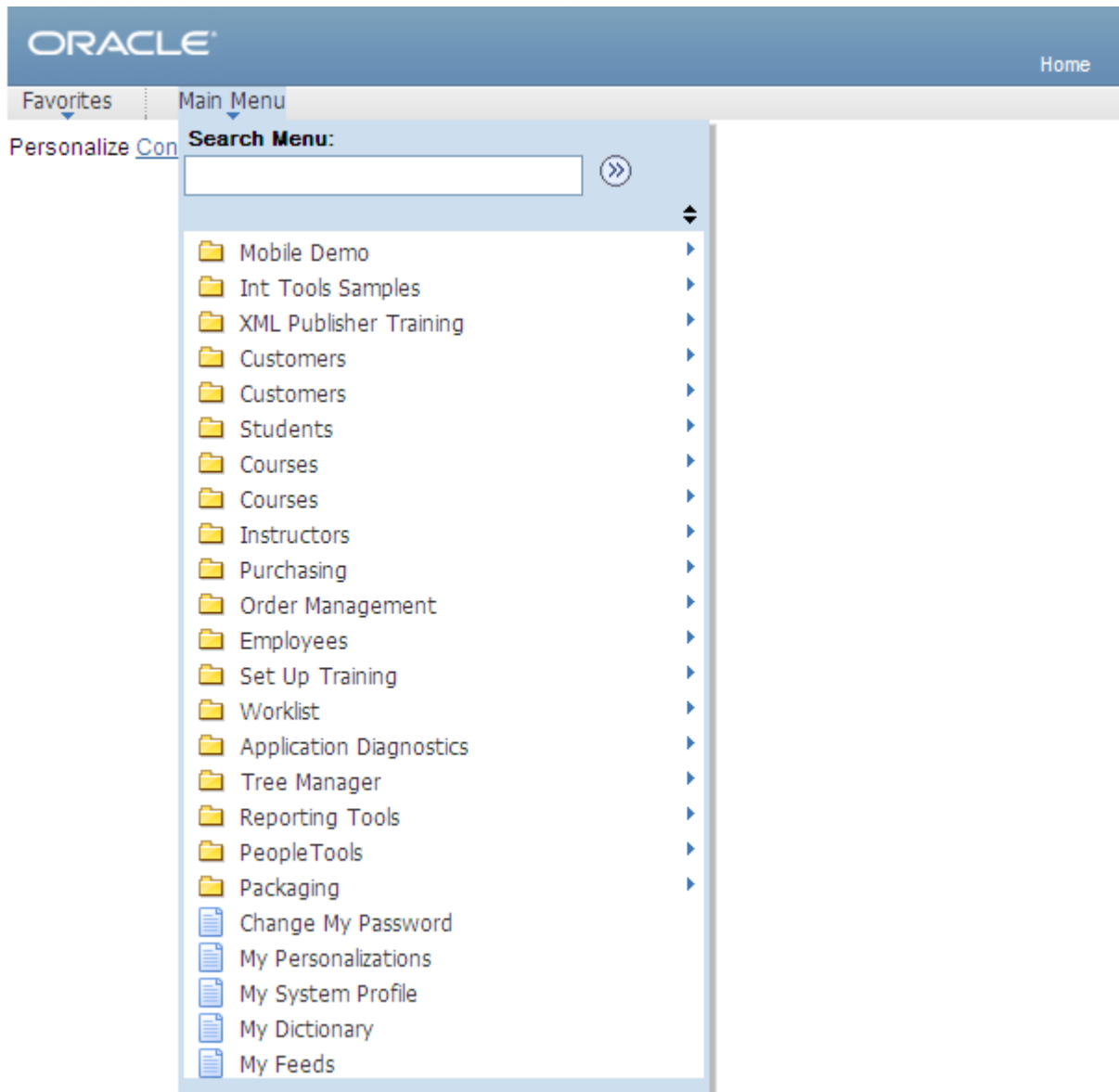
This chapter discusses:

- Portal architecture.
- Node definitions.
- Portal templates and template pagelets.
- Navigation.
- Portal registry.
- Portal servlets.
- Portal URL formats.
- Page-based template proxying.
- 

---

## Portal Architecture

PeopleSoft portal technology provides basic web-based navigation for application pages. The default portal homepage contains the drop-down menu navigation and the search area, as shown in the following example:



Home page showing drop-down menu and Search field.

The main features of the portal architecture are:

- Templates and template pagelets.
- Ability to register PeopleSoft transactions.
- The portal servlet.
- Navigation.
- Favorites.
- Related information.
- Search.

- Single-signon support for PeopleSoft applications.

---

## Node Definitions

A node is a source of HTML content. A node is a URL string that defines the entire database and server. It is used when the portal servlet attempts to retrieve content, whether internal PeopleSoft or external references, and assemble pages.

The use of nodes simplifies PeopleCode programming, since long URLs can be referred to by the appropriate node name. Some node names, such as HRMS, EPM, SA, FDM and CRM, are preset in your portal. You can add additional nodes.

---

**Note.** Node names can consist of any combination of letters, digits, and underscores, but they must not contain any spaces or special characters.

---

### See Also

*Enterprise PeopleTools 8.50 PeopleBook: PeopleCode API Reference*, "PortalRegistry Classes," Node Class

*Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Integration Broker Administration*, "Adding and Configuring Nodes"

---

## Portal Templates and Template Pagelets

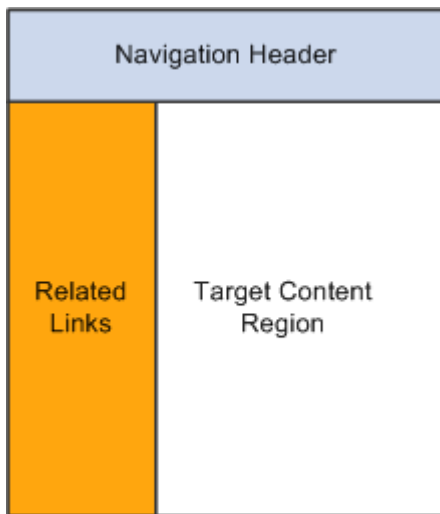
Developers create portal templates in PeopleSoft Application Designer. At runtime, a template is constructed into a web page by PeopleTools portal technology. Each template is made from various template pagelets.

Each template (and each template pagelet) is designed individually and stored as an HTML object in the application database. PeopleSoft delivers a default portal template for each installed PeopleSoft database—such as HRMS, ERP, and so on.

In addition to template pagelets, portal templates also contain special PeopleSoft tags that indicate where template pagelets are to be inserted in the template. These XML tags specify one or more regions of a page, the insertion point of the target page, and any other template pagelets that provide HTML for the other regions.

Each portal template reserves space for a target page, which contains the specific HTML page that a user requested. For example, if a user is running a PeopleSoft Financial application, the page that the user is currently accessing appears in the target region, in addition to the other template pagelets displayed elsewhere on the page. The target region is typically the largest area of the template.

The template in the following example comprises three separate template pagelets: one for the navigation header, one for related links, and one for the target content region. At runtime, the target content region is filled by the HTML returned by the target page, as are the other template pagelet regions.



Three template pagelets comprising a portal template

---

## Navigation

You can navigate in an application by using the menu and the universal navigation header.

Menu navigation is provided through a drop-down menu that enables you to move through the portal registry visually, using folders. Using the drop-down menu, you can view the registry by drilling down into the folders, accessing the favorites list, or using the search field.

The universal navigation header includes several links.

<b>Search</b>	Enter a term or phrase in the Searchfield and click the Go button to search for a registered content reference within the system.
<b>Home</b>	Click to return to the homepage.
<b>Worklist</b>	<p>Click to access the Worklist page, which provides summary information about all items on your personal worklist. This page also provides links enabling you to view additional details about the work, perform the indicated work, and reassign work items. This is an optional setting that your system administrator enables; it might not display to all users.</p> <p>See <i>Enterprise PeopleTools 8.50 PeopleBook: Using PeopleSoft Applications</i>, "Using Workflow," Working with Received Notifications.</p>

<b>MultiChannel Console</b>	Click to access and manage tasks assigned to you as an agent in PeopleSoft MultiChannel Framework. This is an optional setting that your system administrator enables. If you do not have access to this feature, the system displays a warning message to that effect.  <i>See Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft MultiChannel Framework, "Configuring PeopleSoft MCF Agents" and Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft MultiChannel Framework, "Managing Tasks and Using Chat in PeopleSoft MultiChannel Framework."</i>
<b>Performance Trace</b>	Click to monitor the performance of a business process that you typically complete. This is an optional setting that your system administrator enables; it might not display to all users.
<b>Add to Favorites</b>	Click to add the current page to your favorites list for easy accessibility.
<b>Sign out</b>	Click to terminate the user session and return to the Sign In page.

### See Also

*Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration, "Configuring Search and Building Search Indexes"*

*Enterprise PeopleTools 8.50 PeopleBook: Using PeopleSoft Applications*

*Enterprise PeopleTools 8.50 PeopleBook: Workflow Technology*

---

## Portal Registry

Each portal is defined by a portal registry. The portal registry is a set of dedicated PeopleSoft database tables that store every content reference, typically a URL, available through the portal. A portal registry has a tree-like structure in which content references are organized, classified, and registered. A portal registry contains folders and content references.

Content references are definitions that have been entered or registered in the portal registry. Content references fall into four categories: homepage tabs, target content, templates, and pagelets. Although each type of content reference differs slightly from the other types, every content reference includes information, such a name, label, creator, effective date, and expiration date.

Content references include a URL, which points to portal content. The content can be PeopleSoft components, iScripts, external web pages, or other external content. The URL can point to any web site that responds to HTTP or HTTPS requests with an HTML response—any static or dynamic web page. One example of a content reference is a URL that points to a PeopleSoft application component. Other examples include static HTML pages on an intranet site or dynamic pages created by a reporting system. Access to content references is controlled by permission lists assigned when the content reference is created.

In the portal registry, folders organize content references into a multilevel hierarchy. Each folder can contain content references as well as other folders. Folders have many of the same properties as content references.

Every portal registry contains a root folder and a Portal Objects folder. The Portal Objects folder includes the following folders: Templates, Pagelets, Template Pagelets, and Homepage, Navigation Collections, and WSRP Produced Portlets.. In addition to these standard folders, one folder per PeopleSoft application located directly below the root folder. These application folders contain the folders and content references associated with each PeopleSoft application that you license. All application pages in PeopleSoft product line databases are registered, secured, and described in the portal registry.

The hierarchical structure of the registry enables a portal administrator to create a classification and navigation system in which content references can be registered and managed for all portal users. PeopleSoft provides a web-based portal administration utility for this purpose. Additionally, a registry API is provided for accessing each portal registry from PeopleCode, Component Object Model (COM), or C programs.

A portal registry can reside in an existing PeopleSoft application database or in an independent PeopleSoft database. A single database can support multiple portal registries, and therefore multiple portals, but only one portal registry is associated with any given portal. To improve performance, a copy of the registry is also stored on the application server in cache files; this is how the portal servlet accesses the registry at runtime.

### ***Maintaining the Portal Registry***

There are several ways to access and maintain the portal registry:

- Use the registration wizard to register content references, assign security, and update.
- Use the Menu Import feature to upgrade custom menu group definitions.
- Use portal administration pages to add, change, or delete folders and content references from a portal registry.
- Use the portal registry API for programmatic access to the registry.
- Use the security synchronization process to update the portal registry security based on the menu and script security.

### **See Also**

Chapter 10, "Configuring the Portal Environment," Importing Menu Groups into the Portal Registry, page 260

*Enterprise PeopleTools 8.50 PeopleBook: Security Administration*

*Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Application Designer Developer's Guide, "Using the Registration Wizard"*

*Enterprise PeopleTools 8.50 PeopleBook: PeopleCode API Reference, "PortalRegistry Classes," Using the PortalRegistry API*

Chapter 3, "Understanding Portal Technology," Node Definitions, page 25

Chapter 10, "Configuring the Portal Environment," Specifying the Node Type, page 257

*Enterprise PeopleTools 8.50 PeopleBook: PeopleCode API Reference, "PortalRegistry Classes," Node Class*

*Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Integration Broker Administration, "Adding and Configuring Nodes"*

---

## Portal Servlets

This section discusses:

- Portal servlets.
- Page assembly for page-based templates.

## Portal Servlets

A portal servlet is a Java servlet that runs on the portal web server. It intercepts a user request for an HTML page, retrieves the requested page, wraps additional content around it, and then sends the resulting page to the user's browser. It enables the portal to integrate content from a wide variety of sources and present the content on a single page in a coherent, consistent fashion.

The portal servlet performs these functions:

- Retrieves user-requested content.

The content can be either target content, such as a PeopleSoft application page, for display in the large target region of the browser, or content for the smaller-sized pagelets.

---

**Note.** Portal servlets do not accept HTTP PUT requests from clients. They only accept HTTP GET or POST data from clients. If the portal servlet is sent a PUT request, a 405 error page appears.

---

See World Wide Web Consortium, Hypertext Transfer Protocol—HTTP/1.1, <http://www.w3.org/Protocols/rfc2616/rfc2616.html>.

- Provides a consistent user interface.

The portal servlet checks properties associated with each content reference, including the name of a portal template. When a user accesses content through the portal, the portal servlet wraps the target page with the portal template specified in the content reference. This template provides a consistent user interface.

- Ensures that PeopleSoft-specific tags are processed correctly.

Developers create portal pages using a template-based layout system. In addition to traditional HTML tags, templates can contain PeopleSoft-specific tags that a normal browser cannot interpret. The portal servlet can interpret these PeopleSoft-specific tags when constructing templates, and can interpret any other HTML content, at runtime. The portal servlet then sends the resulting page to a browser as a single HTML document.

The portal servlet's behavior changes slightly depending on whether it's handling page-based or frame-based templates.

### ***Page-Based Template Handling***

For page-based templates, the portal servlet assembles pages for the browser and ensures that all URL references in the HTML on the assembled pages are referenced back to the portal servlet itself.

The portal servlet receives user requests for content and then constructs a single, complete HTML page for the user's browser by combining the requested content with the appropriate portal template. This process is called *page assembly*. The portal servlet uses a template to wrap the contents of the requested page into the context of the site (headers, navigation, and so on). Each content reference can be associated with a template in the portal registry. The template tells the portal servlet what URLs to combine and how to arrange them in the user's browser.

---

**Note.** The portal servlet uses relative URLs. The portal servlet performs URL rewriting (proxying) only when the URL is generated by an external system or a PeopleTools release prior to 8.4.

---

Pages that the portal servlet assembles for the browser might point to numerous other web pages from different sources on the internet. Because the user's request goes through the portal servlet, the servlet must ensure that requests for content can be fulfilled during the page assembly process. In some cases, each URL in the HTML document assembled by the portal servlet must be rewritten to reference the portal servlet, not the originally requested URL. This process of redirecting URLs so that they point to the portal servlet is called *proxying*.

See [Chapter 3, "Understanding Portal Technology," Proxying and URL Conversion, page 38.](#)

### ***Frame-Based Template Handling***

For a frame-based template, the portal servlet updates the SRC tags in the frameset with the target content before sending the page to the browser.

The portal servlet inserts a URL into each frame in the SRC tag and sends the page along to the browser, rather than retrieving documents for the browser, as it does with page-based templates.

### ***IFrame-Based Template Handling***

For an iframe-based template, the portal servlet updates the SRC tags for each iframe element in the document with the target content before sending the page to the browser. The portal servlet inserts a URL into each frame in the SRC tag and sends the page along to the browser, rather than retrieving documents for the browser, as it does with page-based templates.

### ***Portal Servlet Setup***

A portal servlet must be set up properly on a web server before the portal can service user requests. The web server and portal servlet environment should be configured and tested during the installation of PeopleSoft applications.

See PeopleTools 8.50 Installation Guide for your web server platform.

## **Page Assembly for Page-Based Templates**

When a user clicks a link on a page-based template, the portal servlet receives the user's request, retrieves the content, properly formats it using a predefined portal template, and sends the assembled page back to the user's browser. The portal servlet also ensures that the user has the proper security permissions and is allowed to view the page.

The following processing steps occur during the page assembly process:

1. The browser sends a request for content to the portal web server.

The browser sends an HTTP request for target content; the request includes the target URL. The portal web server invokes the portal servlet. If the request includes a URL from a PeopleTools release prior to 8.4, the portal servlet then identifies the target content by looking at the query string parameter that was put there when the referring page was proxied.

2. The portal servlet checks the portal registry to see if there is a content reference for the target URL.

If there is a content reference for the target URL, and the user does not have access to the content reference, the portal servlet responds with an error message. If the user does have access, the portal servlet continues its processing. If the content reference has been registered with a frame template, the portal servlet constructs the template and returns it as the response to the browser. The browser gets the content for each frame in the usual way. If the content reference has been registered as having no template, the servlet sends a redirect response to the browser for the original content. Otherwise, it goes on to the next step.

3. The portal servlet retrieves the appropriate template.

If a registered content reference was found in the previous step, then the template associated with that content reference is used to wrap that content. If no template is associated, the portal servlet uses the template for the node associated with the content reference. If there is no node associated with the content reference, then the default template for the portal is used. If there is no default portal template, the content appears without any template.

4. The portal servlet issues HTTP requests for content.

The portal servlet issues an HTTP request for the requested content to the appropriate web server and receives an HTML document in return. It also issues an HTTP request for each pagelet used in the template.

5. The portal servlet merges content from previous versions of PeopleTools 8.4 from all HTML documents that it retrieved into the template HTML, and sends a single, complete HTML document back to the user's browser.

The template HTML may contain special PeopleSoft tags, including the Target tag and the Pagelet tag. Each Target tag is replaced with whatever content is retrieved when fulfilling the request for target content. Each Pagelet tag is replaced with the proxied content specified by that tag.

If a style sheet is associated with the template, it is used. If not, the style sheet associated with the target content is used. Style sheets included in template pagelets are used if they are present. Additionally, the portal servlet ensures that cookies and headers returned in the responses for the template pagelets are merged into the main response.

---

## Portal URL Formats

This section discusses:

- Basic portal URL format.
- URL format for PeopleSoft Pure Internet Architecture content types.
- URL format for unwrapped PeopleSoft Pure Internet Architecture content.

- Pagelet URLs.
- System URLs.
- Proxy architecture and relative URLs.

## Basic Portal URL Format

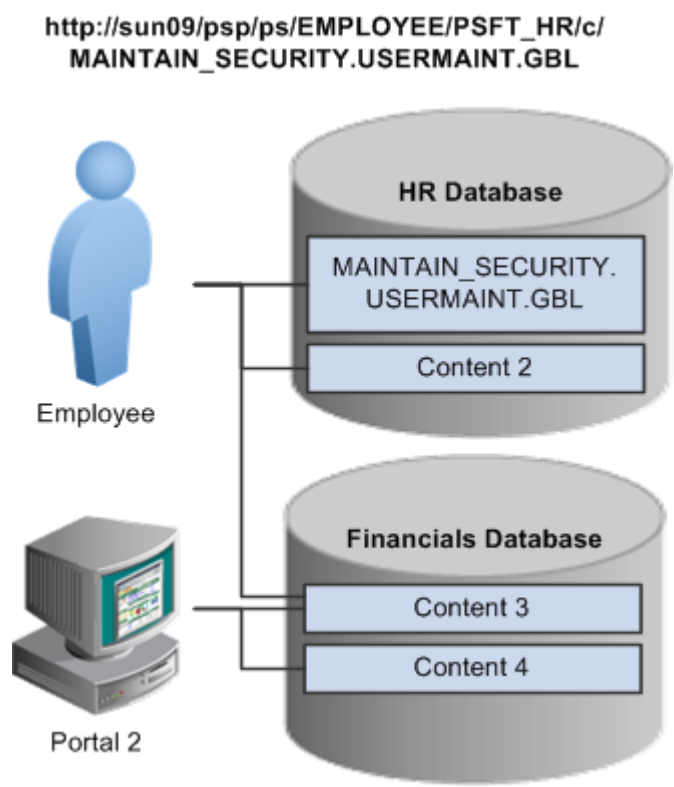
URLs provide the addresses for content, so that it can be located and correctly identified. The portal servlet needs three pieces of information in order to present a page to the user. These are integral parts of a PeopleSoft portal URL:

- The name of the portal through which content is being accessed.
- The node that hosts the content.
- The type and ID of the content.

The portal servlet uses the node name in the URL to look up the location of the servlet for that node (stored as part of the node object). If the content is hosted by the local database, the portal servlet talks to the content servlet directly (Java method calls), not through HTTP (using the portal content servlet URL).

The ID format is different for each content type. Components are identified by menu, component, and market; scripts are identified by the program name, and so on. The portal registry determines the content reference for this content, and for the template and the pagelets that appear around it.

The following diagram shows how the URL lists the required information from left to right: portal (EMPLOYEE), node (PSFT\_HR), content type (c), content ID (MAINTAIN\_SECURITY.USERMAINT.GBL):



URL structure

This is consistent with the logical organization of content in portals and databases. At the left side of the figure, portals point to nodes, and ultimately, to content within those nodes.

Here's what a portal URL looks like:

[http://server/servlet\\_name/SiteName/PortalName/NodeName/content\\_type/content\\_id?content\\_parm](http://server/servlet_name/SiteName/PortalName/NodeName/content_type/content_id?content_parm)

The following table describes the different sections of the URL:

Section	Description
<a href="http://Server/">http://Server/</a>	Scheme (HTTP / HTTPS ) and web server name.  <b>Important!</b> The server name and port in this section must not exceed 30 characters in length. For example, <i>http://mybiglongservername.peoplesoft.com:8080</i> , not including the <i>http://</i> prefix, is 39 characters—nine characters too long.
<a href="#">servlet name/</a>	The name of the physical servlet that the web server invokes to handle the request.

<b>Section</b>	<b>Description</b>
SiteName/	<p>The site name specified during PeopleSoft Pure Internet Architecture setup. This enables you to set up multiple sites on one physical web server. The site name is ultimately mapped by the web server to the appropriate configuration.properties file.</p> <p><b>Important!</b> The site name can include underscores ( _ ), but an underscore cannot be followed by a numeric character or the string "newwin" (for example, <i>my_site_3</i> or <i>my_newwin_site</i>).</p>
PortalName/	Name of the portal to use for this request. The portal definition contains metadata that describes how to present the content (template, pagelets, and so on).
NodeName/	Name of the node that contains the content for this request.
Content type/	Type of the content for this request.
content id	The identification of the content. The ID and type ensure the correct content is retrieved.
?content parm	Query string parameters (name value pairs) for the content.

## URL Format for PeopleSoft Pure Internet Architecture Content Types

This table lists the URL formats for each PeopleSoft Pure Internet Architecture content type:

<b>Content Type</b>	<b>URL Format</b>	<b>Example</b>
Component	<i>/c/menu.component.market/?Page=page&amp;Action=action &amp;key_id=key_value. . .</i>	http://sun09/ps/ps/EMPLOYEE/PSFT_HR/c/MAINTAIN_SECURITY.USERMAINT.GBL?page=view&view=narrow
Script	<i>/s/recordname.fieldname.event.function/?parm_id=parm_value. . .</i>	http://sun09/ps/ps/EMPLOYEE/PSFT_HR/s/WEBLIB_Portal.PORTAL_HEADER.FieldChange.Iscrip_DoSomething
External	<i>/e/?url=URL</i>	http://sun09/ps/ps/EMPLOYEE/PSFT_HR/e/?url=http%2f%3a%3awww.peoplesoft.com

<b>Content Type</b>	<b>URL Format</b>	<b>Example</b>
Homepage	<i>/h/?tab=homepage_tab</i>	http://sun09/psp/ps/EMPLOYEE/PSFT_HR/h/?tab=HR homepage tab  <b>Note.</b> This homepage URL tells the portal servlet to serve up the specified tab of the current user's homepage.
Query	<i>/q/query</i>	http://sun09/psp/ps/EMPLOYEE/PSFT_HR/q/my_query
Worklist	<i>/w/worklist</i>	http://sun09/psp/ps/EMPLOYEE/PSFT_HR/w/my_worklist
Navigation	<i>/n/business_process_map</i>	http://sun09/psp/ps/EMPLOYEE/PSFT_HR/n/my_BusProcMap
File	<i>/f/filename</i>	http://sun09/psp/ps/EMPLOYEE/PSFT_HR/f/myfile.html  <b>Note.</b> The file URL for file content tells the servlet to retrieve the named file from the database and return it to the browser.
Disconnect	<i>/?disconnect=y</i>	http://sun09/psc/ps/EMPLOYEE/PSFT_HR/s/WEBLIB_QETEST.ISCRIPT1.FieldFormula.IScript_WhoAmI/?disconnect=y&postDataBin=y  This is an internal flag used by xmllink to process a request and to invalidate the HTTP session afterwards. In a normal browser, the HTTP session is maintained so the subsequent requests are not creating new HTTP sessions.

## URL Format for Unwrapped PeopleSoft Pure Internet Architecture Content

PeopleSoft Pure Internet Architecture content is accessible with no template wrapping using the content servlet. This enables portals to implement a proxied architecture and enables you to include PeopleSoft Pure Internet Architecture content in other portal products and web sites.

URLs for unwrapped PeopleSoft Pure Internet Architecture content are similar to URLs for wrapped content. Unwrapped PeopleSoft Pure Internet Architecture content has not gone through the portal servlet template process. In the URL, the content servlet is specified rather than the portal servlet. The following table shows sample URLs for a component and an iScript.

---

**Note.** The content servlet ignores the portal and node name, but they are still necessary as placeholders. Omitting them causes a runtime error, since the psc servlet checks that the URL contains a portal and node name, even though it doesn't use them.

---

<i>Unwrapped Content Type</i>	<i>URL Example</i>
Component	http://sun09/psc/ps/EMPLOYEE/PSFT_HR/c/E_PRO.CheckOut.GBL
Script	http://sun09/psc/ps/EMPLOYEE/PSFT_HRm/s/WEBLIB_Portal.PORTAL_HE ADERFieldChange.Iscrip_DoSomething

## Pagelet URLs

Pagelets are snippets of HTML content that appear in one section of a template. Unlike target content, they are referenced by name within the template. Here is some sample HTML that refers to a pagelet:

```
<Pagelet Name="UniversalNavigation">
<Source Node ="LOCAL_NODE" Pagelet="MyPagelet" />
</Pagelet>
```

When the servlet resolves this tag, it generates the URL using the following:

- The name of the portal containing this template.
- The node name specified in the Source tag (for example, Node = "SomeNode").
- Content type and content name specified in the pagelet definition for the specified pagelet.

The resolved URL for this example is (assume this template is in the Employee portal):

http://sun09/psc/ps/EMPLOYEE/PSFT\_HR/s/WEBLIB\_TEST.ISCRIPT1.FieldFormula.IScript\_WhoAmI

## System URLs

System URLs do not have content or query strings; instead, they issue system commands, such as Login or Expire. System URLs can be issued to both the content and portal servlets. This table lists system URLs:

<i>URL</i>	<i>Description</i>
http://sun09/psp/ps/?cmd=expire	Closes the current session and returns the expire page.  <b>Note.</b> The system detects if the user is still active during the cmd=expire process on the server. The system leaves the session alone once detecting the user is still active so that ongoing transactions in the active window are not disturbed. However, if there is another session open simultaneously that <i>is</i> inactive, the inactive session will be closed.
http://sun09/psp/ps/?cmd=logout	Closes the current session and returns the signon page.

<i>URL</i>	<i>Description</i>
http://sun09/psp/ps/?cmd=login	Tells the servlet to return the signon page.

## Proxy Architecture and Relative URLs

A relative URL is written to an HTML document without some portion of the scheme, server, or path. When the browser downloads the document containing the relative URL, it makes the relative URL absolute by adding the scheme, server, and path of the downloaded document. Relative URLs simplify moving static documents around on web servers, because you don't have to change the URLs embedded within the documents that you move.

One portal servlet can proxy content from several other content servlets. The portal servlet acts as an intermediary in the conversation between the browser and the various content services, relaying HTTP requests and responses from the content servlet to the browser.

The portal servlet acts as a reverse proxy server, by ensuring that all URL references on portal pages point back to the portal servlet itself. The portal servlet does this by rewriting all content retrieved through the portal to contain relative URLs in appropriate URL formats.

To increase performance, you can include the custom header `UsesPortalRelativeURL` with the value *True* to indicate that the URL is already set in the correct format. All content from databases using PeopleTools 8.42 and later, generates URLs with the correct format and uses this command in the header.

### Examples of Relative PeopleSoft URLs

For example, assume the `MAINTAIN_SECURITY.USERMAINT_SELF.GBL` component is in the `PSFT_HR` node, and it is being accessed by the `EMPLOYEE` portal.

If you wanted a navigation iScript within the `PSFT_HR` node to construct a link to the `MAINTAIN_SECURITY.USERMAINT_SELF.GBL` component, add the following HTML to the response:

```
<a href="../../../EMPLOYEE/PSFT_HR/c/MAINTAIN_SECURITY.USERMAINT_SELF.GBL" . . . >
```

When this HTML is downloaded to the browser, the absolute URL would include the scheme, server, and servlet directory of the proxying portal servlet, even though the iScript may have run on a content servlet on a different web server. The absolute URL continues with the portal, node, service type, and component name, as specified by the iScript.

Here is what the final URL looks like:

```
http://sun09/psp/ps/EMPLOYEE/PSFT_HR/c/MAINTAIN_SECURITY.USERMAINT.GBL
```

---

**Note.** The content services always specify the portal, node and content type (with the `"../.."`) even if those values are the same as the current page.

---

Now assume that you want the navigation iScript to create a link to the `MAINTAIN_SECURITY.USERMAINT_SELF.GBL` component in the `HRMS` node. Also assume that the component is being accessed by the employee portal. The navigation iScript would add the following HTML to the response:

```
<a href="../../../EMPLOYEE/HRMS/c/MAINTAIN_SECURITY.USERMAINT_SELF.GBL" . . . >
```

The absolute URL looks like this:

`http://sun09/ps/ps/EMPLOYEE/HRMS/c/MAINTAIN_SECURITY.USERMAINT.GBL`

The URL correctly points to the appropriate content without any HTML parsing or URL rewriting.

Finally, assume that you want a navigation iScript running within the EMPLOYEE portal to construct a link to the MAINTAIN\_SECURITY.USERMAINT\_SELF.GBL component within the e\_benefits portal. To construct this link, the iScript generates the following HTML:

```
<a href="../../E_BENEFITS/HRMS/c/MAINTAIN_SECURITY.USERMAINT_SELF.GBL" . . . >
```

---

**Note.** The HREF tag with a relative URL can only be used to change a portal or node if the HTML is being accessed through an HTML template. It won't work with a frame template, since the base URI of the frame points to the content servlet, which ignores the portal and node names. Use the PeopleCode transfer function to specify a target portal and node.

---

### See Also

*Enterprise PeopleTools 8.50 PeopleBook: PeopleCode Language Reference*

---

## Page-Based Template Proxying

This section discusses:

- Proxying and URL conversion.
- Anchor tag conversions.
- Form tag conversions.
- JavaScript conversions.

The following discussion only applies to content that is not in the simple URL format.

See [Chapter 3, "Understanding Portal Technology," Portal URL Formats, page 31.](#)

## Proxying and URL Conversion

When processing page-based templates, the portal servlet uses a process called *proxying* to help ensure that users always stay within the context of the portal and to ensure that familiar portal features, such as the universal navigation header, do not disappear when a user clicks a link.

When users sign in to a PeopleSoft portal, they sign in to a web server on which the portal servlet is running. The portal servlet processes all the HTML that isn't in the simple URL format, converting all URL references to point to the portal web server rather than the original URL. The original URL is still necessary to retrieve the requested content; it is stored in the new URL in the URL query string parameter. The portal servlet proxies all links and all form actions in this manner. Additionally, it converts any relative URLs into absolute URLs.

As an example, imagine that a user requests a page from an external web site through a proxied link in the portal. The request arrives at the portal web server, invoking the portal servlet. The portal servlet then programmatically retrieves the page from the web server associated with the requested page. It proxies all the links on the retrieved response and sends the page (the contents of the HTTP response) back to the browser, formatted as the user would expect within the portal.

### ***Converting Relative URLs to Absolute URLs***

The use of relative URLs is common in web page design for external content. They are often used when a web page includes links to content on the same web server that the page is on. This works fine when a browser communicates directly with a web server, because there is no ambiguity about where the relative URL points. However, because the portal servlet—and the proxying process—is placed between the browser and the target page, relative URLs become relative to the portal web server instead of the original target server. To prevent this from occurring and causing broken links, part of the proxying process includes the conversion of all relative URLs to absolute URLs. The following examples show the original, relative version of an HTML tag, and the rewritten absolute version created by the portal servlet.

The following example shows a relative tag:

```
<IMG src="/image/cache/image.gif" lowsrc="/image/cache/image2.gif">
```

The following example shows an absolute tag:

```
<IMG src="http://originalserver/image/cache/image.gif"
  lowsrc="http://originalserver/image/cache/image2.gif">
```

## **Anchor Tag Conversions**

The portal servlet rewrites all anchor tags so that their SRC attributes direct the browser's request to the portal web server instead of the server that created the URL. After rewriting the anchor tag, the portal servlet can determine if the target URL should be wrapped with a template.

### ***Old Anchor Tag***

The following example shows an old anchor tag:

```
<a src=http://server/targetpage.html?Action=New>
```

### ***New Anchor Tag***

The following example shows a new anchor tag:

```
<a src=http://portalserver/psp/ps/EMPLOYEE/NODE/e/?url=http%3a%2f%2fserver%2ftarget%3d
=>
=>
page.html%3fAction%3dNew>
```

## **Form Tag Conversions**

Similar to anchor tags, all Form tags must have their Action attributes rewritten. However, with Form tags, the original URL is captured in a hidden form field instead of a query string parameter.

***Old Form Tag***

The following example shows an old Form tag:

```
<form action=http://server/targetpage.html>
```

***New Form Tag***

The following example shows a new Form tag:

```
<form action=http://portalserver/ps/ps/EMPLOYEE/NODE/e>  
  <input type=hidden name=URL>  
    http://server/targetpage.html  
</input>
```

**JavaScript Conversions**

The portal servlet ensures that URL references contained in JavaScript are rewritten to point to the portal servlet instead of their original reference.

## **Part 2**

# **Using Navigation Pages**

## **Chapter 4**

### **Working With Navigation Pages**



## Chapter 4

# Working With Navigation Pages

This chapter provides overviews of navigation pages, navigation collections, their common elements, and discusses how to:

- Set up navigation collection options.
- Create and maintain navigation collections.
- Publish navigation collections.
- Run portal utility processes.

---

**Note.** Examples of PeopleSoft Pure Internet Architecture pages in this chapter were captured with the PeopleTools default style sheet applied. PeopleSoft Pure Internet Architecture pages may display differently in your database depending on the applied style sheet. However, regardless of the style sheet applied, all page elements displayed remain the same.

---

---

## Understanding Navigation Pages

Navigation pages provide a user-friendly navigation tool in the form of task-driven pages that provide intuitive access to pages needed to complete business processes. These navigation pages can be configured to incorporate the use of icons to further increase the ability to intuitively navigate tasks.

There are two types of navigation pages:

- Standard navigation pages.

These pages are dynamically generated based on folder and content reference sequence numbers defined in your portal registry, as well as settings defined by your portal administrator in the Navigation Collection System Options and Registry Options components.

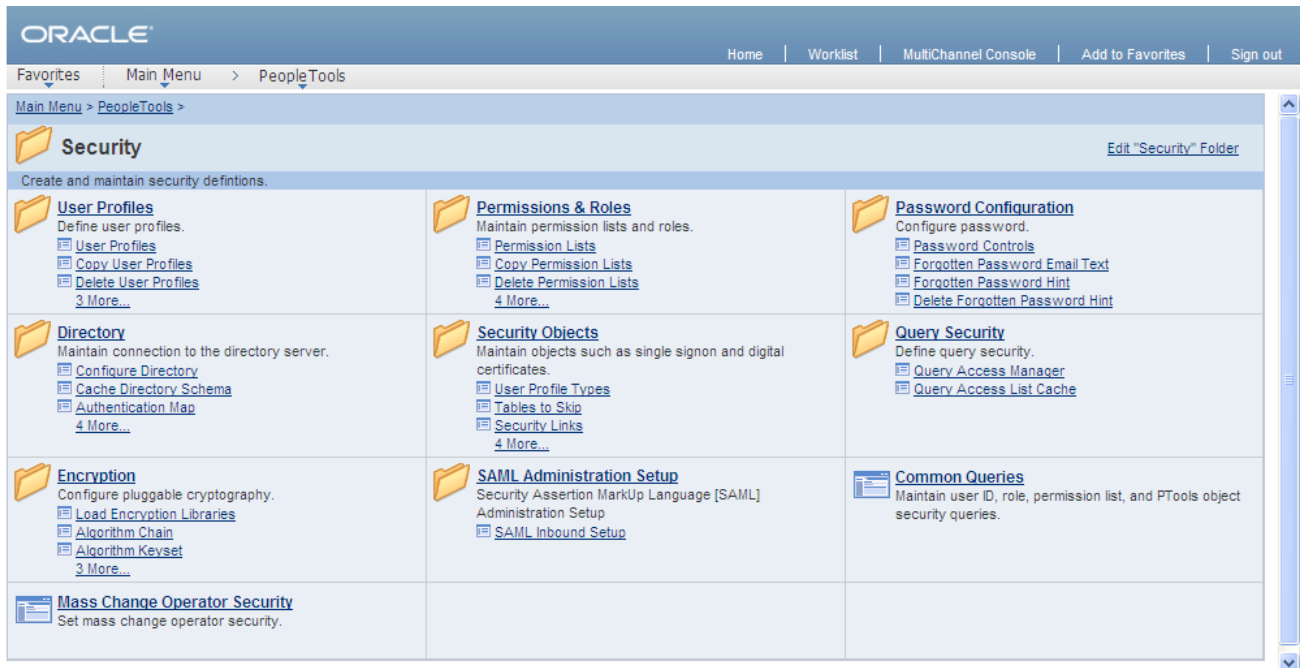
- Custom navigation pages.

These pages are delivered to address commonly used tasks and business processes in your PeopleSoft application. Authorized users can edit delivered custom navigation pages, as well as create new ones tailored to suit the business process needs of your organization.

## Standard Navigation Pages

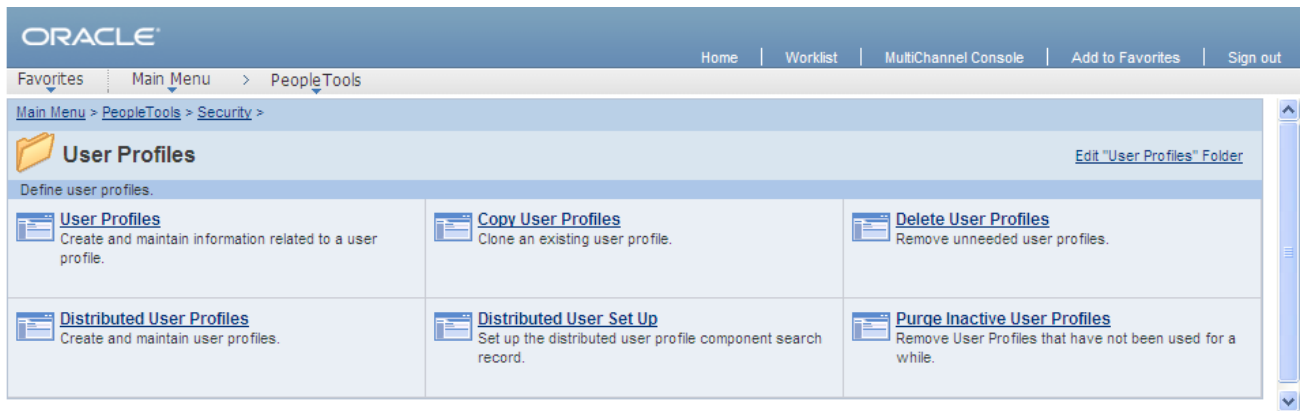
Standard navigation pages are available for every folder that appears in your menu navigation that has also been configured to display a standard navigation page. You enable the display of standard navigation pages at the registry level on the General Settings page. You can then disable the display of standard navigation pages for specific folders in the registry on the Folder Administration page.

You access standard navigation pages by clicking a folder label link in the menu navigation. Elements embedded in the folder, including content references and other folders, are arranged in a graphical format on a standard navigation page displayed to the right of the menu navigation. For example, clicking the Security folder in the menu navigation displays the Security standard navigation page as shown here:



Security standard navigation page

When you click a folder link on a standard navigation page, you access another standard navigation page that reflects the contents of the selected folder. For example, when you click the User Profiles folder, the User Profiles standard navigation page displays as shown here:



User Profiles standard navigation page

From a navigation page, you can click a content reference link to directly access the component to which the content reference points without drilling down through the folder hierarchy.

See [Chapter 4, "Working With Navigation Pages," Setting Up Navigation Collection Options, page 52.](#)

### Standard Navigation Page Display

To enable the display of standard navigation pages:

1. Select PeopleTools, Portal, General Settings.
2. Select the Is Folder Navigation Enabled check box.
3. Enter `PT_PTPP_SCFNAV_BASEPAGE_SCR` (the delivered Base Navigation Page content reference) in the Folder Navigation Object Name field.

This Base Navigation Page content reference is designed to display standard navigation pages as they are documented in this chapter.

---

**Note.** To enable the display of standard navigation pages for the registry in a portal other than the one you are signed into, select PeopleTools, Portal, Portal Definitions, then click Edit next to the portal you want to change to access the Portal Definitions - Edit Portal page.

---

**Note.** You can disable the display of a standard navigation page for an individual folder in the selected registry by selecting the Is Folder Navigation Disabled option on the Folder Administration page.

You can designate a content reference for an individual folder that overrides the content reference defined at the registry level by selecting a value in the Folder Navigation Object Name field on the Folder Administration page. The content reference you designate as the folder navigation object is launched when the folder label is selected.

---

See [Chapter 5, "Administering Portals," Administering Folders, page 105](#) and [Chapter 5, "Administering Portals," Administering Portal Definitions, page 101.](#)

### Folder and Link Sequence on Standard Navigation Pages

The sequence in which folders and links appear on standard navigation pages is defined in the Structure and Content component.

Although the contents of a standard navigation page correspond to that of the menu navigation folder selected to display it, the sequence of elements on the page may not be the same as the sequence in which they are displayed in the menu navigation. As dictated by menu navigation functionality, folder elements are displayed before content reference links. The order of elements within these groupings is determined by the sequence numbers listed in the Structure and Content component.

Unlike the grouped elements in the menu navigation folder, the folder and link sequence on standard navigation pages is determined strictly by sequence numbers in the Structure and Content component, and not on the type of element being displayed. However, if a folder element and content reference link have the same sequence number, the folder is displayed before the content reference on the standard navigation page.

See [Chapter 5, "Administering Portals," Administering Folders, page 105](#) and [Chapter 5, "Administering Portals," Administering Content References, page 109](#).

## Custom Navigation Pages

While standard navigation pages automatically display only the contents of a selected menu folder, custom navigation pages can be designed to display any content references and folders available in the menu navigation, regardless of their location.

You access custom navigation pages by way of links in the menu navigation. When you click a menu navigation link to a custom navigation page, the page displays to the right of the menu navigation. For example, selecting the Context Manager Home link in the menu navigation displays the Context manager Home custom navigation page as shown here:



Context Manager Home custom navigation page

When you click a folder link on a custom navigation page, you access another navigation page that reflects the contents of the selected folder. When you click a content reference link on a custom navigation page, you access the actual page to which the content reference points.

See [Chapter 4, "Working With Navigation Pages," Creating or Editing a Navigation Collection, page 64](#).

## Folder and Link Sequence on Custom Navigation Pages

The sequence in which folders and links appear on custom navigation pages is defined by the Placement Properties set in the Navigation Collection definition for the custom navigation page.

Once the folder and link sequences for a custom navigation page are registered, their sequence numbers are propagated to the registry and are available in the Structure and Content component. However, any modifications to custom navigation pages *must* be performed in the Navigation Collection definition. Do not modify custom navigation pages using the Structure and Content component.

---

**Note.** When you save the Navigation Collection, these modifications to the Placement Properties sequences will be updated to the registry.

The Structure and Content component does not display folder reference links.

---

See [Chapter 4, "Working With Navigation Pages," Editing a Folder, page 73](#) and [Chapter 4, "Working With Navigation Pages," Editing a Link, page 78](#).

## Navigation Page Display Caching

You can enable the caching of displayed navigation pages. Enabling caching improves performance because the next time the same navigation page is accessed from any computer using the same web server with the same cache designation (role, user, or public), the navigation page is retrieved from the web server cache memory. To clear the web server cache memory, stop and restart the web server or wait until the cache time elapses.

Methods for enabling caching depend on method that you use to access navigation pages:

- To enable caching for a custom navigation page accessed by clicking a content reference link, select the Enable Caching option on the Publish Collection page when you publish the custom navigation page.

On publication, the Cache Enabled Value field value you entered on the System Options page is added to the published custom navigation page registry structure definition.

- To enable caching for a folder or breadcrumb on a navigation page, select the Enable Drill Down Cache option on the System Options page.

Enter a value in the Cache Enabled Value field (*role,max-age=30*, for example)

- To enable caching for a standard navigation page accessed by clicking a menu navigation folder, edit the Base Navigation Page content reference (PT\_PTPP\_SCFNAV\_BASEPAGE\_SCR) on the Content Ref Administration page.

The Base Navigation Page content reference is located in the Tools - Hidden folder. In the Content Reference Attributes group box, enter the *PSCACHECONTROL* attribute name and an attribute value, such as *role,max-age=30*.

See [Chapter 4, "Working With Navigation Pages," Setting Up Navigation Collection Options, page 52](#); [Chapter 4, "Working With Navigation Pages," Publishing Navigation Collections, page 80](#) and [Chapter 5, "Administering Portals," Administering Content References, page 109](#).

---

## Understanding Navigation Collections

This section discusses:

- Navigation collections
- Navigation collection images

### Navigation Collections

Navigation Collections provide you with a flexible tool for building alternative taxonomies of the contents stored in your portal registry. These alternative taxonomies, or groupings of links to portal content, can then be deployed to different users or groups of users, creating navigation that specifically addresses your users' business needs.

The links in Navigation Collections provide an alternate method of accessing content in the portal registry. A Navigation Collection is a published compilation of these links. A link in a Navigation Collection is a pointer to a content reference or folder reference in the portal registry; however, it does not store the contents of the content reference or folder reference. Navigation Collection links store only the references needed to construct a URL at runtime to redirect output to the original content reference or folder reference. These references include the portal link, folder reference link, or content reference link.

Once created, a Navigation Collection can be published as a navigation page or navigation pagelet for use on a user homepage.

When you define and save a Navigation Collection, the system automatically stores it in the hidden Navigation Collections folder under Portal Objects in the portal registry.

You can choose to publish Navigation Collections as public content. If you do not choose to provide public access to a Navigation Collection, you can run the Sync Collection Security (synchronize collection security) process to apply content reference and folder security defined in the portal registry to the published navigation page or pagelet content reference. You can also allow manual maintenance to the security access to published pages and pagelets by selecting the *Do Not Allow Security Sync* option for the publication on the Publish Collection page.

---

**Note.** Public access is allowed by default.

---

Navigation Collections allow you to distribute the creation and maintenance of collections to subject matter experts in your organization, such as functional administrators and business analysts. The subject matter experts creating Navigation Collections should understand the content their user communities need to access on a frequent basis. They should also know where to find links to this content, whether the content is accessed by portal navigation or external URLs.

Navigation Collections can contain these elements:

- Links to existing non-hidden content references.
- Links to existing non-hidden folders in the menu.

The Navigation Collection will contain all contents of the folder and its subfolders.

- Custom user-defined folders that can be used to group combinations of the elements listed above.

## Navigation Collection Published as a Navigation Page

The following example illustrates the appearance of a Navigation Collection published as a navigation page:



Context Manager Home navigation page

Top-level folders in the Navigation Collection may be transformed into bold and linked section headings that are used to organize the folders and content references they contain, as shown in these examples. However, the appearance of your published Navigation Collections is dependent on its assigned style sheet. The assigned style sheet must use style classes contained in the PTPP\_SCSTYLEDEF style class definition. Selecting the top-level folder link accesses a page containing any folders and content references nested in the parent folder.

---

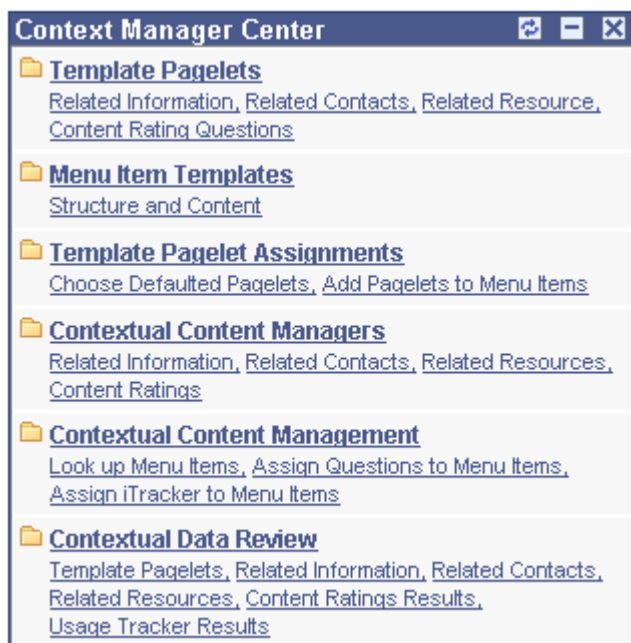
**Note.** Select the Edit <Navigation Collection name> Collection link on the navigation page to access the Navigation Collection definition on the Maintain Collection page. This link displays only for users who can access the Navigation Collection definition component.

---

See [Chapter 4, "Working With Navigation Pages," Creating and Maintaining Navigation Collections, page 61.](#)

## Navigation Collection Published as a Navigation Pagelet

The following example illustrates the appearance of a Navigation Collection published as a navigation pagelet:



Content Home navigation pagelet

Folders and content references work in much the same way as they do for the published navigation page.

### **See Also**

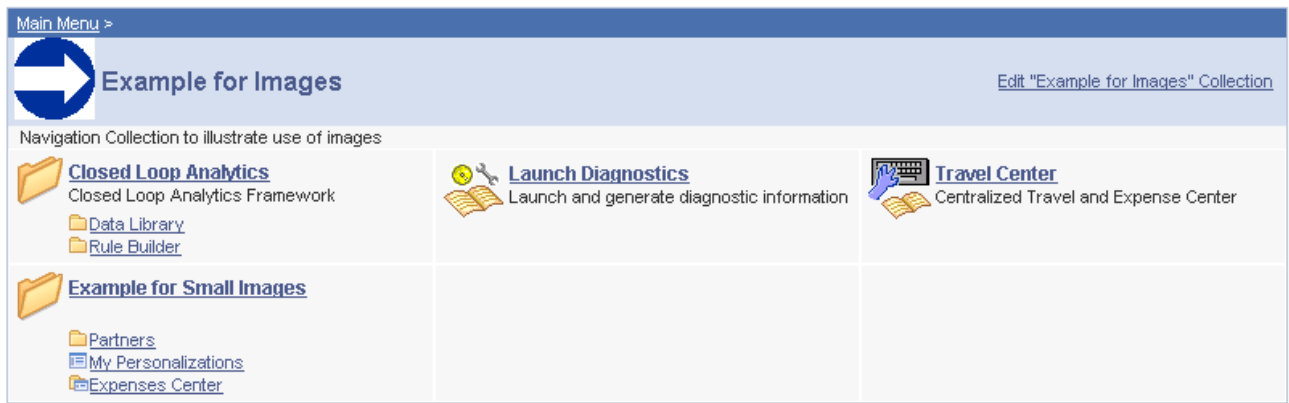
[Chapter 4, "Working With Navigation Pages," Publishing Navigation Collections, page 80](#)

[Chapter 4, "Working With Navigation Pages," Understanding Portal Utility Processes, page 88](#)

## **Navigation Collection Images**

The images you define for your Navigation Collections are published and displayed on Navigation Collection pages. You can associate images with Navigation Collections themselves, as well as with Navigation Collection folders and content references. You can define these images at the system, registry, source, and individual Navigation Collection levels.

The following example Navigation Collection page publication illustrates the different ways in which images can display:



### Example for Images page

In the example Navigation Collection, suppose that the following images have been defined for the different image types:



This image was defined as the collection image for this specific Navigation Collection.



This image was defined as the default large folder image on the System Options page, and was not overridden at the source reference, registry, or Navigation Collection level. Large folder images display for folders at the top level of a Navigation Collection publication.



This image was defined as the content image in the Navigation Collection definition. Large content images display for content references at the top level of a Navigation Collection publication.



This image represents a Navigation Collection that was added to this Navigation Collection. This image was selected as the collection image in the referenced Navigation Collection definition, and was not overridden in this Navigation Collection definition. Large collection images display for Navigation Collections at the top level of a Navigation Collection publication.



This image was defined as the small folder image. Small folder images display for folders nested within top-level folders on a Navigation Collection publication.



This image was defined as the small content image. Small content images display for content references nested within top-level folders of a Navigation Collection publication.



This image was defined as the small collection image. Small collection images display for Navigation Collections nested within top-level folders of a Navigation Collection publication.

While large (parent) images can be set at the system or registry default levels, and overridden with unique images defined at the source reference or Navigation Collection levels, small (child) images can be defined only at the system and registry default levels. This means that the small images displayed on Navigation Collection publications will always be the default small image defined at the system or registry level.

If the Navigation Collection content, folder, or Navigation Collection reference for which a small image is displayed has a unique image defined at the source reference or Navigation Collection level, this unique large image does not display until the element is accessed as a top- or parent-level element on a Navigation Collection page.

For example, while the Expenses Center is displayed with its default small image, when you access the Example for Small Images navigation page, the Expenses Center displays with its unique large image.

## Common Elements Used on Navigation Pages

### Breadcrumbs

Your portal administrator may configure your navigation pages to display breadcrumbs. These breadcrumbs display across the top of a navigation page and are generated according to the navigation path used to access the current content displayed. You can select the breadcrumbs to navigate through the path you used to access the current content.

See [Chapter 4, "Working With Navigation Pages," Setting Up Navigation Collection Options, page 52.](#)

### Edit <folder name> Folder

The Edit <folder name> Folder link displays if you have accessed a standard navigation page and you have permissions defined for your user ID or role that enable you to access the Structure and Content component.

Select the Edit <folder name> Folder link to access the Folder Administration page, where you can edit aspects of the folder that was used to generate the selected standard navigation page.

See [Chapter 5, "Administering Portals," Defining Folders, page 105.](#)

### Edit <Navigation Collection name> Collection

The Edit <Navigation Collection name> Collection link displays if you have accessed a custom navigation page and you have permissions defined for your user ID or role that enable you to edit Navigation Collections.

Select the Edit <Navigation Collection name> Collection link to access the Navigation Collections component, where you can edit the definition that was used to build the selected custom navigation page.

See [Chapter 4, "Working With Navigation Pages," Creating and Maintaining Navigation Collections, page 61.](#)



Click the spell-check button to check the spelling of text in the associated field.

See *Enterprise PeopleTools 8.50 PeopleBook: Using PeopleSoft Applications*, "Working With Pages," Using Spell Check.

---

## Setting Up Navigation Collection Options

This section provides an overview of navigation collection options and discusses how to:

- Define Portal Utilities system options.

- Define Portal Utilities registry options.

---

**Note.** These options are used by Navigation Collections and Pagelet Wizard.

---

## Understanding Navigation Collection Options

You define and override navigation collection options at different levels. Options defined at the system level provide default values, with options defined at the registry, source, and navigation collection levels being capable of overriding each previous level.

PeopleSoft delivers a set of system-level defaults. You can change these values on the System Options page or override these values at successive levels in the default hierarchy. This table lists the navigation collection options and the levels at which they are set up or overridden:

<b>Options</b>	<b>System (default)</b>	<b>Registry (overrides system default)</b>	<b>Source Reference (overrides system and registry values)</b>	<b>Collection Reference (overrides system, registry, and source reference values)</b>
Registry object prefix	X			
Show breadcrumbs	X	X		
Show main menu breadcrumb	X	X		
Registry node name	X			
Large (parent) image	X	X	X (with attribute name PTPP_IMAGE)	X
Small (child) image	X	X		
Style sheet	X	X		X
Show images	X	X		X
Maximum child links	X	X		X
Maximum columns	X	X		
Owner ID	X			X

Navigation collection options are defined as follows:

- System options on the System Options page.
- Registry options on the Registry Options page.

- Source reference options are set up using the Structure and Content component.
- Navigation Collection reference options on the Maintain Collections page.

The default option to display standard navigation pages when you click a menu folder is set at the registry level on the General Setting page. You can override this default on the source reference.

### See Also

[Chapter 4, "Working With Navigation Pages," Defining Portal Utilities System Options, page 56](#)

[Chapter 4, "Working With Navigation Pages," Defining Portal Utilities Registry Options, page 59](#)

[Chapter 4, "Working With Navigation Pages," Creating and Maintaining Navigation Collections, page 61](#)

[Chapter 5, "Administering Portals," page 99](#)

[Chapter 5, "Administering Portals," Managing General Portal Settings, page 135](#)

## Common Elements Used in This Section

### Show Breadcrumbs

Select to indicate that you want breadcrumbs to display on published Navigation pages.

These breadcrumbs reflect the path taken by users as they navigate through left-navigation menu folders and navigation page folders. Displaying breadcrumbs facilitates user navigation, as the left-navigation menu is not updated to reflect the user's navigation through Navigation pages.

The first breadcrumb displayed is the Main Menu collection name link that points to the delivered PT\_PTPP\_PORTAL\_ROOT Navigation Collection, which is based on the PORTAL\_ROOT\_OBJECT folder. Alternatively, you can select a different Navigation Collection to be used as the first (main menu) breadcrumb using the Main Menu Collection Name field, or you can choose not to display a first breadcrumb.

The next set of breadcrumbs display dependent on the left-navigation menu folders the user opened to access the Navigation Collection page.

Subsequent breadcrumbs display dependent on the folder path the user accesses on the Navigation Collection page.

### Show Main Menu Breadcrumb

If you have selected the Show Breadcrumbs option, select this option to include the Main Menu collection name as the first entry in the breadcrumb path displayed on the Navigation Collection page publication.

If you clear this option, the first entry in the breadcrumb path is dependent on the level-1 left-navigation folder that the user opens to access the Navigation Collection page. The Main Menu link does not display as the first link in the breadcrumbs.

<b>Main Menu Collection Name</b>	<p>This field displays when you have the Show Main Menu Breadcrumb option selected. The default value is set to <i>Main Menu</i>, which designates that the first breadcrumb displayed accesses the Main Menu page. You may choose to replace the first breadcrumb Main Menu link with a link that accesses an alternate Navigation Collection you have created.</p> <p>The first breadcrumb Main Menu link to the Main Menu page provides access to the entire navigation for your database, filtered by security. You may want to provide a first breadcrumb link that provides access to a Navigation Collection that you have configured to contain only selected folders and content references.</p>
<b>Style Sheet Name</b>	<p>Select the cascading style sheet (CSS) that you want to use to generate Navigation Collection pages. Cascading style sheets are defined in PeopleSoft Application Designer.</p> <p>This setting applies to both navigation pages and navigation pagelets.</p> <hr/> <p><b>Note.</b> You must use the same styleclass names found on the stylesheet PTPP_STYLEDEFN.</p> <hr/> <p>See <i>Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Application Designer Developer's Guide</i>, "Creating Style Sheet Definitions."</p>
<b>Maximum Columns</b>	<p>Enter the maximum number of columns to display on a navigation page.</p> <p>Navigation pagelets display one or two columns depending on the user's settings on the Personalize Layout page. Narrow pagelets display contents in one column. Wide pagelets display contents in three columns.</p> <p>This setting applies to only navigation pages.</p> <p>See <i>Enterprise PeopleTools 8.50 PeopleBook: Using PeopleSoft Applications</i>, "Setting User Preferences," Personalizing Your Homepage.</p>
<b>Maximum Child Links</b>	<p>Enter the maximum number of links to display beneath each folder on the Navigation Collection publication. If the number of links in the folder exceeds this number, a X More... link appears beneath the folder that jumps to a page from which you can access all links stored in the Navigation Collection folder.</p> <p>For example, if you set Maximum Child Links to 4 and the Navigation Collection folder contains five links, three links display on the Navigation Collection page along with the X More... link.</p> <p>When you select the X More... link, a page containing all five links in the Navigation Collection folder displays.</p> <p>This setting applies to both navigation pages and navigation pagelets.</p>

## Pages Used to Define Portal Utilities Options

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
System Options	PTPP_OPTIONS	PeopleTools, Portal, Portal Utilities, System Options	Define system-level default values for Navigation Collection options. It may be possible to override these defaults at the registry, source reference, and Navigation Collection reference levels.
Registry Options	PTPP_SITE_OPT	PeopleTools, Portal, Portal Utilities, Registry Options	Define registry-level values for Navigation Collection options. These values may override system-level defaults, as well as be overridden at the source reference and Navigation Collection reference levels.

## Defining Portal Utilities System Options

Access the System Options page (PeopleTools, Portal, Portal Utilities, System Options).

## System Options


Select the system options and defaults. Most of the values can be overridden by the registry options, and by the defined collection.

### Portal Registry Structures

These values are used on the registry structure objects that are generated when a collection is created. The prefix is used on the name of the object. The node is used on the content reference.

\*Registry Object Prefix:


Node Name:


Owner ID:  

### Navigation Page

☒ Show Breadcrumbs

☒ Show Main Menu Breadcrumb

\*Main Menu Collection Name:  

\*Style Sheet Name:  


\*Maximum Columns:



\*Maximum Child Links:



☒ Enable Drill Down Cache



\*Cache Enabled Value:



### Navigation Image Defaults



\*Show Images:  



\*Large Folder Image:   

\*Large Content Image:   

\*Large Collection Image:   

\*Small Folder Image:   

\*Small Content Image:   

\*Small Collection Image:   

### Pagelet Wizard

\*Package Name:

\*Path:

\*Application Class ID:

[Return to Portal Utilities](#)

## System Options page

PeopleSoft delivers a set of system-level defaults. You can change the default values on this page or override the defaults at successive levels in the default hierarchy. However, you cannot delete the default values because system-level default values are required.

See [Chapter 4, "Working With Navigation Pages," Understanding Navigation Collection Options, page 53.](#)

## Portal Registry Structures

**Registry Object Prefix** Enter the prefix for the generated registry structure object names (content and folder references) that are created by Navigation Collection definitions, as well as for the published Navigation Collection pages and pagelets.

For example, the object name of a published navigation pagelet has this format: `<prefix>_SC_PGT_<Navigation Collection label>`. The object name of a published navigation page has this format: `<prefix>_SC_SP_<Navigation Collection label>`.

**Node Name** You should not change the default value delivered with your system. This default value corresponds to the content provider node for your database. For example, the following databases are delivered with the following default node names:

PeopleSoft Enterprise Financials/SCM: *ERP*

PeopleSoft Enterprise CRM: *CRM*

PeopleSoft Enterprise HCM: *HRMS*

PeopleSoft Enterprise Portal: *Local Node*

**Owner ID** Use this value to easily identify Navigation Collections created by your organization when searching for Navigation Collections through PeopleSoft Application Designer or SQL queries on the portal registry table.

Do not select a delivered PeopleSoft owner ID. You can define unique owner IDs for your organization by entering field translate values for the OBJECTOWNERID field using PeopleSoft Application Designer .

## Navigation Page

**Enable Drill Down Cache** Select to enable caching for navigation pages accessed by selecting a folder on a navigation page. You must also enter a value in the Cache Enabled Value field on this page.

See [Chapter 4, "Working With Navigation Pages," Navigation Page Display Caching, page 47.](#)

**Cache Enabled Value** Enter the value to assign to the PSCACHECONTROL attribute in the registry structure definition when enabling caching for applicable navigation page access methods. For example, you can enter a value such as *role,max-age=30*.

## Navigation Image Defaults

<b>Show Images</b>	<p>Select one of these values:</p> <p><i>Always Show Images:</i> Select if you always want images to display on Navigation pages and pagelets.</p> <p><i>Never Show Images:</i> Select if you never want images to display on Navigation pages and pagelets.</p> <p><i>Only Show Defined Images:</i> Select if you only want specifically defined images to display on Navigation pages and pagelets. If you select this option, only images defined as part of the Navigation Collection or defined on the source reference are displayed on the Navigation Collection's publications. System and registry level default images are not displayed.</p>
<b>Large Folder Image</b>	Select the default image that you want to display for top-level folders on a Navigation page or pagelet.
<b>Large Content Image</b>	Select the default image that you want to display for top-level content references on a Navigation page or pagelet.
<b>Large Collection Image</b>	Select the default image that you want to display for top-level Navigation Collection references on a Navigation page or pagelet.
<b>Small Folder Image</b>	Select the default image that you want to display for a folder within a folder on a Navigation page or pagelet.
<b>Small Content Image</b>	Select the default image that you want to display for a content reference within a folder on a Navigation page or pagelet.
<b>Small Collection Image</b>	Select the default image that you want to display for a Navigation Collection within a folder on a Navigation page or pagelet.

## Pagelet Wizard

<b>Package Name</b>	Specify the application package name to use for the display format.
<b>Path</b>	Enter the application class path to use for the display format.
<b>Application Class ID</b>	Enter the application class ID to use for the display format. The class must exist in the application package name that you specify.

## Defining Portal Utilities Registry Options

Access the Registry Options page (PeopleTools, Portal, Portal Utilities, Registry Options).

## Registry Options

Select the registry specific options and defaults. These values will override the defined system options. Most of the values can be overridden by a defined collection. When adding the registry options, the values are initially defaulted from the system options. Click 'Delete Registry Override' to revert to the system defaults for the selected registry.

Registry Name:

EMPLOYEE

Employee-facing registry content

Navigation Page

☒ Show Breadcrumbs

☒ Show Main Menu Breadcrumb

\*Main Menu Collection Name:

Main Menu

\*Style Sheet Name:

PSSTYLEDEF\_SWAN

\*Maximum Columns:

3

\*Maximum Child Links:

4

Navigation Image Defaults

\*Show Images:

Always Show Images

\*Large Folder Image:

PTPP\_FN\_LARGE\_FOLDER\_ICN

\*Large Content Image:

PTPP\_FN\_LARGE\_CONTENT\_ICN

\*Large Collection Image:

PTPP\_FN\_LARGE\_COLLECTION\_ICN

\*Small Folder Image:

PTPP\_FN\_SMALL\_FOLDER\_ICN

\*Small Content Image:

PTPP\_FN\_SMALL\_CONTENT\_ICN

\*Small Collection Image:

PTPP\_FN\_SMALL\_COLLECTION\_ICN

Delete registry Override

Registry Options page

All options you set on this page override defaults defined at the system level on the System Options page. See [Chapter 4, "Working With Navigation Pages," Understanding Navigation Collection Options, page 53.](#)

<b>Registry Name</b>	Select the portal registry for which you want to define Navigation Collection options that will override system-level defaults.
----------------------	---

## Navigation Image Defaults

<b>Show Images</b>	<p><i>Always Show Images.</i> Select if you always want images to display on Navigation pages and pagelets accessed using the selected portal registry.</p> <p><i>Never Show Images.</i> Select if you never want images to display on Navigation pages and pagelets accessed using the selected portal registry.</p> <p><i>Only Show Defined Images.</i> Select if you only want specifically defined images to display on Navigation pages and pagelets accessed using the selected portal registry. If you select this option, only images defined as a part of the Navigation Collection or defined on the source reference are displayed on the Navigation pages and pagelets. System and registry level default images are not displayed.</p>
<b>Large Folder Image</b>	Select the image that you want to display for top-level folders on a Navigation page or pagelet accessed using the selected portal registry.
<b>Large Content Image</b>	Select the image that you want to display for top-level content references on a Navigation page or pagelet accessed using the selected portal registry.
<b>Large Collection Image</b>	Select the image that you want to display for top-level Navigation Collection references on a Navigation page or pagelet accessed using the selected portal registry.
<b>Small Folder Image</b>	Select the image that you want to display for a folder within a folder on a Navigation page or pagelet accessed using the selected portal registry.
<b>Small Content Image</b>	Select the image that you want to display for a content reference within a folder on a Navigation page or pagelet accessed using the selected portal registry.
<b>Small Collection Image</b>	Select the image that you want to display for a Navigation Collection within a folder on a Navigation page or pagelet accessed using the selected portal registry.
<b>Delete Registry Override</b>	Click to revert the portal registry's Navigation Collection options to the system-level defaults defined on the System Options page.

---

## Creating and Maintaining Navigation Collections

This section discusses how to:

- Search for a Navigation Collection.
- Create or edit a Navigation Collection.
- Add a folder.
- Select a source folder.
- Select Navigation Collection images.
- Edit a folder.

- Add a link.
- Select a source link.
- Edit a link.

## Pages Used to Create and Maintain Navigation Collections

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Find an Existing Collection	PTPP_SCSRCH	<ul style="list-style-type: none"> <li>• PeopleTools, Portal, Portal Utilities, Navigation Collections</li> <li>• Click the Return to Search link on the Maintain Collections page.</li> </ul>	Search for an existing Navigation Collection that you want to edit or delete. Access pages you can use to create a new Navigation Collection.
Maintain Collection	PTPP_SCMAINTCOLL	<ul style="list-style-type: none"> <li>• Select the Edit link for an existing Navigation Collection on the Find an Existing Collection page.</li> <li>• Select the Add Collection link on the Find an Existing Collection page.</li> <li>• Select the Edit &lt;Navigation Collection name&gt; Collection link on a custom navigation page.</li> </ul>	Create and maintain Navigation Collections.
Add Folder	PTPP_SCAE_FOLDER	Click the Add Folder button on the Maintain Collection page.	Add a Navigation Collection folder reference.
Select Source Folder	PTPP_SCBROWSEREG	<ul style="list-style-type: none"> <li>• Click the Browse Portal Registry icon on the Add Folder page.</li> <li>• Click the Browse Portal Registry button on the Edit Folder page.</li> </ul>	View a graphical display of a selected portal registry and select the content reference that you want to add to your Navigation Collection.

<b>Page Name</b>	<b>Definition Name</b>	<b>Navigation</b>	<b>Usage</b>
Browse Image Catalog	PTPP_SCIMGSRCH	<ul style="list-style-type: none"> <li>Click the Browse Image Catalog button on the Add Folder page.</li> <li>Click the Browse Image Catalog button on the Edit Folder page.</li> <li>Click the Browse Image Catalog button on the Add Link page.</li> <li>Click the Browse Image Catalog button on the Edit Link page.</li> </ul>	Search for and select an image to associate with the selected folder or content reference. This image displays on the published navigation page or pagelet.
Edit Folder	PTPP_SCAE_FOLDER	Click the Edit Folder button on the Maintain Collection page.	Edit an existing Navigation Collection folder reference.
Add Link	PTPP_SCAE_SHORTCUT	Click the Add Link button on the Maintain Collection page.	Add a content reference link.
Select Source Link	PTPP_SCBROWSEREG	<ul style="list-style-type: none"> <li>Click the Browse Portal Registry button on the Add Link page.</li> <li>Click the Browse Portal Registry button on the Edit Link page.</li> <li>Click the Move button on the Publish Collections page.</li> </ul>	<p>View a graphical display of a portal registry.</p> <p>When accessed from the Add Link page and Edit Link page, you select a portal registry and select the folder that you want to reference in your Navigation Collection.</p> <p>When accessed from the Publish Collections page, you view the portal registry defined for the published navigation page and select the folder in which you want the navigation page to reside.</p>
Edit Link	PTPP_SCAE_SHORTCUT	Click the Edit Link button on the Maintain Collection page.	Edit an existing content reference link.

## Searching for a Navigation Collection

Access the Find an Existing Collection page (PeopleTools, Portal, Portal Utilities, Navigation Collections).

Navigation Collections

Find an Existing Collection

Search By:



Name

begins with

Search

[Add Collection](#)

Search Results

Customize | Find |  |  | First 1-2 of 2 Last

Main Menu	<a href="#">Edit</a>	
XML Publisher	<a href="#">Edit</a>	<a href="#">Delete</a>

Find an Existing Collection page

<b>Search by</b>	<p>Select one of these values:</p> <p><i>Description.</i> Select to conduct your search based on Navigation Collection description text.</p> <p><i>Name.</i> Select to conduct your search based on Navigation Collection name text.</p> <p>Use the <i>begins with</i> and <i>contains</i> options to limit your search results.</p>
<b>Add Collection</b>	<p>Select to access the Maintain Collection page, where you can define a new Navigation Collection.</p>
<b>Search Results</b>	
<b>Edit</b>	<p>Select to access the Maintain Collection page, where you can edit an existing Navigation Collection.</p>
<b>Delete</b>	<p>Click to delete the existing Navigation Collection. You are prompted with the Navigation Collections - Delete Confirmation page, where you can confirm or cancel the deletion of the Navigation Collection.</p> <hr/> <p><b>Note.</b> The Delete button does not display for the Main Menu Navigation Collection. While you can edit this Navigation Collection, you should not delete it as the Main Menu Navigation Collection is used by the system as a default page to display when it encounters certain errors. Displaying the Main Menu rather than an error page enables you to continue to navigate through your system.</p> <hr/>

Creating or Editing a Navigation Collection

Access the Maintain Collection page (click the Edit link for an existing Navigation Collection on the Find an Existing Collection page).

Maintain Collection

Publish Collection

## Navigation Collections

Define the Navigation Collection. The main folder of the Navigation Collection tree is determined from the Navigation Collection name. Add additional folders or links to the Navigation Collection by clicking on a tree node, and then clicking on one of the displayed action buttons.

Collection Properties

\*Name:

XML Publisher

Description:

(254 Characters)

\*Valid from date:

01/01/1900

Valid to date:

Owner ID:

PeopleTools

Override Default Options

XML Publisher

Add Link

Add Folder

Register Data Sources

Define XMLP Reports

Run XMLP Reports

Complete Setup

[Return to Search](#)

Navigation Collections page

**Note.** If you modify a Navigation Collection that has already been published, you do not need to republish the collection using the Publish Collection page. Saving your edits on the Maintain Collection page is sufficient because Navigation Collection publications are dynamically generated for display based on their saved Navigation Collection definitions.

### Collection Properties

<b>Name</b>	Enter a name for the Navigation Collection. This text displays as the label on the published navigation page or pagelet.
<b>Description</b>	<p>Enter a description of the Navigation Collection. This text can display in several locations.</p> <p>If the Navigation Collection is published as a navigation page, the description text displays as hover text for the menu item link used to access the page. The description text also displays on the navigation page.</p> <p>If the Navigation Collection is published as a navigation pagelet, the description text displays on the navigation pagelet.</p>

<b>Valid from date/Valid to date</b>	<p>Enter a range of dates during which this Navigation Collection is available. This information is copied to the portal registry when you register the Navigation Collection using the options on the Publish Collection page.</p> <p>It is recommend that you leave the Valid to date field clear, if you anticipate that the Navigation Collection should remain active indefinitely.</p>
<b>Owner ID</b>	<p>Use this value to identify Navigation Collections created by your organization when searching for Navigation Collections through PeopleSoft Application Designer or SQL queries on the portal registry table.</p> <p>Do not select a delivered PeopleSoft owner ID. You can define unique owner IDs for your organization by entering field translate values for the OBJECTOWNERID field using PeopleSoft Application Designer .</p> <p>If a default value has been set at the system or registry level, it displays, but can be overridden.</p> <p>See <a href="#">Chapter 4, "Working With Navigation Pages," Understanding Navigation Collection Options, page 53.</a></p>

### **Override Default Options**

The Override Default Options group box enables you to enter values that override any Navigation Collection defaults defined at the system, registry, and source reference levels and which apply to only the Navigation Collection you are creating. Leave fields clear if you don't want to override any default values.

See [Chapter 4, "Working With Navigation Pages," Understanding Navigation Collection Options, page 53.](#)

<b>Style Sheet Name</b>	Choose the stylesheet for this Navigation Collection.
<b>Maximum Child Links</b>	<p>Enter the maximum number of child links to display.</p> <hr/> <p><b>Note.</b> If more links are available for this collection, a More link will display.</p> <hr/>
<b>Show Images</b>	<p>Select one of these values:</p> <p><i>Always Show Images:</i> Select to always displays images on pages and pagelets for this Navigation Collection.</p> <p><i>Never Show Images:</i> Select if you don't want images to display on pages and pagelets for this Navigation Collection.</p> <p><i>Only Show Defined Images:</i> Select to display only specifically defined images on pages and pagelets for this Navigation Collection. If you select this option, only images defined as a part of this Navigation Collection or the source reference are displayed on the pages and pagelets for this Navigation Collection. System and registry level default images are not displayed.</p> <hr/> <p><b>Note.</b> If you select the <i>Only Show Defined Images</i> option, small images do not display on your Navigation Collection pages or pagelets as small images can only be defined as system and registry level defaults.</p> <hr/>

**Collection Image** Select the large image you want to display as the top level image for this Navigation Collection.

The Navigation Collection you define is displayed as a tree hierarchy of nodes at the bottom of the page.

The Name value that you entered for the Navigation Collection displays as a root node link at the top of the hierarchy. Each node below the Navigation Collection name is either a folder or content reference link you defined using the following options and their associated pages.

A Navigation Collection may be composed of a series of individual content reference links, a series of folders containing content reference links, or a combination of individual content reference links and folders.

**Add Link** Click to access the Add Link page. This option displays when you select the Navigation Collection root node folder or user-defined folders in the Navigation Collection hierarchy.

If you use this option at the root node level of the hierarchy, the content reference link you create appears directly off the root node, at the same level in the Navigation Collection structure as a folder.

**Edit Link** Click to access the Edit Link page. This option displays when you select a content reference link in the Navigation Collection hierarchy.

**Delete Link** Click to access the Navigation Collections - Delete Confirmation page, where you are prompted to confirm or cancel the deletion of the selected content reference link. This option displays when you select a content reference link in the Navigation Collection hierarchy.

**Add Folder** Click to access the Add Folder page. This option displays when you select the Navigation Collection root node folder or user-defined folders in the Navigation Collection hierarchy.

If you use this option at the root node level of the hierarchy, the folder you create appears directly off the root node. Using this option while in any folder in the hierarchy creates a folder within the selected folder.

**Edit Folder** Click to access the Edit Folder page. This option displays when you select any folder other than the root node folder.

**Delete Folder** Click to access the Navigation Collections - Delete Confirmation page, where you are prompted to confirm or cancel the deletion of the selected folder. This option displays when you select any folder other than the root node folder in the Navigation Collection hierarchy. This only deletes the folder from the Navigation Collection.



Designates a folder whose content is user-defined . These folders are added on the Add Folder page using the *User Defined Folder* folder type.

Click to expand the node and access options that enable you to edit or delete the folder.

Alternatively, select the associated folder name link to display options for editing or deleting the folder, but not expand folder contents.



Designates a folder whose content is menu-based. These folders are added on the Add Folder page using the *Menu Folder* folder type.

You cannot view folder contents because a menu-based folder automatically inherits the content and folder references registered in the selected menu folder. This content cannot be edited from within Navigation Collections.

Select the associated folder name link to access options for editing aspects of the folder or delete the folder from the Navigation Collection.



Designates a content reference link in the hierarchy.


Click to access options that enable you to edit or delete the content reference from the Navigation Collection.


## Adding a Folder

Access the Add Folder page (Click the Add Folder button on the Maintain Collection page).


### Add Folder

\*Folder Type: User Defined Folder ▼

Label:  

Description:    
(254 Characters)

**Override Options**

Override Image:  

Add Folder page (user-defined folder)

## Add Folder

\*Folder Type:

\*Source Portal:

\*Menu Folder:

Label:

Description:  
(254 Characters)

**Override Options**

Override Image:

Override Label:

Override Description:  
(254 Characters)

OK Cancel [Find Source](#)

Add Folder page (menu folder)

### Folder Type

Select one of these values:

*Menu Folder:* Creates a reference to an existing folder in the portal registry. The reference automatically inherits all of the content and folder references registered in the selected menu folder.

*User Defined Folder:* Creates a user-defined folder, enabling you to specify the folder label, description, and valid from and to dates for the folder.

### Source Portal

This field displays when the Folder Type value is set to *Menu Folder*.

Select the portal registry from which you want to select a folder. The current portal is selected by default. Available source portal values include all portals defined in the PeopleSoft databases, as well as a special *Local Portal* value. The *Local Portal* option is useful for cases when a Navigation Collection will be copied between portals, and the folder reference you are defining should automatically refer to the current portal.

See [Chapter 5, "Administering Portals," page 99](#).

### Source Folder

This field displays when the Folder Type value is set to *Menu Folder*.

Select a folder to reference. Click the Browse Portal Registry button to access the Select Source Folder page, where you can view a graphical display of the selected portal registry and select the folder you want to reference.

**Label**

If you are adding a menu-based folder, the label on the source populates this field, but can be overridden in the Override Label field.

This text appears as link text for the folder on the Maintain Collection page, as well as on the Navigation Collection page or pagelet.

**Description**

If you are adding a menu-based folder, the description on the source populates this field, but can be overridden in the Override Description field.

This text displays as hover text for a folder link, as well as descriptive text beneath the folder link on the Navigation Collection page or pagelet.

***Override Options*****Override Image**

Select the image that you want to display as the icon for the folder reference. If a default folder image has been defined at the system or registry level, this image overrides the default. This image displays as the large image for the folder on the Navigation Collection publication.

When defined for folders nested within folders, the image displays on the Navigation Collection page that is accessed to display the contents of the parent folder.

If you override the image, changes to the image on the source reference are not reflected on the Navigation Collection publication. If you want source reference changes to be reflected on the Navigation Collection publication, do not override this image.

See [Chapter 4, "Working With Navigation Pages," Understanding Navigation Collection Options, page 53](#) and [Chapter 4, "Working With Navigation Pages," Navigation Collection Images, page 50](#).

**Override Label**

If you are adding a menu-folder, you can enter a label to override label text in the Label field.

If you override label text, changes to label text on the source reference are not reflected on the Navigation Collection publication. If you want source reference changes to be reflected on the Navigation Collection publication, do not enter override text in this field.

**Override Description**

If you are adding a menu-based folder, you can enter a description to override description text in the Description field.

If you override description text, changes to description text on the source reference are not reflected on the Navigation Collection publication. If you want source reference changes to be reflected on the Navigation Collection publication, do not enter override text in this field.

**Find Source**

This link displays for users who can access Structure and Content when the Folder Type value is set to *Menu Folder*.

Select to access the Folder Administration page, where you can view details about the selected menu folder. You may want to access this page to add a source reference image, change a source reference label, or check on source reference security.

See [Chapter 5, "Administering Portals," Administering Folders, page 105](#).

**Selecting a Source Folder**

Access the Select Source Folder page (click the Browse Portal Registry icon on the Add Folder page).


**Select Source Folder**












\*Source Portal:  ▼

▶ Search

Left | Right

---

 Root
 

-  [My Favorites](#)
-  [Worklist](#)
-  [Tree Manager](#)
-  [Reporting Tools](#)
-  [PeopleTools](#)
-  [\[Algorithm Chain\]](#)
-  [\[Encryption Profile\]](#)
-  [\[Upgrade Conversion\]](#)
-  [\[Change My Password\]](#)
-  [\[My Personalizations\]](#)
-  [\[My System Profile\]](#)
-  [\[My Dictionary\]](#)
-  [\[My Feeds\]](#)

---

Select Source Folder page

**Source Portal**

Select the portal registry from which you want to select a folder to reference. The portal designated on your originating page is selected by default. Available source portal values include all portals defined in the PeopleSoft databases, as well as a special *Local Portal* value. The *Local Portal* option is useful for cases when a Navigation Collection will be moved between portals, and the folder reference you are defining should automatically refer to the current portal.

See [Chapter 5, "Administering Portals," page 99](#).

## Search

### Search by

Select one of these values:

*Description:* Select to conduct your search based on folder description text. This search is not case sensitive. You can perform partial match searches. For example, doing a label search on "source" may not return any "source" hits, but may return hits on "resource."

*Label:* Select to conduct your search based on folder label text.

Alternatively, search by drilling down into the tree structure displayed at the bottom of the page.

## Search Results

### Label

Select the linked label text to locate and select the referenced folder in the portal registry hierarchy displayed at the bottom of the page.

### Description

Displays the text description of the folder.

Once you have located your desired folder, click the folder link in the hierarchy to select the folder and return to your originating page.

## Selecting Navigation Collection Images

Access the Browse Image Catalog page (click the Browse Image Catalog button on the Add Folder page).

## Browse Image Catalog

Choose an image you would like to use

Search By:

Customize   Find   View 100      First 1-10 of 1078 Last				
	Image Name	Description	WxH (Pixels)	
<input type="radio"/>	AE_STEP_TBL42	Deprecated Image: Do Not Use	16 x 16	
<input type="radio"/>	AE_STEP_TBL44	Deprecated Image: Do Not Use	16 x 16	
<input type="radio"/>	AE_STEP_TBL45	Deprecated Image: Do Not Use	16 x 16	
<input type="radio"/>	AE_STEP_TBL47	Deprecated Image: Do Not Use	16 x 16	
<input type="radio"/>	AMM_CAMA	PSCAMA Icon	16 x 16	
<input type="radio"/>	AMM_CAMA_2	Deprecated Image: Do Not Use	16 x 16	
<input type="radio"/>	AMM_FIELD_1	Field icon - 1	16 x 16	
<input type="radio"/>	AMM_FIELD_2	Deprecated Image: Do Not Use	16 x 16	
<input type="radio"/>	AMM_MINUS	Deprecated Image: Do Not Use	9 x 9	
<input type="radio"/>	AMM_MSG	Message Data Icon	17 x 16	

Browse Image Catalog page

### Search by

Select one of these values:

*Name:* Select to conduct your search based on the image file name.

*Description:* Select to conduct your search based on image description text.

You can further refine your search by selecting the adjacent *begins with* or *contains* option.

This search is not case sensitive. You can perform partial match searches. For example, doing a search on "source" using the *Description* and *contains* options may not return any "source" hits, but may return hits on "resource."


Once you have located the desired image, select the radio button for the image and click OK.


## Editing a Folder

Access the Edit Folder page (click the Edit Folder button on the Maintain Collection page).



## Edit Folder

**\*Folder Type:**

**Label:**  

**Description:**    
(254 Characters)


**Override Options**

**Override Image:**     
Use on approved FAN page only

**Placement Properties**

To move the current folder to another folder, select Move to New Parent Folder. To change the sequencing of the current folder, make a selection from the Placement in Folder dropdown. Note that placement values reflect folder sequencing that is already saved to the database.

**Parent Folder:**

**Placement in Folder:**  

Edit Folder page

With the exception of the Placement Properties group box, which is discussed below, all other options on this page are available on and documented as a part of the Add Folder page.

See [Chapter 4, "Working With Navigation Pages," Adding a Folder, page 68.](#)

### Placement Properties

<b>Move to New Parent Folder</b>	Click to move the folder you are editing to a folder other than the folder designated in the Parent Folder field.  You can only change the placement of a folder reference after you have initially saved the Navigation Collection.
<b>Parent Folder</b>	Displays the parent folder of the folder you are editing.
<b>Placement in Folder</b>	Select a value to change the placement of the folder within the parent folder. Available placement values reflect saved folder sequencing.

## Adding a Link

Access the Add Link page (click the Add Link button on the Maintain Collection page).

## Add Link

\*Source Portal:

\*Menu Item:

Label: Query Manager

Description: Create or update queries and query specifications.  
(254 Characters)

☐ Open in a new window

Additional Parameters:

Example: name1=value1&name2=value2

**Override Options**

Override Image:

Override Label:

Override Description:   
(254 Characters)

OK Cancel [Find Source](#)

Add Link page

### Source Portal

Select the portal registry from which you want to select a content reference. The current portal is selected by default. Available source portal values include all portals defined in the PeopleSoft databases, as well as a special *Local Portal* value. The *Local Portal* option is useful for cases when a Navigation Collection will be copied between portals, and the content reference you are defining should automatically refer to the current portal.

See [Chapter 5, "Administering Portals," page 99](#).

### Source Link

Select a content reference to add to the Navigation Collection. Click the Browse Portal Registry button to access the Select Source Link page, where you can view a graphical display of the selected portal registry and select the content reference you want to add.

You can use this option to add a published navigation page to the Navigation Collection that you are creating.

### Label

Displays the existing menu content reference label text, but you can override this default using the Override Label field. This text appears as link text for the content reference on the Maintain Collection page, as well as on the Navigation Collection publication.

### Description

Displays the existing description text, but you can override the default using the Override Description field. This description text displays as hover text for the content reference link.

- Open in a new window** Select if you want the page accessed by the content reference to open in a new browser window.
- Additional Parameters** Define additional query parameters that are appended to the uniform resource locator (URL) used to access the content reference. Defining these query parameters here make it unnecessary to register the same content reference multiple times with different parameters.
- For example, you can use these additional parameters to point users to a specific row of data on an application business transaction page. The query string parameter can be the search record field name = field value.
- These query parameters are stored as attributes with the content reference and are applied to the URL at runtime.

### **Override Options**

- Override Image** Select the image to display as the large image for the content reference. If a default content image has been defined at the system or registry level, the selected image overrides the default. This large image displays for the content reference on the Navigation Collection publication.
- See [Chapter 4, "Working With Navigation Pages," Understanding Navigation Collection Options, page 53.](#)
- Override Label** Enter a label to override label text in the Label field.
- If you override label text, changes to label text on the source reference are not reflected on the Navigation Collection publication. If you want source reference changes to be reflected on the Navigation Collection publication, do not enter override text in this field.
- Override Description** Enter a description to override description text in the Description field.
- If you override description text, changes to description text on the source reference are not reflected on the Navigation Collection publication. If you want source reference changes to be reflected on the Navigation Collection publication, do not enter override text in this field.
- Find Source** This link displays for users who can access Structure and Content once a Source Link value has been selected.
- Select to access the Content Ref Administration page, where you can view content reference details, such as security, image attributes, valid to and from dates, and content provider for the selected source content reference.
- See [Chapter 5, "Administering Portals," Administering Content References, page 109.](#)

## **Selecting a Source Link**

Access the Select Source Link page (click the Browse Portal Registry button on the Add Link page).


## Select Source Link















\*Source Portal:

▶ Search

Left | Right

---

 Root
 

-  My Favorites
-  Maintain Partner Information
-  Catalog Management
-  Worklist
-  Tree Manager
-  Reporting Tools
-  PeopleTools
  -  [\[Upgrade Conversion\]](#)
  -  [\[Warehouse Tools\]](#)
  -  [\[Change My Password\]](#)
  -  [\[My Personalizations\]](#)
  -  [\[My System Profile\]](#)
  -  [\[My Dictionary\]](#)
  -  [\[My Feeds\]](#)

---

Select Source Link page

### Source Portal

Select the portal registry from which you want to select a content reference. The portal designated on your originating page is selected by default. Available source portal values include all portals defined in the PeopleSoft databases, as well as a special *Local Portal* value. The *Local Portal* option is useful for cases when a Navigation Collection will be moved between portals, and the content reference you are defining should automatically refer to the current portal.

See [Chapter 5, "Administering Portals," page 99](#).

### Search

#### Search by

Select one of these values:

*Description:* Select to conduct your search based on content reference description text. This search is not case sensitive. You can perform partial match searches. For example, doing a label search on "source" may not return any "source" hits, but may return hits on "resource."

*Label:* Select to conduct your search based on content reference label text.

Alternatively, search by drilling down into the tree structure displayed at the bottom of the page.

## ***Search Results***

**Label** Select the linked label text to locate and select the content reference in the portal registry hierarchy displayed at the bottom of the page.

**Description** Displays the text description of the content reference.

Once you have located your desired content reference, click the folder link in the hierarchy to select the folder and return to your originating page.

## **Editing a Link**

Access the Edit Link page (click the Edit Link button on the Maintain Collection page).

Edit Link

\*Source Portal:

EMPLOYEE

\*Menu Item:

PSXPDATASRC\_GBL

Label:

Data Source

Description:  
(254 Characters)

Register schema and sample data files for XML Publisher data sources.

☐ Open in a new window

Additional Parameters:

Example: name1=value1&name2=value2

Override Options

Override Image:

Override Label:

Override Description:  
(254 Characters)

Placement Properties

To move the current link to another folder, select Move to New Parent Folder. To change the sequencing of the current link, make a selection from the Placement in Folder dropdown. Note that placement values reflect link sequencing that is already saved to the database.

Move to New Parent Folder

Parent Folder:

Register Data Sources

Placement in Folder:

01 - First

OK

Cancel

[Find Source](#)

Edit Link page

With the exception of the Placement Properties group box, which is discussed below, all other options on this page are available on and documented as a part of the Add Link page.

See [Chapter 4, "Working With Navigation Pages," Adding a Link, page 74.](#)

Placement Properties

Move to New Parent Folder

Click to move the content reference you are editing to a folder other than the folder designated in the Parent Folder field.

Parent Folder

Displays the parent folder of the content reference you are editing.

Copyright © 1988, 2009, Oracle and/or its affiliates. All Rights Reserved.

79



Maintain Collection
Publish Collection

---

**Name:**

Publishing Options

Publishing a Navigation Collection enables it to be accessed by users. Select one or more of the following options to publish this Navigation Collection. Set the Security Access for each publishing option. Selecting 'Allow Collection Sync' requires running the Sync Collection Security process for this collection. Selecting 'Do Not Allow Collection Sync' requires manually adding the security to the published page or pagelet. Selecting 'Enable Caching' turns on role-based caching.

☒ **Navigation Pagelet**

Security Access

☒ Public Access
☐ Allow Collection Sync
☐ Do Not Allow Collection Sync

☒ **Enable Caching**

\*Pagelet Category:

☒ **Navigation Page**

Security Access

☒ Public Access
☐ Allow Collection Sync
☐ Do Not Allow Collection Sync

☒ **Enable Caching**

EMPLOYEE > Root

Publish Collection page

**Name** Displays the name of the Navigation Collection defined on the Maintain Collection page.

### ***Publishing Options***

**Navigation Pagelet** Select to publish the Navigation Collection as a navigation pagelet, which can be added to a user's homepage tab.





## Pages Used to Publish Pagelets to Other Portals

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Publish to Multiple Portals	PTPP_PMPUBPRTL	PeopleTools, Portal, Portal Utilities, Publish Pagelets, Publish to Multiple Portals	Copy a pagelet from a source portal and publish it to multiple target portals that reside in the same database as the source portal.
Publish Multiple Pagelets	PTPP_PMPUBPGLT	PeopleTools, Portal, Portal Utilities, Publish Pagelets, Publish Multiple Pagelets	Copy multiple pagelets from a source portal and publish them to a selected target portal that resides in the same database as the source portal.

## Publishing a Pagelet to Multiple Portals

Access the Publish to Multiple Portals page (PeopleTools, Portal, Portal Utilities, Publish Pagelets, Publish to Multiple Portals).



## Target Portals

---

**Warning!** Selecting a target portal in which the selected pagelet already exists overwrites the pagelet content reference in the target portal.

---

<b>Select</b>	Select to publish the selected pagelet to the selected target portal. Use the Portal Name and Title fields to identify the available target portals.
<b>Pagelet Exists As</b>	<p>If the pagelet you have selected to copy and publish to a target portal already exists in the target portal, this field is populated with the pagelet title as it exists in the portal registry.</p> <p>For instance, in the example shown here the Technology News pagelet is selected for copying from the EMPLOYEE source portal to the CUSTOMER target portal. The Pagelet Exists As field shows that this pagelet also already exists in the CUSTOMER target portal. If you copy the pagelet to the CUSTOMER portal, the pagelet definition in the CUSTOMER portal is overwritten by the pagelet definition in the EMPLOYEE source portal.</p>
<b>Go to Tab Content/Layout</b>	Click this link to display the Tab Content and Tab Layout pages where you can specify display settings for the pagelet.
<b>Publish Pagelet</b>	<p>Select to copy the pagelet definition from the source portal and publish it in the selected target portals.</p> <p>Content reference portal registry entries are created in the registries of all selected target portals using the same values that exist in the original pagelet registration in the source portal.</p> <p>If the folder in which the original pagelet is registered does not exist for a selected target portal registry, it is dynamically generated.</p>

## Publishing Multiple Pagelets to a Portal

Access the Publish Multiple Pagelets page (PeopleTools, Portal, Portal Utilities, Publish Pagelets, Publish Multiple Pagelets).



<b>Pagelet Type</b>	<p>Select the type of pagelet you want to copy. Options are:</p> <ul style="list-style-type: none"> <li>• <i>Homepage Pagelet.</i> A homepage pagelet is a version of a pagelet that is available for use on the portal homepage and can be configured by the end user.</li> <li>• <i>Template Pagelet.</i> A template pagelet is a version of a pagelet that can be added to a PeopleSoft template. These pagelets can extract and leverage keys specified by the target transaction for the template. Publishing a pagelet as a template pagelet is useful when the the PeopleTools Context Manager feature will be used to display the template pagelet.</li> </ul>
<b>Pagelet Keyword</b>	Enter keywords to narrow the scope of pagelets displayed in the Pagelet group box.

### ***Pagelets***

<b>Select</b>	Select to publish selected pagelets to the selected target portal. Use the Pagelet ID, Title, and Description fields on the Title and Description tabs to identify the pagelet.
<b>Publish Selected Pagelets</b>	<p>Select to copy the pagelet definition from the source portal and publish it any selected target portals.</p> <p>Content reference portal registry entries are created in the target portal registry using the same values that exist for the original pagelet in the source portal registry.</p> <p>If a folder in which an original pagelet is registered does not exist for a selected target portal registry, it is dynamically generated.</p>

---

## **Running Portal Utility Processes**

This section provides overviews of portal utility processes and discusses how to:

- Run the Synchronize Collection Security process.
- Run the Delete Empty Folders process.
- Run the Clean Portal Project process.

## **Understanding Portal Utility Processes**

This section discusses:

- The Synchronize Collection Security Process.
- The Delete Empty Folders Process.

- The Clean Portal Project Process.

### ***The Synchronize Collection Security Process***

The Synchronize Collection Security Application Engine process (PTPP\_SCSYNC) updates security for published Navigation Collection pages and pagelets by adding permission lists and/or roles defined for content references and folders in the Navigation Collection to the published navigation page or pagelet permissions.

The Synchronize Collection Security process runs for Navigation Collection publications according to the Security Access options set for the publication on the Publish Collection page. The process runs for Navigation Collection publications that are published with the *Allow Collection Sync* option selected. The process is not run for Navigation Collection publications that are published with the *Do Not Allow Collection Sync* option selected. The process removes extraneous security objects for Navigation Collections that were published with the *Public Access* option selected.

Run this process after security updates, and when you publish or change a navigation page or pagelet with the *Allow Collection Sync* option selected.

See [Chapter 4, "Working With Navigation Pages," Publishing Navigation Collections, page 80.](#)

### ***The Delete Empty Folders Process***

The Delete Empty Folders Application Engine process (PTPP\_FD\_DEL) deletes portal registry folder references that do not contain child folders or content. The process does not delete empty Navigation Collection folder placeholders, nor the My Favorites folder. The user ID running the process must have been assigned the security role of Portal Administrator.

Run this process when you are deleting content references during an upgrade or implementation, and you are removing out-of-date navigation and replacing it with the current delivered navigation. You should also run this process after copying a "delete" portal project.

### ***The Clean Portal Project Process***

The Clean Portal Project Application Engine process (PTPP\_CPPROJ) removes unnecessary and potentially harmful data from a portal project definition before you copy it to the PeopleSoft Enterprise Portal. Run this process as a part of an upgrade or implementation when you want to load navigation from a content provider into the PeopleSoft Enterprise Portal.

The Clean Portal Project process removes common objects from the content provider portal project that, because they are common, also exist in the PeopleSoft Enterprise Portal. Run this process on the portal project before copying it to the PeopleSoft Enterprise Portal so that you don't override these common objects configured for use with the PeopleSoft Enterprise Portal with the common objects that were configured in the content provider.

The Clean Portal Project process ensures that you only copy the navigation objects that you feel are necessary, such as registry structures, templates, and permissions.

**Warning!** Navigation Collections cannot be rendered in the PeopleSoft Enterprise Portal if the content provider registry is hosted by a non-local node.

Because Navigation Collections actively access the registry, a published Navigation Collection page or pagelet cannot be rendered for a registry that is hosted by a non-local node. If you want to render a content provider Navigation Collection page or pagelet in the PeopleSoft Enterprise Portal, the content provider's registry must be hosted by a local node in the content provider database.

See [Chapter 5, "Administering Portals," page 99](#).

## Pages Used to Run Portal Utility Processes

Page Name	Definition Name	Navigation	Usage
Synchronize Collection Security	PTPP_SCSYNC_RUN	PeopleTools, Portal, Portal Utilities, Sync Collection Security	Set run control parameters for the Sync Collection Security process.
Delete Empty Folders	PTPP_FD_RUN	PeopleTools, Portal, Portal Utilities, Delete Empty Folders	Set run control parameters for the Delete Empty Folders process.
Clean Portal Project	PTPP_CPPROJ_RUN	PeopleTools, Portal, Portal Utilities, Clean Portal Project	Set run control parameters for the Clean Portal Project process.

## Running the Synchronize Collection Security Process

Access the Synchronize Collection Security page (PeopleTools, Portal, Portal Utilities, Sync Collection Security).



### Synchronize Collection Security

The Synchronize Collection Security process updates the security on the published Navigation Collection pages. Select a Collection Name to only synchronize the published pages for that collection. Leave the Collection Name blank to synchronize all the published pages in the specified Portal Name. The process adds the security objects of the folders and links, in the collection definition, to the published pages that are marked as 'Allow Collection Sync'. It removes security objects from the published pages marked as 'Public'.

Run Control ID: 1

[Report Manager](#) [Process Monitor](#)

[Run](#)

Request Parameters	
*Portal Name:	<input type="text" value="CUSTOMER"/>  Customer-facing registry content
Collection Name:	<input type="text"/> 

Synchronize Collection Security page

**Request Parameters**

<b>Portal Name</b>	Select the portal for which you want to synchronize security for published Navigation Collections.
<b>Collection Name</b>	<p>Select the Navigation Collection for which you want to synchronize security. Only Navigation Collections created for the selected portal are available for selection.</p> <p>Leave this field blank if you want to synchronize security for all non-public Navigation Collection publications that exist in the selected portal.</p> <hr/> <p><b>Note.</b> Navigation Collection pages or pagelets that were published with the <i>Do Not Allow Security Sync</i> option selected on the Publish Collection page are not eligible for processing.</p> <hr/>

**See Also**

Chapter 4, "Working With Navigation Pages," Publishing Navigation Collections, page 80

**Running the Delete Empty Folders Process**

Access the Delete Empty Folders page (PeopleTools, Portal, Portal Utilities, Delete Empty Folders).

**Delete Empty Folders**

The Delete Empty Folder process deletes empty portal registry folder references that do not contain any child folders or child content. It will not delete empty Navigation Collection folder placeholders or the 'My Favorites' folder. The User ID running this process must have the security role 'Portal Administrator'.

Run Control ID: 1

[Report Manager](#) [Process Monitor](#)

Run

Request Parameters

\*Portal Name:

EMPLOYEE

Delete Empty Folders page

**Request Parameters**

<b>Portal Name</b>	Select the portal for which you want to delete portal registry folder references that do not contain child folders or content.
--------------------	--

## Running the Clean Portal Project Process

Access the Clean Portal Project page (PeopleTools, Portal, Portal Utilities, Clean Portal Project).

### Clean Portal Project

The Clean Portal Project process removes unnecessary data from a Portal Project definition before it is copied to the PeopleSoft Enterprise Portal. To create a clean Portal Project, perform the following tasks.

- 1) Create a project in Application Designer, include all Permission Lists and a Portal Registry Definition, include the Related Definitions of Portal Registry Structures.
- 2) Run this Clean Portal Process against the created project.
- 3) Delete the 2-tier database cache using Configuration Manager.
- 4) In Application Designer, export the cleaned project to file.


Run Control ID: 2

[Report Manager](#) [Process Monitor](#)

Run


\*Project Name:

PPLTOOLS

 Composite PeopleTools Project


\*Portal Name:

EMPLOYEE

 Employee-facing registry content

\*Content Provider Name:

PT\_LOCAL

 [Node Definition](#)

Project Options

☐ Full Navigation

☒ Select Objects

Clean Portal Project page (1 of 2)

Select Objects

☒ Include Pagelets

Select Pagelets

To include content references for Pagelet Personalizations you must have the Content URI text populated for the Content Provider Name selected above. Use the Node Definition link provided to add the Content URI text.

☐ All Pagelets
☒ Selected Pagelets

	*Pagelet Name	Pagelet Label	
1	<input type="text"/>		<input type="button" value="+"/> <input type="button" value="-"/>

☒ Include Collection Pages

Select Collection Pages

☐ All Collection Pages
☒ Selected Collection Pages

	*Collection Page Name	Collection Page Label	
1	<input type="text"/>		<input type="button" value="+"/> <input type="button" value="-"/>

☒ Include Folders

Select Folders

☐ All Folders
☒ Selected Folders

	*Folder Name	Folder Label	
1	<input type="text"/>		<input type="button" value="+"/> <input type="button" value="-"/>

Clean Portal Project page (2 of 2)

**Project Name** Select the portal project that you want to clean for export into the PeopleSoft Enterprise Portal. Projects available for selection are those in the database that contain at least one registry structure.

**Portal Name** Select the portal that contains the registry structures you want to copy. Available portal names are derived from the portal project you selected, as well from as the project data that contains the portal name of the registry structures.

**Content Provider Name** Select the content provider node name of the registry structure you want to include in your portal project. Available node names are derived from the portal project you selected.

Copyright © 1988, 2009, Oracle and/or its affiliates. All Rights Reserved.

93



**Include Folders**

Select to include folders in the portal project in the run of the Clean Portal Project process. When you select this option, the Select Folders group box displays, enabling you to specify which folders you want to be processed.

Selecting a folder for inclusion in the process also includes all of its children and parents for copying into the PeopleSoft Enterprise Portal.

**Select Pagelets****All Pagelets**

Select to include all pagelets in the portal project in the run of the Clean Portal Project process.

**Selected Pagelets**

Select to include only selected pagelets in the portal project in the run of the Clean Portal Project process. Selecting this option displays the Pagelet Name field, enabling you to select specific pagelets that you want to be processed.

**Select Collection Pages****All Collection Pages**

Select to include all published Navigation Collection pages in the portal project in the run of the Clean Portal Project process.

**Selected Collection Pages**

Select to include only selected published Navigation Collection pages in the portal project in the run of the Clean Portal Project process. Selecting this option displays the Collection Page Name field, enabling you to select specific published Navigation Collection pages that you want to be processed.

**Select Folders****All Folders**

Select to include all folders in the portal project in the run of the Clean Portal Project process.

**Selected Folders**

Select to include only selected folders in the portal project in the run of the Clean Portal Project process. Selecting this option displays the Folder Name field, enabling you to select specific folders that you want to be processed.



## **Part 3**

# **Working with PeopleTools Portals**

### **Chapter 5**

#### **Administering Portals**

### **Chapter 6**

#### **Administering Portal Homepages and Pagelets**

### **Chapter 7**

#### **Modifying the Portal Interface**

### **Chapter 8**

#### **Working with Portal Templates**

### **Chapter 9**

#### **Using Portal Caching Features**

### **Chapter 10**

#### **Configuring the Portal Environment**



## Chapter 5

# Administering Portals

This chapter provides an overview of portal administration and discusses how to:

- Administer portal definitions.
- Administer folders.
- Administer content references.
- Manage folders and content references.
- Manage general portal settings.
- Manage portal objects.
- Review menu item information.
- Build registry search indexes.

---

## Understanding Portal Administration

Every component delivered with PeopleSoft applications is pre-registered within the portal registry as a content reference. Common administrative tasks include adding, deleting, and renaming portal definitions. If you have portal administrator access, you can use the administration pages to manage the registry, including folders, content references, and security.

Additionally, to provide programmatic access to the portal registry, PeopleSoft delivers a portal registry API. This API provides the same kind of registry management capability as the administration pages and is fully described in the PeopleCode documentation. You can also manage the registry with the registration wizard.

You use the administration pages to:

- Manage the structure and content of a portal, including folders and external content references.
- Manage general portal settings, including the portal description and node templates.
- Populate a portal registry with custom menus by importing menu definitions.
- Generate a Verity search index or collection for use with your portal, based on the content of the portal registry.
- Add, edit, and delete local and remote portal definitions.





















## Pages Used to Administer Content References

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Content Ref Administration (content reference administration)	PORTAL_CREF_ADM	PeopleTools, Portal, Structure and Content, Portal Objects  Navigate to the Content References list and click Edit.. Or click Add Content Reference.	Add new content references or edit existing content references.
Content Reference Security	PORTAL_CREF_SEC	PeopleTools, Portal, Structure and Content, Portal Objects, Content Ref Administration, Security	Set content reference security.
User ID Queries	USER_QUERY	PeopleTools, Security, User Profiles, User Profiles, User ID Queries	Review content reference security by user.
Permission List Queries	PLIST_QUERIES	PeopleTools, Security, Permissions & Roles, Permission Lists, Permission List Queries	Review content reference security by permission list.
Role Queries	ROLE_QUERY	PeopleTools, Security, Permissions & Roles, Roles, Role Queries	Review content reference security by role.
Select a Content Reference or Content Reference Link	PORTAL_CREF_SELECT	PeopleTools, Portal, Structure and Content, Portal Objects  Navigate to the Content References list and click Add Content Reference Link.	Select a content reference or content reference link.

<b>Page Name</b>	<b>Definition Name</b>	<b>Navigation</b>	<b>Usage</b>
Content Reference Link Administration	PORTAL_CREF_LNK	<ul style="list-style-type: none"> <li>Click a content reference link on the Content Reference or Content Reference Link page.</li> <li>PeopleTools, Portal, Structure and Content, Portal Objects</li> </ul> <p>Navigate to the Content References list and click Create Link.</p> <ul style="list-style-type: none"> <li>PeopleTools, Portal, Structure and Content, Portal Objects</li> </ul> <p>Navigate to the Content References list and click Edit for any content reference of usage type <i>Target</i>.</p> <p>Click Create Content Reference Link.</p>	Add new or edit existing content reference links.
Related Links Group	PORTAL_RLNK_ADM	<ul style="list-style-type: none"> <li>PeopleTools, Portal, Structure and Content, Portal Objects</li> </ul> <p>Navigate to the Content References list and click the Edit link.</p> <ul style="list-style-type: none"> <li>Select the template name <i>RELATEDLINKS_TEMPLATE</i>.</li> </ul> <p>Enter <i>RELLINK</i> in the Name field of the Content Reference Attributes section.</p> <p>Enter a label for the attribute.</p> <p>Click the Edit Related Links link.</p>	Create related links groups.
Security	PORTAL_CLINK_SEC	PeopleTools, Portal, Structure and Content, Portal Objects, Content Reference Link Administration, Security	Set content reference link security.



































































































































































- Using inline frames enables you to click the navigation and see only the target content change. The entire page does not refresh. This approach allows for much more flexibility and makes it easier for the header and navigation to work together.
- When navigating to different content within the iframe template, the URL in the browser address bar may not be up to date.
- If you create your own iframe-based template, include the `!DOCTYPE` at the top of the code.

---

## Developing Portal Templates

This section discusses:

- Template pagelets based on pages.
- Inheritance of style classes.
- Template pagelet tags.
- Considerations for non-PeopleSoft content providers.
- Considerations for frame-based templates.
- Partial path nodes.
- Pagelet names in templates.
- Considerations for forms and templates.
- Error message display.

### Template Pagelets Based on Pages

Template pagelets can be provided by any URL, but usually they are based on either a page or an iScript. If you are using template pagelets based on pages, add the query string parameter `&target= PanelGroupName` at the end of the query string parameter list for the template pagelet in the template HTML. This prevents PeopleSoft Pure Internet Architecture from assuming the same default name, `Main`, for each page. Failure to specify the target parameter results in a JavaScript error when the portal servlet tries to load the template. If there is more than one form with the name `Main` on the same page, the JavaScript on that page cannot determine which `Main` component to use. In the template HTML, add the target parameter to the source specification of each page-based template pagelet that is not meant to be the target content.

Here is an example:

```
<tr>
<td><Pagelet Name="Related Links">
    <Source Node="LOCAL_NODE" href="c/PORTAL_ADMIN.PORTAL_RLNK_1COL.GBL?target=>
relatedlinks"/></Pagelet>
</td>
</tr>
```

## Inheritance of Style Classes

When the portal servlet assembles portal pages based on PeopleSoft Application Designer pages, the assembled pages inherit the style classes that are defined for them in PeopleSoft Application Designer. Because you can develop a template that includes pagelets built with different style sheets in PeopleSoft Application Designer, you can have conflicting style classes in your template.

For example, if you have a style class named Big on page X and you also have a style class named Big on page Y, there may be conflicting styles on the resulting page. When conflicting style classes exist, the style sheet that is associated with the page that is used for the target content always takes precedence over competing style sheets in the template.

## Template Pagelet Tags

Template HTML can contain three PeopleSoft-specific tags that are used as portal servlet directives to assemble content on the page: Pagelet tags, Source tags, and Target tags.

### ***Pagelet***

Note the following attribute, element, and content information for the Pagelet tag:

<b>Attributes</b>	Name. Used to identify the component in a comment in the final assembled page.
<b>Elements</b>	Source tag.
<b>Contents</b>	None.

### ***Source***

Note the following attribute, element, and content information for the Source tag:

<b>Attributes</b>	Product tag. Used to identify to which PeopleSoft Pure Internet Architecture web server to route. The Product tag value should be the name of the registered content provider for the content.
<b>Elements</b>	None.
<b>Contents</b>	Query string to access the page or iScript implementing the component.

Contents must be escaped. Ampersands must be written as *&amp;*. Use the IClientComponent tag to specify content that is not specific to the PeopleSoft Pure Internet Architecture. If a pagelet is implemented by a web server that is not a registered content provider, the entire URL of the pagelet can be specified in the Source tag contents, and the product attributes can be omitted.

### ***Target***

Note the following attribute, element, and content information for the Target tag.

<b>Attributes</b>	Name. Used to identify the component in a comment on the final assembled page.
<b>Elements</b>	None.
<b>Contents</b>	None.

## Considerations for Non-PeopleSoft Content Providers

To determine which template to apply to requested content, the portal servlet invokes the portal registry API to look up the content reference that is associated with the target content URL.

---

**Warning!** The information in this section applies to any content that is generated by a system where the PeopleTools release is earlier than 8.4, or any time the portal cannot determine from the submitted URL what content reference is being requested.

---

The same URL can supply different content pages depending on the context. For example, with PeopleSoft application pages, the URL for submit buttons before PeopleTools 8.4 always looks like this:  
<http://server/servlets/iclientservlet/peoplesoft8>.

Because of this, the target content can specify its registered URL by using a custom response header, `PortalRegisteredURL`. When the target content supplies this response header, the portal servlet uses it to identify the URL that is registered in the portal, rather than using the actual URL that is used to get the content. All PeopleSoft Pure Internet Architecture technologies, such as pages, iScripts, and queries, supply this URL.

The value of this header must be the string with which the page should be registered. You must register the content with the identical string that it will return as this header.

If content does not return the header information, the portal servlet identifies the content by the URL in the request that is submitted to the portal.

You can override the value of the `PortalRegisteredURL` response header in a PeopleSoft Pure Internet Architecture script or page by adding the header to the response yourself, like this:

```
%Response.SetHeader("PortalRegisteredURL", &myURL);
```

You can do this to register the content with additional parameters.

## Considerations for Frame-Based Templates

In a frame-based template, all relative URL references refer back to the psc servlet. To construct a URL reference to a page that is assembled by the portal servlet, use either the `GenerateXXXXPortalURL` or `GenerateXXXXPortalRelURL` PeopleCode functions.

## Partial Path Nodes

URLs that are not registered but partially match a registered node have the template of the associated node applied. This helps you to manage unregistered URLs and to ensure that they achieve a consistent appearance within the portal. Most pages have the template formatting that you and portal users expect, rather than resorting to the default template when an exact node cannot be found.

---

**Note.** When the content reference isn't registered, it doesn't use the template defined for the node. Instead, it uses the default template. Therefore, unregistered URLs use the default template, not the template assigned to the node.

---

This table shows examples of nodes, the URI, and sample URLs that would use the template that is specified by the node:

<i><b>Node</b></i>	<i><b>URI</b></i>	<i><b>Sample Match</b></i>
Yahoo	http://www.yahoo.com	http://www.yahoo.com/business/webl eaders/peoplesoft.html
Yahoo Sports	http://www.yahoo.com/sports	http://www.yahoo.com/sports/nba/hea t/teamreport.html
Yahoo NFL	http://www.yahoo.com/sports/nfl	http://www.yahoo.com/sports/nfl/dolp hins/teamreport.html

## Pagelet Names in Templates

You can specify a pagelet name in the template rather than specifying a URL. This is the syntax:

```
<Pagelet Name="My Pagelet">
<Source Product="Portal" Pagelet="MY_PAGELET" />
</Pagelet>
```

If the portal servlet sees the Pagelet attribute in the Source tag, it looks up that pagelet name in the registry and uses the associated URL there.

## Considerations for Forms and Templates

When posting data from a form in an HTML template, post data is lost if the action of the form is in a frame template. To solve this problem, make the frame source for the target a special request to the portal to repost the form data. The HTML response to that request includes a form containing the posted field and value pairs in hidden input fields and a Script tag that submits that form. When the browser receives this request, the JavaScript immediately submits the form to the original URL, causing the original post request to occur.

To make the frame source for the target a special portal request:

1. Construct the frameset.

The source of the target frame is the portal, with the query string `cmd=framerepost&key=xxx` (or something similar).

2. Store the post data on the session object as a `PropertySet` (name and value pairs) for later retrieval, with a key.

The key must uniquely identify the post data. The URL to post to can be kept in the `PropertySet` or put on the query string.

3. When the portal receives a frame repost request, it uses the incoming key to look up the stored `PropertySet`, then uses `PropertySet` to construct the HTML response with the name and value pairs in hidden input fields.

The form also has a Submit button. The HTML response contains an inline JavaScript (after the form) that submits the form.

See the PeopleTools 8.50 installation guide for your database platform

### ***Combining Forms and Frames***

If you include a form on a page that's been assembled by the portal servlet, and that page's target is displayed in a frame (either in a frame template or in a nontemplate frame), then the form action cannot be Post. The portal servlet retrieves the page by using Get—not Post—regardless of the form action. Therefore, form data is lost if the form action was originally Post. Making the form action Get resolves this problem.

## **Error Message Display**

When a pagelet cannot be retrieved, an error message appears.

You can disable this error message. If the error is not displayed, then the pagelet disappears from the template without a message to the user.

To control the display of this error message, use the `DisplayError` attribute of the `IClientComponent` tag:

```
<IClientComponent Name="Example" DisplayError=false>
  <Source Product="">http://www.InvalidServer/InvalidPage.html</Source>
</IClientComponent>
```

If the portal servlet finds `DisplayError` equal to `False`, it does not display the error but adds the empty pagelet comment tags. If `DisplayError` is equal to `True`, or if the attribute is not found, the error message is displayed.

This feature works best when the HTML for the template can flow around the empty content properly.



## Chapter 9

# Using Portal Caching Features

This chapter provides an overview of portal caching and discusses how to:

- Administer server-based caching.
- Administer browser-based caching.
- Administer web server-based navigation caching.

---

## Understanding Portal Caching

Information on your PeopleSoft portal that changes frequently must be regularly updated, but for best efficiency, information that remains static should be cached for quick retrieval. Information that changes less frequently should be updated on a more deliberate schedule. The composition of changing and static information varies depending on the user, role, application, portal, web server, information structure, and information content involved.

### ***Caching Configuration Options***

When properly configured, caching significantly boosts portal performance. PeopleSoft portal technology provides a wide range of caching options. These options target different elements of the portal environment, use a variety of criteria for applying caching, and have varying degrees of scope. Not all caching options apply to all circumstances.

You can target portal caching to a specific pagelet, template, portal, or web server. You can also apply caching based on the content reference type or browser type. In some cases, you can specify the scope of caching to be by user, by role, or by application.

Some types of caching take place on the portal web server, and others are applied at the browser without requiring individual browser configuration. You can make various caching settings in the Web Profile Configuration component (WEB\_PROFILE), on a personalization page, or in a template HTML element.

With server caching, the cached data is stored on the portal web server. When a user's browser requests a piece of information from the portal, the server transmits the currently available data. The currency of the data provided depends on whether the server has refreshed the data since the last request. This is controlled by the server-based caching criteria that you defined for that data, such as a time interval or a specified number of requests.

---

**Note.** When browser caching is enabled, you can manually refresh the browser, which reloads the entire browser HTML content from the portal server. This doesn't update the information that's cached on the server, however.

---

### **Default Homepage Caching**

The default homepage is automatically cached on the server, but only when you select the Allow Public Access check box in the Public Users group box of the Web Profile Configuration - Security page. Caching the default homepage according to these rules protects the server from having to regenerate the same homepage repeatedly. This is useful for guest sign-in scenarios.

### **See Also**

*Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*, "Setting Application Server Domain Parameters," Cache Settings

[Chapter 10, "Configuring the Portal Environment," Configuring Portal Security, page 225](#)

*Enterprise PeopleTools 8.50 PeopleBook: PeopleCode Language Reference*

---

## **Administering Server-Based Caching**

This section provides an overview of caching scope and discusses how to:

- Implement target content caching.
- Implement pagelet caching.
- Implement metadata caching.
- Implement proxied JavaScript caching.

## **Understanding Caching Scope**

You can define caching for server-based portal content with one of three levels of scope:

- Private scope.

This is also known as the *user* scope. The portal maintains the cached content independently for each user. This scope is useful when the content that is displayed, such as a 401K balance, is specific to the user who's signed in.

- Role scope.

Each user is assigned one or more roles. The portal maintains a single cache of the content for all users with a given combination of roles. This scope is useful when the content that is displayed, such as the menu structure, is specific to the role set of the user who's signed in. If your user population has many unique role combinations, your caching memory requirements may be nearly the same as private scope caching.

- Public scope.

This is also known as the *application* scope. The portal maintains a single cache of the content for all users who are using a given application. This scope is useful when all users who have access to this content, such as company news articles, bulletins, or messages, see the same thing. Because only one cache object is managed in memory for your entire user population, this is the least memory-intensive scoping option.

## Implementing Target Content Caching

You can cache the content of an individual target by specifying several caching parameters in the HTML of each locally defined template that invokes the target. You must also enable target content caching for the portal.

---

**Important!** Target content is cached in memory on the web server. If you use this feature excessively, the cache could consume all available memory. Use target content caching only to cache content that is static or doesn't require constant updating and that is accessed frequently by many users.

---

### Specifying Caching Parameters

You apply target content caching by inserting a Cache element inside a TargetContent element in the HTML of a locally defined template.

The Cache element has these attributes:

Attribute	Description
Scope	Use one of these values: <ul style="list-style-type: none"> <li>• <i>user</i>: The portal maintains the cached target content independently for each user.</li> <li>• <i>role</i>: The portal maintains a single cache of the target content for all users in a role.</li> <li>• <i>application</i>: The portal maintains a single cache of the target content for all users who are using an application.</li> </ul>
Interval	Specify the number of seconds that should pass before the portal refreshes the target content cache.

This example shows caching that is defined with the application scope and an interval of 1200 seconds:

```
<TargetContent Name="MyContent">
  <Cache Scope="application" Interval="1200">ANY</Cache>
</TargetContent>
```

---

**Note.** The Cache element must not be empty, but its content can be anything.

---

### Enabling Target Content Caching

Any Cache elements that you insert in template HTML are ignored by the portal unless you enable target content caching globally for the portal. Select the Cache Target Content check box in the On the Web Server group box of the Web Profile Configuration - Caching page.

## See Also

[Chapter 8, "Working with Portal Templates," page 181](#)

[Chapter 10, "Configuring the Portal Environment," Configuring Caching, page 235](#)

## Implementing Pagelet Caching

If you're developing a pagelet and you know that its content will remain static for all users, you can specify the application or public scope. If you know that the pagelet's content will remain static for all users with the same role, you can specify the role scope. However, if the pagelet content is static only for individual users, you must specify the user or private scope. If you're not the pagelet developer, you probably won't know when the pagelet content will change, so to be sure you must specify the user scope.

---

**Important!** Pagelet content is cached in memory on the web server. If you use this feature excessively, the cache could consume all available memory. Use pagelet caching only to cache content that is static or doesn't require constant updating, and is accessed frequently by many users.

---

You can cache the content of an individual pagelet by specifying caching parameters in one of these locations:

- The HTML of any locally defined template that invokes the pagelet.
- A content reference attribute in the pagelet definition.

---

**Note.** If you specify caching parameters for a pagelet in both locations, the caching parameters in the template take precedence, and the caching parameters in the content reference attribute are ignored.

---

The presence of the caching parameters is sufficient to enable caching for the associated pagelet; no additional setting is required. To disable pagelet caching, delete the caching parameters.

### ***Specifying Pagelet Caching in the Template***

You apply pagelet caching in a template by inserting a Cache element inside a Pagelet element or IClientComponent element in a locally defined template.

The Cache element has these attributes:

Attribute	Description
Scope	<p>Use one of these values:</p> <ul style="list-style-type: none"> <li><i>user</i>: The portal maintains the cached pagelet independently for each user.</li> </ul> <p><b>Note.</b> In a high-volume environment, caching pagelets with the user scope requires a separate web server cache of each pagelet for every user, which can result in significant memory problems. Because of this, caching pagelets with the role or application scope wherever possible is strongly recommended.</p> <ul style="list-style-type: none"> <li><i>role</i>: The portal maintains a single cache of the pagelet for all users in a role.</li> <li><i>application</i>: The portal maintains a single cached version of the pagelet for each user. The cache manager clones a copy of the meta pagelet and stores a copy for each user. For example, if 1000 users have signed in and they all have this pagelet on their homepage, there will be 1000 copies of the pagelet stored in cache. Each of these pagelet copies has the associated time to live (TTL) variable. Each time the system builds the users homepage, it reviews the TTL to see if it is time to refresh the pagelet. When a TTL time expires, that copy of the meta pagelet is invalidated and refreshed from the application server. As each user's copy expires their copy will be updated with the latest version.</li> </ul>
Interval	<p>Specify the number of seconds that should pass before the cache expires and is subject to an automatic refresh. This value depends on the frequency with which the data changes, and the need for data to be 100-percent accurate.</p> <p>For example, a company news article rarely changes once it's been published. You might want to set the refresh period for the article pagelet to be <i>86400</i> seconds so that it's cached for a 24-hour period.</p>

This example shows caching that is defined in an `IClientComponent` element with the role scope and an interval of 1800 seconds:

```
<IClientComponent Name="Header">
  <Cache Scope="role" Interval="1800">ANY</Cache>
  <Source Product="Portal">ICType=Script...</Source>
</IClientComponent>
```

---

**Note.** The `Cache` element must not be empty, but its content can be anything.

---

### Specifying Pagelet Caching in the Content Reference

You apply pagelet caching in a template by defining a special content reference attribute on the pagelet definition.

To implement caching for a pagelet:

1. In PeopleSoft Pure Internet Architecture, open the content reference administration page of the pagelet you want to cache.

Navigate to PeopleTools, Portal, Structure and Content, Portal Objects, Pagelets, select the folder containing the pagelet, and edit its content reference.

2. Add a new content reference attribute.

3. Specify the attribute name and label.
  - In the Name field, enter *PSCACHECONTROL*.
  - In the Label field, enter *PSCache-Control*.
4. Specify the attribute value with this syntax:

*cache\_scope,max-age=interval*

Parameter	Description
<i>cache_scope</i>	<p>Use one of these values:</p> <ul style="list-style-type: none"> <li>• <i>private</i>: The portal maintains the cached pagelet independently for each user.</li> </ul> <p><b>Note.</b> In a high-volume environment, caching pagelets with the private scope requires a separate web server cache of each pagelet for each user, which can result in significant memory usage. Because of this, caching pagelets with the role or public scope wherever possible is strongly recommended.</p> <ul style="list-style-type: none"> <li>• <i>role</i>: The portal maintains a single cache of the pagelet for all users in a role.</li> <li>• <i>public</i>: The portal maintains a single cache of the pagelet for all users who are using an application.</li> </ul>
<i>interval</i>	Specify the number of minutes that should pass before the cache expires and is subject to an automatic refresh.

For example, consider a PeopleSoft Enterprise Portal environment in which users can personalize their company news pagelet. The articles don't change frequently, and users don't need them to be up-to-the-minute; therefore, you might specify private caching with a three hour (180 minute) expiration:

*private,max-age=180*

5. Save the content reference.

The next time a user signs in and displays the pagelet, the caching that you specified takes effect.

### ***Managing Caching of Personalizable Pagelets***

You can make it possible for users to personalize their homepage pagelets. When a user personalizes a pagelet, the result should be visible only to that user, and therefore should be cached only for that user. To achieve this goal, as soon as the user attempts to personalize the pagelet, regardless of the caching scope that you originally specified, the portal automatically changes the caching scope to apply on a private basis and maintains the cached pagelet independently for that user.

---

**Important!** This automatic scope change applies only to the user who is personalizing the pagelet and to the portal on which the pagelet is personalized. On other portals, and for other users, the originally defined caching scope for that pagelet still applies. If the personalizations aren't stored as portal-specific data, then the personalizations for a pagelet that was originally defined with a wider caching scope can become visible to others who are accessing the same site through another portal.

Therefore, when developing personalizable pagelets, it's important to ensure that stored pagelet personalization metadata is keyed by both user ID and portal name, so that you can populate the pagelet independently for each portal. This prevents user personalizations on one portal from being cached and visible to a larger audience on another portal.

---

### ***Disabling Manual Pagelet Refresh***

When you implement pagelet caching, a refresh button appears by default in the pagelet header. You can prevent users from refreshing the pagelet manually by hiding the refresh button, which you do by using a check box on the pagelet's content reference administration page.

### **See Also**

[Chapter 5, "Administering Portals," Administering Content References, page 109](#)

[Chapter 6, "Administering Portal Homepages and Pagelets," Managing Pagelet Attributes, page 161](#)

[Chapter 9, "Using Portal Caching Features," Understanding Caching Scope, page 200](#)

*Enterprise PeopleTools 8.50 PeopleBook: Using PeopleSoft Applications*

*PeopleSoft Enterprise Portal 9.1 PeopleBook: Enterprise Portal Application Technology*

## **Implementing Metadata Caching**

Metadata caching is governed by portal-wide properties that control caching for these portal metadata objects as a group:

- The portal registry.
- Node definitions.
- Content references.
- Locally defined templates.

Portal metadata is likely to change far less frequently than pagelets or target content, so caching it on the server is an easy way to improve portal performance. Caching improves system performance by decreasing service requests from the web server to the application server.

---

**Note.** Metadata caching always applies with the application scope—the portal maintains a single metadata cache for all users who are using an application.

---

You implement metadata caching for a portal by setting these properties in the On the Web Server group box of the Web Profile Configuration - Caching page:

<b>Cache Portal Objects</b>	<p>Select to enable metadata caching.</p> <p>Metadata caching is enabled by default. You must restart the web server or wait for the stale interval to expire when you change:</p> <ul style="list-style-type: none"> <li>• The default portal for a database.</li> <li>• The default local node for a database.</li> <li>• The remote node to local or a local node to remote.</li> <li>• Templates that are associated with a content reference, node, or portal.</li> <li>• HTML within a template.</li> </ul> <hr/> <p><b>Note.</b> When you add or delete a content reference or folder, this change shows up in the menu structure immediately, because it's rendered within the application server, not the portal servlet.</p> <hr/>
<b>Cache Stale Interval</b>	<p>Specify the number of seconds that should pass before the portal refreshes the metadata cache.</p> <p>The default value of this property is <i>86400</i> (24 hours). With this setting, if you change the default local node in the database, the change doesn't take effect until the following day.</p>
<b>Cache Purge All Hit Count</b>	<p>Set to the maximum total number of HTTP requests that the web server should receive for objects in the metadata caches of all portals. When the web server receives this number of hits, it purges all metadata caches, forcing the portals to refresh them.</p> <p>This setting enables you to manage application server memory consumption. If the web server is consuming too much memory, you can lower this value. This causes the web server to reclaim the memory that is used for cached portal objects more often. The cost of this is increased CPU usage on both the web and the application server when the web server deletes the cached objects and then retrieves them again from the application server.</p> <p>The default value of this property is <i>1000</i>. Set to <i>-1</i> to disable the feature.</p>

---

**Note.** The timing properties—the cache stale interval and the cache purge all hit count—compete for effect. The first timing property to reach the specified interval or number of hits takes precedence. When the metadata cache on a portal is refreshed, the cache stale interval and cache purge all hit count counters for all portals on the same web server are reset.

---

### See Also

[Chapter 10, "Configuring the Portal Environment," Configuring Caching, page 235](#)

## Implementing Proxied JavaScript Caching

In addition to target content, pagelets, and metadata, you can cache proxied JavaScripts on the server. This is recommended for production systems; you might want to disable this caching during development.

You enable caching of proxied JavaScripts for a portal by selecting the Cache Proxied JavaScripts check box in the On the Web Server group box of the Web Profile Configuration - Caching page.

### **See Also**

Chapter 10, "Configuring the Portal Environment," Configuring Caching, page 235

---

## Administering Browser-Based Caching

This section provides an overview of browser-based caching and discusses how to:

- Implement homepage caching.
- Control navigation pagelet caching.
- Implement PeopleSoft page caching.

## Understanding Browser-Based Caching

A modern web browser can cache the information that it receives from a web server in several ways. Its preference settings determine how much storage space to use for caching, as well as the criteria by which it should refresh its cache.

### ***How the Portal Controls Browser-Based Caching***

If the browser is accessing a straight HTML page, it caches the page as a whole. If it's accessing a frameset, it caches the individual frames in the frameset, so that each frame's content can be refreshed independently.

Rather than depend on users to enable and set the timing criteria for caching, PeopleSoft portal technology takes advantage of a feature of HTTP. The portal web server can specify a caching interval in the HTTP response header of the HTML page or frame content that it delivers. The browser applies this value to determine when to stop using its cached copy of the data and request the most recent version from the server.

---

**Note.** Even when the browser requests a fresh copy of an HTML page or frame content, the portal web server controls whether the content it sends is cached data or fresh data, based on its own caching settings.

---

### ***When to Disable Browser-Based Caching***

The caching system increases system response performance in typical production environments, in which users access the portal web site by using computers that have been assigned specifically to them or computers that they own. However, some environments don't conform to this scenario.

You should consider disabling browser-based caching in these environments:

- Any environment where multiple users access the same computer.

These include kiosk-type environments, as well as corporate hoteling environments, where multiple users access the same machine. Users might access the same machine within minutes of each other, before the browser cache for the previous user times out. Security is enforced in this situation, but the new user may be confused by an inability to access certain items.

- A development or testing environment.

As content references are created or changed for a portal during development or testing, the new data might not appear to be immediately available due to caching.

- Any environment where you are using web server-based navigation caching.

### **See Also**

Chapter 10, "Configuring the Portal Environment," Configuring Caching, page 235

## **Implementing Homepage Caching**

In a PeopleSoft portal, a homepage is a combination of portal content elements that are assembled by the portal and delivered as a single HTML page, which is cached on each user's browser.

Based on the configuration that you specify, the portal enables or disables homepage caching by using the homepage's HTTP header. If caching is enabled, the header also conveys the time interval before the web server is accessed again to get a fresh homepage. The browser does not access the web server after it initially receives the homepage until the specified interval elapses. You can also specify whether a particular browser model and version should use or ignore the caching information in the header.

In any case, if a user clicks the browser's Refresh button, the homepage is accessed from the web server again, overwriting the homepage that is cached on the browser and restarting the interval counter. If any of the content is cached on the web server, the browser is refreshed from the server cache.

### ***Enabling Homepage Caching***

You enable homepage caching for a site by selecting the Cache Homepage check box in the On the Browser group box of the Web Profile Configuration - Caching page. When this check box is selected (the default value), two related page elements become editable:

- Homepage Stale Interval.
- Browsers.

### ***Specifying the Homepage Stale Interval***

Specify the homepage caching interval for a site by entering a value in seconds in the Homepage Stale Interval field in the On the Browser group box of the Web Profile Configuration - Caching page. When the specified interval elapses, the browser sends a new request for the homepage to the portal web server the next time a user accesses the homepage. The default value for this field is *1200*.

### ***Disabling Caching for Selected Browsers***

Because browsers don't all process HTML in exactly the same way, you might find it necessary to disable homepage caching for selected browser versions. This can be useful if you have one or two supported browsers and want to disable caching for nonstandard browsers that could pose an administration problem.

To disable homepage caching for a browser version, add a row to the Browsers grid of the Web Profile Configuration - Caching page, enter the browser's user agent ID, and select the Cache Home Page check box for that row. Homepage caching is enabled for all browser versions except those with this check box cleared, regardless of whether they're listed in the grid.

---

**Note.** As a convenience, several well-known browser types are included as example rows in the configuration. All of the examples have homepage caching enabled by default.

---

### **See Also**

Chapter 10, "Configuring the Portal Environment," Configuring Caching, page 235

## **Controlling Navigation Pagelet Caching**

When you use the portal menu to access target content other than your homepage, the portal delivers that content within a frameset. The portal header, menu, and target content regions are independent and can be managed separately by the browser. The menu region contains the portal's navigation pagelet, which can be cached on the browser with the user scope.

Without caching, every time the user clicks a menu folder, the browser requests a new copy of the navigation pagelet from the portal web server, which can significantly affect performance. With caching, the browser saves all of the menu states that are generated by user activity and can immediately restore them on demand.

Users manage navigation pagelet caching by using the Time page held in cache personalization option to specify a caching interval.

To specify a caching interval:

1. In PeopleSoft Pure Internet Architecture, select My Personalizations.
2. On the Personalizations page, click the Personalize Option button for general options.
3. Enter an override value in minutes for the Time page held in cache option and click OK.

The default value of this option is 900 minutes. To disable navigation pagelet caching, specify a value of 0 minutes. The maximum value can be 525600 minutes, which is one year.

---

**Note.** For the new setting to take effect, you must delete the browser cache.

With caching in effect, any changes in the menu structure while the user is signed in to the portal aren't available until the user signs out and back in, or until the caching interval elapses.

---

## Implementing PeopleSoft Page Caching

PeopleSoft pages are application pages that appear in the target content region of the portal frameset.

Being able to control PeopleSoft page caching on the browser is useful for situations in which PeopleSoft applications are deployed to kiosk workstations where multiple users access the applications. Disabling caching means that users can't click the browser's Back button to view another person's transaction or to view any other sensitive data.

### ***Enabling PeopleSoft Page Caching***

You enable PeopleSoft page caching for a site by selecting the Cache Generated HTML check box in the On the Browser group box of the Web Profile Configuration - Caching page. PeopleSoft page caching is enabled by default.

With PeopleSoft page caching disabled, when users click the Back button they receive a data missing message in Netscape Navigator and a page expired message in Microsoft Internet Explorer.

---

**Note.** The side effect of turning off caching completely is degraded performance. For each new page, the system calls the database. However, PeopleTools offers a compromise related to browser caching in the form of navigation pagelet caching, which doesn't expose any sensitive information.

---

See [Chapter 9, "Using Portal Caching Features," Controlling Navigation Pagelet Caching, page 209.](#)

### ***Specifying Supported States***

When the browser caches a PeopleSoft page, it stores the state of that page, including any unsaved changes. With the page state in the cache, users clicking the browser's Back button see the cached page in the state that it was in when they last viewed it. The browser caches as many page states as its own settings for storing temporary data allow.

The portal also maintains its own cache of page states for each browser window. You can specify how many of these states the portal should consider valid and support for further interaction. This setting, combined with the number of states the browser maintains, determines whether users can continue to work with previously accessed page states.

For example, if you specify that the portal should support two states, and the user clicks the Back button three times, any activity that requires refreshing the displayed page from the web server (such as field validation) fails, producing a page unavailable message in the browser.

You specify how many states the portal should support by entering a number in the Number of States Supported field in the On the Browser group box of the Web Profile Configuration - Caching page. The default value of this field is 5 states.

### ***See Also***

[Chapter 10, "Configuring the Portal Environment," Configuring Portal Security, page 225](#)

[Chapter 10, "Configuring the Portal Environment," Configuring Caching, page 235](#)

---

## Administering Web Server-Based Navigation Caching

This section provides an overview and discusses how to implement web server-based navigation caching.

### Understanding Web Server-Based Navigation Caching

Web server-based navigation caching considerably boosts performance by automatically storing cached data on the web server as users select items from the navigation menu. This form of caching saves unnecessary calls to the application server.

The portal maintains a single cache for all users with a given combination of roles. If your user population has limited role combinations, your caching memory requirement is significantly reduced.

See [Chapter 9, "Using Portal Caching Features," Understanding Caching Scope, page 200.](#)

### Implementing Web Server-Based Navigation Caching

Implement web server-based navigation caching by selecting the Cache Menu check box in the On the Web Server group box of the Web Profile Configuration - Caching page.

---

**Note.** This is the default setting for the DEV and TEST web profiles.

---

The cached data is stored in the following folder and cannot be moved:

\\PIA\_HOME\webserv\peoplesoft\applications\peoplesoft\PORTAL\psftcache

You must delete the cache periodically to keep the folder at a manageable size.

To prevent conflicts, disable browser-based caching and homepage caching.

To disable browser-based caching:

1. Select PeopleTools, Personalization, Personalization Options.
2. Select the *PeopleTools (PPTL)* option category level.
3. Remove the *METAXP* personalization definition.
4. Click Save.

To disable homepage caching:

1. Select PeopleTools, Web Profile, Web Profile Configuration.
2. Select the appropriate web profile.
3. Access the Browsing grid on the Caching page.
4. Clear the Cache Homepage check box for the appropriate browser.
5. Click Save.

**See Also**

Chapter 10, "Configuring the Portal Environment," Configuring Caching, page 235

## Chapter 10

# Configuring the Portal Environment

This chapter provides an overview of the authentication domain and discusses how to:

- Configure web profiles.
- Define portal nodes.
- Implement single signon functionality.
- Redirect pages with sensitive data.
- Override page encoding.
- Import menu groups into the portal registry.
- Improve same-server performance under Secure Sockets Layer (SSL).
- Use SSL accelerators.
- Use reverse proxy servers.
- Use firewalls with reverse proxy servers.
- Use reverse proxy servers with SSL accelerators.
- Apply load balancing.

---

**Important!** The PeopleSoft portal configuration environment has moved to a PeopleSoft Pure Internet Architecture interface, the web profile. Web profiles are documented in this chapter. A complete listing that maps PeopleTools release 8.43 and earlier portal settings to current web profile settings is documented at the end of this PeopleBook.

---

### See Also

[Chapter 10, "Configuring the Portal Environment," Configuring Web Profiles, page 216](#)

[Appendix C, "Understanding Changes in Portal Configuration Settings," page 547](#)

---

## Understanding the Authentication Domain

The portal *authentication domain* is the domain in which the portal is running and across which the single signon authentication token is valid. It's specified as a web server property and is used extensively throughout the PeopleSoft Pure Internet Architecture and portal runtime systems. An authentication domain is expressed as a string that completes the domain portion of an HTTP address, for example, .peoplesoft.com.

---

**Note.** The leading period is required. The correct string is, for example, .peoplesoft.com, and not peoplesoft.com.

---

The authentication domain supports the following functionality:

- Cross-frame JavaScript updates between the PeopleSoft Pure Internet Architecture and the portal.

Failure to set the authentication domain correctly for the portal and PeopleSoft Pure Internet Architecture applications causes JavaScript security errors to appear in the browser when PeopleSoft Pure Internet Architecture pages are accessed through a portal frame-based template. (The default template through which all PeopleSoft Pure Internet Architecture pages are displayed is frame-based.) The authentication domain must be set for both the portal web server and other PeopleSoft content web servers.

- PeopleCode global variable sharing between components on the homepage and components within a frame.

Failure to set the authentication domain correctly for the portal and PeopleSoft Pure Internet Architecture applications that use different web servers causes a new, incompatible session to be created on the PeopleSoft Pure Internet Architecture web server when the user accesses a PeopleSoft Pure Internet Architecture component through a frame-based template.

- Single signon functionality between PeopleSoft applications.

Failure to specify the authentication domain correctly prevents the PeopleSoft authentication cookie from being passed to the target PeopleSoft application and forces the target system to reauthenticate the user.

- Cookie sharing between the portal and third-party web applications.

If cookies need to be shared between web applications, then each web application must be accessed over a common domain name.

To share cookies, specify the authentication domain as the Cookies Passed to Server (forwarding domain ) property in the portal's web profile. You specify this property on the web Profile Configuration - Cookie Rules page.

See [Chapter 10, "Configuring the Portal Environment," Configuring Cookie Rules, page 234.](#)

### **Base-Level and Extended Authentication Domains**

You can define the portal authentication domain as a *base-level* authentication domain and as an *extended* authentication domain.

You define the base-level authentication domain during the PeopleSoft Pure Internet Architecture setup. This domain is stored as part of your web server configuration. It enables PeopleCode global variable sharing, which is required for initial access to the portal. The portal uses the base-level domain if you don't define an extended authentication domain.

---

**Important!** You must supply a base-level authentication domain at setup for every PeopleSoft application with which the portal interacts. This value is stored on each application's web server and must be identical for each application.

---

See the PeopleTools 8.50 installation guide for your database platform.

You can define an optional extended authentication domain in your portal's web profile. An extended authentication domain overrides, but must be compatible with, the base-level authentication domain. For example, if you entered *.customer.com* during the PeopleSoft Pure Internet Architecture setup, only values such as *.enterprise.customer.com* and *.individual.customer.com* are valid.

---

**Note.** If you defined a base-level or extended authentication domain, you must use it in all URLs that you specify in your portal. For example, if your authentication domain is *.mydomain.com*, then instead of using the URL *http://mymachine:8080/pshome/signon.html*, you must use the URL *http://mymachine.mydomain.com:8080/pshome/signon.html*.

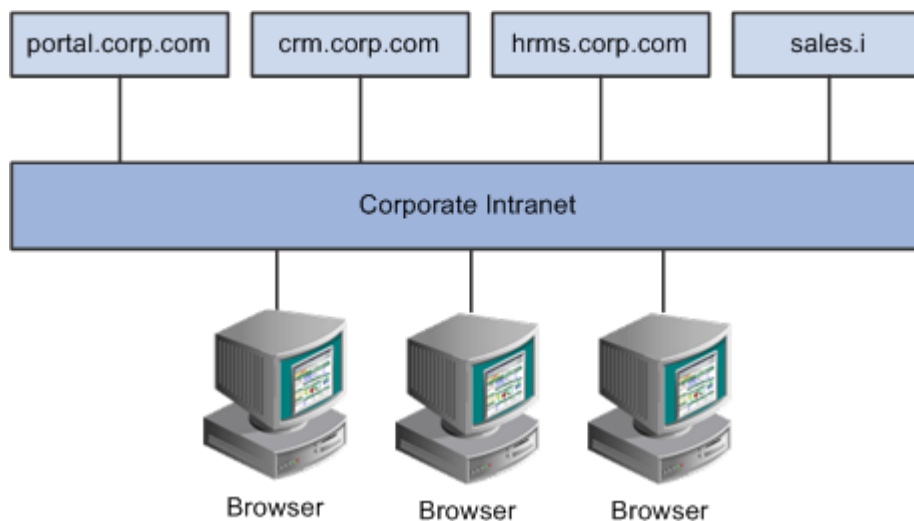
---

You specify the extended authentication domain on the Web Profile Configuration - General page.

See [Chapter 10, "Configuring the Portal Environment," Configuring General Portal Properties, page 220.](#)

### ***An Example of Multiple Applications on a Portal***

In the following example, the CRM and HRMS web profiles need to be defined with Domain Name Server (DNS) names that include the same authentication domain as the DNS name of the portal web server. They also each need the Authentication Domain property in their web profiles set to this value.



Example of the portal interacting with several different PeopleSoft applications

Web servers that don't have the same server domain as the portal (such as *sales.i*) can still be used to serve content to the portal. However, cookies set by the portal are not forwarded to these servers. The *sales.i* server in the example can provide pages and applications to the portal, but it cannot host a PeopleSoft application that supports single signon functionality with the portal.

## See Also

*Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*, "Working with Oracle WebLogic"

*Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*, "Working with IBM WebSphere"

No link destination found for g002868c23e2ff96d\_ef90c\_1109e9d28da\_\_7598

---

## Configuring Web Profiles

This section provides an overview of web profiles and discusses how to:

- Configure general portal properties.
- Configure portal security.
- Configure virtual addressing.
- Configure cookie rules.
- Configure caching.
- Configure trace and debug options.
- Configure look and feel.
- Configure custom properties.
- Viewing web profile histories.

## Understanding Web Profiles

A *web profile* is a named group of configuration property settings that the portal applies throughout your PeopleSoft system to control all portal-related behavior. The following web profiles are predefined and delivered with the PeopleTools base portal:

- DEV

The DEV web profile provides basic portal functionality for development, including trace and debug settings that are appropriate for development.

---

**Important!** Never run your production system with the DEV web profile. This profile is for development environments and is not tuned for performance.

---

- TEST

The TEST web profile uses the same settings as the DEV web profile, except that fewer trace and debug properties are enabled.

- **PROD**

The PROD web profile uses the settings that are most commonly needed in a production environment that authenticates users.

- **KIOSK**

The KIOSK web profile uses the same settings as the PROD web profile, except that public user access is enabled for the Guest user, and all options for storing caching or persistent cookies on the browser are disabled.

Additional web profiles may be delivered as part of PeopleSoft Enterprise Portal or an application portal pack. You can modify the property settings of any web profile by using the Web Profile Configuration component (WEB\_PROFILE), which includes pages for these types of properties:

- General
- Portal security
- Virtual addressing
- Cookie rule
- Caching
- Debug options
- Look and feel

### ***Specifying an Initial Web Profile***

As you perform the PeopleSoft Pure Internet Architecture setup procedure for your system, you're prompted to specify which web profile to apply to your portal. You're also prompted for an existing user ID and password. This information is saved in encrypted form where the portal servlet can use it to gain secure access to the web profile.

The default web profile is DEV.

The default web profile user ID and password are both PTWEBSEVER.

The PTWEBSEVER account provides the portal servlet with minimal security access, sufficient only to launch the portal environment, but without access to any pages or other PeopleSoft Pure Internet Architecture objects. This account uses the PTPT1500 permission list, which is set to never time out, and provides necessary access for 24 hours a day, seven days a week.

---

**Note.** Your PeopleSoft application might be delivered with the account that you specify here locked. You must unlock it before you can access and configure the web profile, or you receive an error message about incorrect site configuration on the sign-in page.

You can unlock the web profile account by clearing the Account Locked Out check box on the User Profile - General page. You can alternatively issue the following SQL command against your database (this example uses the PTWEBSEVER account):

```
update PSOPRDEFN set ACCTLOCK=0 where OPRID='PTWEBSEVER'
```

---

See *Enterprise PeopleTools 8.50 PeopleBook: Security Administration*, "Administering User Profiles," Setting General User Profile Attributes.

You can specify any of the delivered profiles, or you can enter a different profile name. In that case, the portal is set up to use a profile by that name. If a profile by the name that you specify doesn't exist in the database, internal default settings are used until you sign in to the portal and create a profile with that name. The portal then automatically configures itself according to those settings.

As long as there's no profile by the name that you specified during the PeopleSoft Pure Internet Architecture setup, the internal default settings remain in effect, which is indicated on the site's sign-in page. When the site is in this mode, every browser request triggers an attempt to load the named profile. Therefore, you shouldn't use the site for extended periods like this.

Once you complete the PeopleSoft Pure Internet Architecture setup, you can then use the Web Profile Configuration component to modify the properties to reflect your settings.

---

**Note.** The default internal settings are not the same as the DEV profile. Don't modify any of the delivered profiles, so that you always have access to unmodified reference versions. You can make a copy of any profile and modify it, or you can define a new profile instead.

---

### ***Copying and Deleting Web Profiles***

To copy a web profile:

1. Select PeopleTools, Web Profile, Copy Web Profile.
2. Enter or select a web profile name.
3. On the Web Profile Save As page enter a new name for the web profile in the To edit box.

To delete a web profile:

1. Select PeopleTools, Web Profile, Delete Web Profile.
2. Enter or select a web profile name.
3. On the Delete Web Profile page, click Delete Web Profile.

### ***Changing the Web Profile After the PeopleSoft Pure Internet Architecture Setup***

After the PeopleSoft Pure Internet Architecture setup, to select a different web profile, you must edit the text file that stores this information, called `configuration.properties`. Each PeopleSoft portal that you set up has its own copy of `configuration.properties`, located in this directory:

`PIA_HOME\webserv\peoplesoft\applications\peoplesoft\PORTAL\WEB-INF\psftdocs\sitename`

The `configuration.properties` file includes these properties:

- WebProfile

Change the value of this property to the name of the web profile that you want to apply to the portal, for example, `WebProfile=MYPROFILE`.

- WebUserId and WebPassword

You can specify a different user ID and password for the portal servlet to use to access the web profile, but you must encrypt the password by using a Java encryption utility that you launch from a provided script called PSCipher.

To encrypt the user ID or password:

1. At a command prompt, change to the location of the encryption script file:

```
PIA_HOME\webserv\peoplesoft
```

2. On a UNIX machine, change the PSCipher.sh script file permissions so you can run it.
3. Run the script file with your password as an argument. For example:

```
pscipher MYPASSWORD
```

The utility returns the encrypted password as a string. For example:

```
VOBN5KcQZMg=
```

4. Copy the encrypted string and paste it into the configuration.properties file, for example:

```
WebPassword=VOBN5KcQZMg=
```

---

**Important!** Make sure that the entire encrypted string, (including all symbol characters), is copied.

---

After you save the configuration.properties file, restart your web server and the new profile takes effect.

### See Also

*Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Integration Broker Administration*, "Managing Integration Gateways," Using the integrationGateway.properties File

## Pages Used to Configure Web Profiles

Page Name	Definition Name	Navigation	Usage
Web Profile Configuration - General	WEB_PROF_GENERAL	PeopleTools, Web Profile, Web Profile Configuration, General	Configure general portal properties.
Web Profile Configuration - Security	WEB_PROF_SECURITY	PeopleTools, Web Profile, Web Profile Configuration, General, Security	Configure portal security.
Web Profile Configuration - Virtual Addressing	WEB_PROF_PROXIES	PeopleTools, Web Profile, Web Profile Configuration, General, Virtual Addressing	Configure transfer protocols for proxy servers.

<b>Page Name</b>	<b>Definition Name</b>	<b>Navigation</b>	<b>Usage</b>
Web Profile Configuration - Cookie Rules	WEB_PROF_COOKIES	PeopleTools, Web Profile, Web Profile Configuration, General, Cookie Rules	Define rules that determine how the portal passes cookies to servers in the same domain.
Web Profile Configuration - Caching	WEB_PROF_CACHING	PeopleTools, Web Profile, Web Profile Configuration, General, Caching	Configure caching rules for browsers and web servers.
Web Profile Configuration - Debugging	WEB_PROF_DEBUGGING	PeopleTools, Web Profile, Web Profile Configuration, General, Debugging	Determine the type of information to include in logs.
Web Profile Configuration - Look and Feel	WEB_PROF_LOOKFEEL	PeopleTools, Web Profile, Web Profile Configuration, General, Look and Feel	Determine the pages to be used in situations such as start, errors, expiration, signon, and language support.
Web Profile Configuration - Custom Properties	WEB_PROF_CUST_PROP	PeopleTools, Web Profile, Web Profile Configuration, General, Custom Properties	Configure properties that have been added since the current release of PeopleTools was shipped or that are needed only for backward compatibility.
Web Profile History	WEB_PROF_HISTORY	<ul style="list-style-type: none"> <li>• PeopleTools, Web Profile, Web Profile History</li> <li>• PeopleTools, Web Profile, Web Profile Configuration, General Click the View History link on the Web Profile - General page.</li> </ul>	Review the current portal attributes of a web server, website, and web profile.

## Configuring General Portal Properties

Access the Web Profile Configuration - General page (PeopleTools, Web Profile, Web Profile Configuration).

General	Security	Virtual Addressing	Cookie Rules	Caching	Debugging	Look and Feel	D
Profile Name:	PROD		<a href="#">Save As ...</a>		<a href="#">View History</a>		
Description:	<input type="text" value="Installation Defaults"/>						
Authentication Domain:	<input type="text"/>						?
Help URL:	<input type="text"/>						?
<input checked="" type="checkbox"/> Compress Responses ?							
<input checked="" type="checkbox"/> Compress Response References ?							
Compress Mime Types:		<input type="text" value="application/x-javascript,text/javascript,text/css,text/html"/>					?
<input checked="" type="checkbox"/> Compress Query ?							
Save Confirmation Display Time:	<input type="text" value="3,000"/>	Milliseconds		?			
<input checked="" type="checkbox"/> Enable Processing Message ?							
<input checked="" type="checkbox"/> Enable New Window ?							
<input checked="" type="checkbox"/> Enable PPM Agent ?							
PPM Monitor Buffer Size:		<input type="text" value="51,200"/>	KB		?		
<input type="checkbox"/> Single Thread Netscape ?							
Single Thread Delay:		<input type="text" value="1,000"/>	Milliseconds		?		
Non-standard Base Path:	<input type="text"/>						?
<b>Reports</b>							
<input checked="" type="checkbox"/> Enable Report Repository ?							
Report Repository Path:	<input type="text"/>						?
<b>Compress Report Output</b>							
<input checked="" type="radio"/> All Browsers ?							
<input type="radio"/> Exclude NetScape ?							
<input type="radio"/> Do Not Compress ?							

### Web Profile Configuration - General page

#### Save As

Click to save a copy of the current web profile under a new name. On the Save Web Profile As page, enter a new profile name and click OK.

Use to modify one of the profiles that is delivered with your PeopleSoft application.

#### View History

Click to access the Web Profile History search page in a new browser window.

**Authentication Domain** Enter the name of the extended authentication domain in which the portal is running, starting with a leading period. This value overrides, but must be compatible with, the base-level authentication domain. For example, if you entered *.customer.com* during the PeopleSoft Pure Internet Architecture setup, only values such as *.enterprise.customer.com* and *.individual.customer.com* are valid.

An authentication domain is required for a variety of portal functions. For example, if a cookie is shared on web servers *foo.peoplesoft.com* and *bar.peoplesoft.com*, you must specify an authentication domain of *.peoplesoft.com*.

This field requires a value if you specified a default addressing server name on the Web Profile Configuration - Virtual Addressing page. You must qualify that server name with this domain name.

The default value of the authentication domain is the one that you specified during the PeopleSoft Pure Internet Architecture setup. That value is stored as the web server's session cookie domain and kept in an internal web server file.

For Oracle WebLogic, that file is *weblogic.xml*. The preferred method for changing that value is to rerun the website setup.

For WebSphere, that file is *deployment.xml*. This is located under *PIA\_HOME\websrv\<profile-name>\config\cells\<profile-name>NodeCell\applications\<application.name>.ear\deployments\<application-name>\deployment.xml*

---

**Note.** The value that you enter in this field is automatically forced to lowercase.

---

## Help URL

Specify the URL that is needed to link to the correct location in your HTML PeopleBooks. When users click the Help button, the appropriate context-sensitive PeopleSoft documentation should appear. To remove the help link, leave this value blank, and users won't see a Help link on the application page.

Construct the URL like this:

```
http://helpwebserver:port/productline/f1search.htm?
ContextID=%CONTEXT_ID%&LangCD=%LANG_CD%
```

For example:

```
http://myhelpwebserver:8080/html/doc/f1search.htm?ContextID=%CONTEXT_I
D%&LangCD=%LANG_CD%
```

---

**Note.** This setting applies only to browser access. It does not apply to users connecting in the Microsoft Windows environment by using PeopleSoft Application Designer and other development tools. You can find information about configuring Microsoft Windows-based context sensitive help in the PeopleTools installation documentation.

---

## Compress Responses

Select to enable compression in the communication between the web server and the browser. Gzip and Compress protocols are supported.

This check box is selected by default.

<b>Compress Response References</b>	<p>Select to enable compression of cache files that are delivered from the web server to the user's browser. Only cache files with the Multipurpose Internet Mail Extensions (MIME) types that are specified in the Compress MIME Types text box are compressed. Gzip and Compress protocols are supported.</p> <p>This check box is cleared by default.</p>
<b>Compress MIME Types</b>	<p>Specify the MIME types of the cache files to be compressed as a comma-separated string. This field is available only if you selected the Compress Response References check box.</p> <p>The default value is <i>application/x-javascript,text/javascript,text/css,text/html</i>.</p>
<b>Compress Query</b>	<p>This property applies to browser requests in which the content type section of the URL is /q/, indicating a query. Select to enable compression of query responses to the browser.</p> <p>This check box is selected by default.</p>
<b>Save Confirmation Display Time</b>	<p>Specify in milliseconds how long the save_confirm image should appear for the user if the save confirm personalization option is enabled.</p> <p>The default value is 3000 milliseconds (3 seconds).</p>
<b>Enable Processing Message</b>	<p>Select to enable processing notification while the system processes a request.</p> <p>This check box is selected by default.</p>
<b>Enable New Window</b>	<p>Select to enable the New Window link that users can click to open a new application window.</p> <p>This check box is selected by default.</p>
<b>Enable PPM Agent</b> (enable PeopleSoft Performance Monitor agent)	<p>Select to activate the PeopleSoft Performance Monitor agent for this portal. The performance monitoring system has <i>agents</i> and <i>monitors</i>. An agent captures units of data (PMUs), and a monitor views and analyzes agent flows. This check box controls whether the agent is active on sites that use this profile.</p> <p>This check box is selected by default.</p>
<b>PPM Monitor Buffer Size</b> (PeopleSoft Performance Monitor buffer size)	<p>Specify the maximum size, in kilobytes (KB), of the Monitor Server buffer. Data is discarded once this limit is reached. This field is available only when Enable PPM Agent is selected.</p> <p>The default value is 51200 KB (50 megabytes).</p>
<b>Single Thread Netscape</b>	<p>Select to indicate that requests from a Netscape browser should be single-threaded to prevent crashes.</p> <p>Older versions of Netscape browsers had problems with multithreading responses and would crash. Select this check box if your users are using older versions of Netscape browsers and are experiencing browser crashes.</p> <p>This check box is cleared by default. When selected, the Single Thread Delay option appears.</p>

**Single Thread Delay** Specify a delay, in milliseconds, for single-threaded Netscape requests. This field is available only when Single Thread Netscape is selected.

The default value is *1000* milliseconds (one second).

**Non-standard Base Path** In the rare circumstance that you receive an error message that the base physical path has not been set, you must enter the location of the directory that contains the *signon.html* file for your application.

This field is blank by default.

See [Chapter 10, "Configuring the Portal Environment," Viewing Web Profile Histories, page 253](#); [Chapter 10, "Configuring the Portal Environment," Understanding the Authentication Domain, page 214](#); [Chapter 10, "Configuring the Portal Environment," Configuring Caching, page 235](#); [Chapter 10, "Configuring the Portal Environment," Configuring Virtual Addressing, page 232](#) and *Enterprise PeopleTools 8.50 PeopleBook: Performance Monitor*, "Administering the Performance Monitor," Working with Performance Monitor Web Profile Properties.

See the PeopleTools 8.50 installation guide for your database platform.

## Reports

**Enable Report Repository** Select to enable users to view the report repository. When this check box is cleared, no report files can be viewed by users.

This check box is selected by default.

**Report Repository Path** (Optional) Specify a drive and directory path to indicate where the site that is served by this web profile should look for generated reports.

This value overrides the location that is specified during the PeopleSoft Pure Internet Architecture setup; you can leave this field blank to use the original value.

---

**Note.** Changing the report repository path prevents you from viewing existing reports, unless the old subdirectory structure is moved to the new location. Be careful when editing this field.

---



---

**Note.** (For Windows only) If the report repository resides on a different machine than the web server, specify the report repository path in UNC format instead of mapped drive format. For example, if the report repository is on MachineX in folderY, the Report Repository Path should be specified as:

//MachineX/folderY/psreports

or

\\\\MachineX\\folderY\\psreports

If MachineX/folderY is mapped to the Y: drive, do not specify the Report Repository Path as Y:\psreports.

---

**Compress Report Output**

Specify how reports should be compressed:

*All Browsers:* Reports are always compressed. This is the default value.

*Exclude Netscape:* Reports are compressed, except for users who are using Netscape Navigator.

*Do Not Compress:* Reports are never compressed.

**See Also**

*Enterprise PeopleTools 8.50 PeopleBook: PeopleCode API Reference*, "Internet Script Classes (iScript)," AuthTokenDomain

*Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*, "Working with Oracle WebLogic"

*Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*, "Working with IBM WebSphere"

No link destination found for g002868c23e2ff96d\_ef90c\_1109e9d28da\_\_7598

## Configuring Portal Security

Access the Web Profile Configuration - Security page (PeopleTools, Web Profile, Web Profile Configuration).

General	<b>Security</b>	Virtual Addressing	Cookie Rules	Caching	Debugging	Look and Feel	D
---------	-----------------	--------------------	--------------	---------	-----------	---------------	---

Profile Name: PROD

Days to Auto Fill User ID:  ?

☐ PIA use HTTP Same Server ?

☒ Allow Unregistered Content ?

View File Time to Live:  Seconds ?

**SSL**
☐ Secured Access Only ?
 ☒ Secure Cookie with SSL ?

**Authenticated Users**

Inactivity Warning:  Seconds ?

Inactivity Logout:  Seconds ?

HTTP Session Inactivity:  Seconds ?

Timeout Warning Script: WEBLIB\_TIMEOUT.PT\_TIMEOUTWARNING.FieldFormula.IScript\_TIMEOUTWARNING Override ?

**Public Users**

☐ Allow Public Access ?

User ID:  ?

Password:  ?

HTTP Session Inactivity:  Seconds ?

**Web Server Jolt Settings**

Disconnect Timeout:  Seconds ?

Send Timeout:  Seconds ?

Receive Timeout:  Seconds ?

**XML Link**

User ID:  ?

Password:  ?

☒ XML Link Use HTTP Same Server ?

### Web Profile Configuration - Security page

**Days to Autofill User ID** If enabled, the system caches the user ID and automatically inserts it on the sign-in page. This a convenience for users. This feature is implemented through the use of a stored cookie on the browser.

The default value is 7 days. Specify a value of 0 to disable this feature. Use 0 in a public area or kiosk situation, or if your security policy doesn't allow cookies stored on the browser.

**View File Time to Live** Specify, in seconds, how long the portal should wait after sending a file attachment to a user's browser before removing that file from the web server storage.

If this value is 0, then the amount of time the file remains on the web server is always greater than or equal to one minute and the specific amount of time beyond one minute is dependent upon the size of the file.

If the value is *greater than 0*, then the amount of time the file remains on the web server is only dependent upon the value of that property in seconds.

The default value is 0.

**PIA use HTTP Same Server** (PeopleSoft Pure Internet Architecture use HTTP same server)

Select to indicate that the portal should use the HTTP protocol instead of HTTPS for requests that are issued by the portal for content that is hosted on the same server as the portal servlet.

Setting this property is necessary when the portal web server is behind an SSL accelerator or when SSL is terminated on a device in front of the portal web server, such as a reverse proxy server, or between different sites that are on the same web server.

You can also use this property to improve the performance of homepage pagelets that are provided by the PeopleSoft Pure Internet Architecture servlet that run on the same web server as the portal, and from which the web server receives SSL requests. That is, SSL has not been terminated by a device in front of the web server.

You must also specify the default addressing protocol and port on the Web Profile Configuration - Virtual Addressing page.

See [Chapter 10, "Configuring the Portal Environment," Using SSL Accelerators, page 265](#).

**Allow Unregistered Content**

Select to instruct the portal to serve both registered and unregistered content.

The portal generally allows any external content links to be wrapped with the portal header and navigation frames. Turning this option off will prevent explicitly registered content references from being displayed in the portal.

This check box is selected by default.

---

**Note.** If Allow Unregistered Content is not selected, the portal won't wrap a content link if it is not registered in the portal registry or if it isn't coming from a registered node. The portal checks if the content link is from a registered node by checking if the content link starts with the URI specified in the Content URI Text or the Portal URI Text in the Node Definition page, on the Portal tab. For security purposes, the portal only checks against the Content URI Text and Portal URI Text if it contains text longer than twelve characters.

---

**New Session Upon Signon**

Select to make a successful signon always sign on to a new web server session. This prevents a hacker from planning a fixed session for a particular user and the user signing into to the fixed session. By default, the option is selected.

**SSL**

**Secured Access Only**

Select to enforce SSL if the entire website requires the SSL protocol. This prevents users from using non-SSL protocols to access any link within this website or application.

If only some pages require SSL access, clear this check box (the default setting).

**Secure Cookie with SSL** Select to prevent the single signon token from traveling over an insecure network. If you select this check box and the scheme of the current request is HTTPS (an SSL server), the system sets the secure attribute of the single signon cookie (PS\_TOKEN) to True.

PeopleSoft single signon functionality also applies at the web server level. For example, suppose that you have two web servers: server X and server Y. Web server X is an SSL site, and web server Y is not. In these situations, many sites want server Y to trust the authentication token, PS\_TOKEN, issued by server X. This requires that you set the PS\_TOKEN to be secure.

If the PS\_TOKEN is not secure, when a user signs in through server Y, the browser sends PS\_TOKEN to server Y over the unencrypted, non-SSL link. This is typical behavior for browsers when dealing with cookies that aren't secure. Potentially, in this situation, a hacker could obtain this token from the clear network and use it to sign in to the SSL-secure server X.

Another important use of this property relates specifically to the PeopleSoft portal. When the portal proxies content with an HTML template, it should forward only PS\_TOKEN cookies that are marked secure over SSL connections.

---

**Note.** By selecting this check box, you effectively disable single signon functionality with any non-SSL servers.

---

If, at your site, you want users to sign in to an HTTPS server, and then want to use single signon functionality with HTTP servers, clear this check box to enable single signon functionality between HTTPS and HTTP servers.

---

**Note.** Before you clear this check box, make sure that you are aware of all the security implications, such as the security of the HTTPS server being compromised.

---

This check box is selected by default.

## ***Authenticated Users***

### **Inactivity Warning**

Specify the number of seconds that the portal should wait before warning users that the browser session will expire. Users can continue with the current session by clicking the OK button in the warning message, which resets the inactivity timer. If a user doesn't respond within the inactivity logout interval, the session ends and the expired connection page appears.

The default value is *1080* seconds (18 minutes).























<b>JavaScript Web Directory</b>	Enter the JavaScript web cache directory. The default value is <i>/cache</i> .
---------------------------------	---

Configuring Trace and Debug Options

Access the Web Profile Configuration - Debugging page (PeopleTools, Web Profile, Web Profile Configuration).

Virtual Addressing

Cookie Rules

Caching

Debugging

Look and Feel

Custom Properties

Profile Name: PROD

☐

Trace Monitoring Server ?

☐

Trace PPM Agent ?☐☐☐☐☐☐☐☒☐

Web Profile Configuration - Debugging page

<b>Trace Monitoring Server</b>	Select to have the PeopleSoft Performance Monitor and PPMI servlets write debug information to the web server log. This check box is cleared by default. <hr/> <b>Note.</b> This feature is not an agent trace. For an agent trace use Trace PPM Agent. <hr/>
<b>Trace PPM Agent</b>	Select to enable PeopleSoft Performance Monitor tracing on performance agents. This check box is cleared by default.

<b>Show Connection Information</b>	<p>Select to have the application generate a system information page, which includes the browser, OS, PeopleTools release, application release, service pack, page definition name, component definition name, menu definition name, user ID, database name, database type, and application server address when you press Ctrl+J.</p> <hr/> <p><b>Note.</b> This information is useful for orientation and troubleshooting purposes, but might not be suitable for end users.</p> <hr/>
<b>Show Trace Link at Signon</b>	Select to display a URL link at sign-in. The link opens a page for setting trace parameters.
<b>Show Layout</b>	Select to apply border and color attributes in a table layout for pages. This enables developers to see the position of PeopleSoft Application Designer objects in HTML.
<b>Show Overlapping Fields</b>	Select to include comments in generated HTML pages that may help in diagnosing page layout problems, such as fields overlapping other fields.
<b>Show StyleSheet Inline HTML</b>	Select to insert the page's stylesheet into its generated HTML.
<b>Show JavaScript Inline HTML</b>	Select to display all the JavaScript functions that are used for processing in the generated HTML page.
<b>Generate HTML for Testing</b>	<p>Select to alter the generated HTML to assist with testing and troubleshooting. For example, this option provides additional white space and comments on the page to aid readability. Also, it includes additional name attributes for reference from SQA robot scripts.</p> <hr/> <p><b>Note.</b> Selecting this option may cause some pages or pagelets not to display correctly. You can also set this in the application server configuration file, with the TracePIA option. If you want to ensure that all pages display correctly, you need to verify that TracePIA isn't enabled in the configuration file.</p> <hr/> <p>You can also set this in the application server configuration file. If you want to ensure that you've</p>
<b>Write Dump File</b>	Select to have the system write a log file to the web server if a Oracle Jolt exception error occurs.

**Create File from PIA HTML Page** (create file from PeopleSoft Pure Internet Architecture HTML page)

Select to view and debug the source HTML that the application server generates. The system saves each generated page as *PS\_CFG\_HOME\appserv\domain\LOGS\client\element\N.html*.

The variables in the name are:

- *Domain*: The name of the application server domain.
- *Client*: The name of the machine or Internet Protocol (IP) address where the browser is running.
- *Element*: The name of the process or object that generates the HTML (query name for query, program name for iScripts, and so on).
- *N*: The state number for the generated page.

---

**Warning!** Use this tracing feature only for troubleshooting and testing. Enabling this feature generates numerous directories and files on the application server, which significantly affects performance. Use it for short periods, preferably only for a single-user test scenario. Never enable it on a production website.

---

See [Chapter 10, "Configuring the Portal Environment," Configuring Custom Properties, page 249](#) and *Enterprise PeopleTools 8.50 PeopleBook: Performance Monitor*, "Administering the Performance Monitor," Working with Performance Monitor Web Profile Properties.

## Configuring Look and Feel

Access the Web Profile Configuration - Look and Feel page (PeopleTools, Web Profile, Web Profile Configuration).





















<b>Configuration</b>	<b>User Session</b>	<b>User Action</b>	<b>System Behavior</b>
When the public user access (default signon) is disabled and the custom web property "noDefaultSignonForWorkflow" is set to false/true.	Any	Any	The system behaves as configured.

---

**Note.** If the public user does not have workflow privileges, then close the existing public user session (browser) before clicking the workflow link.

---

See *Enterprise PeopleTools 8.50 PeopleBook: Security Administration*, "Implementing Single Signon" and *Enterprise PeopleTools 8.50 PeopleBook: Performance Monitor*, "Administering the Performance Monitor," Viewing Monitor Servlet Diagnostics.

## Viewing Web Profile Histories

Access the Web Profile History page.



## Understanding Portal Nodes

You can create and maintain a node definition in the Node Definitions component (IB\_NODE). A node is defined once but can be used for multiple purposes. Several pages in the Node Definitions component are used purely for integration purposes by PeopleSoft Integration Broker. For portal purposes, define a portal by using the Node Definitions and Portal pages.

Nodes must be blank for external content references.

---

**Note.** You cannot delete a node that is in use by a content reference when the Save button is clicked. If the node is in use, the node name is not deleted.

---

## Pages Used to Define Portal Nodes

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Portal	IB_NODEURI	PeopleTools, Portal, Node Definitions, Node Definitions, Portal	Define portal content information for the selected node definition.
Node Definitions	IB_NODE	PeopleTools, Portal, Node Definitions, Node Node Definitions	Specify the node type to designate the URL format for the selected node definition.

## Setting Portal Nodes

Access the Portal page. (PeopleTools, Portal, Node Definitions.)

Node Definitions

Connectors

Portal

WS Security

Node Name PT\_LOCAL

Details

Description PT\_LOCAL

☐ Local Node

Tools Release

8.50.00.03

Application Release

Financials/SCM 9.10

Content URI Text

Example: http://someserver/psc/pshome/

Portal URI Text

Example: http://someserver/psp/pshome/

Save

Portal page

Node Name	Displays the node name, description, and local node check box status from the Node Info page.
Tools Release	Enter the version of PeopleTools that is running. This is a free form field. You can enter any information that you like, however, it is recommended that you use the same Tools release that is available when you press Ctrl + J.
Application Release	Enter the version of the PeopleSoft application that you are running. This is a free form field. You can enter any information that you like, however, it is recommended that you use the same Application name release that are available when you press Ctrl + J.
Content URI Text (content uniform resource identifier text) and Portal URI Text (portal uniform resource identifier text)	<div>Enter the uniform resource identifier (URI) of the pscontent (iclient) servlet for the node and the URI text for the portal. These values must be unique when a message node is created or updated.</div> <div><b>Important!</b> The server name and port in these URIs must not exceed 30 characters in length. For example, <i>http://mybiglongservername.peoplesoft.com:8080</i>, not including the <i>http://</i> prefix, is 39 characters—9 characters too long.</div> <div><b>Note.</b> The value that you enter in these fields is automatically forced to lowercase.</div>





































## **Part 4**

# **Working with Pagelets**

### **Chapter 11**

#### **Using Pagelet Wizard**

### **Chapter 12**

#### **Building Pagelets**

### **Chapter 13**

#### **Working with Related Content Services**

### **Chapter 14**

#### **Configuring OBIEE Report Pagelets**



## Chapter 11

# Using Pagelet Wizard

This chapter provides overviews of Pagelet Wizard and its data sources, display formats, display sizes, and pagelet types. This chapter discusses how to:

- Define Pagelet Wizard transformation types, display formats, and data types.
- Define Pagelet Wizard XSL prototypes.
- Define Pagelet Wizard headers and footers.
- Define Pagelet Wizard pagelet categories.
- Enter pagelet identifying information (step 1).
- Select a pagelet data source (step 2).
- Specify pagelet data source parameters (step 3).
- Select a pagelet display format (step 4).
- Specify pagelet display options (step 5).
- Specify pagelet publication options (step 6).
- Use Pagelet Wizard post-transformation processing.

---

## Overview of Pagelet Wizard

Pagelet Wizard pagelets are small pages that provide display-only snapshots of applicable content. Pagelet Wizard provides a user-friendly, browser-based graphical user interface (GUI) that leads you through the series of steps involved in creating and publishing a pagelet.

Portal administrators and nontechnical users can use Pagelet Wizard to integrate and transform data from a variety of data sources.

You do not need to have PeopleSoft-specific application development tools or skills to use Pagelet Wizard to create pagelets. During the pagelet-creation process, the Pagelet Wizard GUI presents a series of numbered steps. Each step appears in a numbered path at the top of the Pagelet Wizard component to indicate where you are in the pagelet creation process.



<b>HTML</b>	Use to present appropriately sized static HTML pages in Pagelet Wizard pagelets. HTML data source pagelets are compatible with the passthru and custom display formats.
<b>IB Connector</b>	Use to interact directly with connectors registered to the Integration Broker and secured using Define IB Connector Security within Pagelet Wizard. This data type should only be used when it is not desirable to use the Integration Broker data type.
<b>Integration Broker</b>	Use for pagelets that utilize Integration Broker to retrieve data from external sources. This middleware technology facilitates synchronous and asynchronous messaging among internal systems and with trading partners, while managing message structure, message format, and transport disparities.
<b>Navigation Collections</b>	Use to create pagelets from existing sets of folders and links stored in Navigation Collections. Navigation Collection data source pagelets are compatible with the menu and custom display formats.
<b>OBIEE Report</b>	Use to create pagelets from existing reports on the OBIEE server.
<b>PeopleSoft Query</b>	Use to access PeopleSoft application data or any tables within the PeopleSoft environment through Pagelet Wizard pagelets. PeopleSoft Query data source pagelets are compatible with the table, list, chart, and custom display formats.
<b>Rowset</b>	<p>Use to access a rowset record whose data is populated at runtime. This data source is geared toward supplying data to embedded pagelets. Rowset data source pagelets are compatible with the table, list, chart, and custom display formats.</p> <p>You designate a record as the rowset data source when you design your pagelet using Pagelet Wizard. When the pagelet is executed for display, the data populated for the rowset must contain that record in the first level of the rowset. If the runtime rowset does not contain the record specified in Pagelet Wizard, no data appears in the pagelet.</p>
<b>Search record</b>	<p>Use to create Pagelet Wizard pagelets that provide quick access to specific transactions on a target transaction page. The search record data type enables you to select a basic search record or a component interface.</p> <p>Use the basic search record data type for simple search lookup pages.</p> <p>Use the component interface search record data type when the basic search record does not provide the required functionality; for example, when security or PeopleCode considerations need to be taken into account.</p> <p>Search record data source pagelets are compatible with the search list display format.</p>
<b>URL</b>	URL data type retrieves data from external sources like the Internet. This includes sources such as RSS newsfeeds. Use to display appropriately sized dynamic HTML pages on Pagelet Wizard pagelets. URL data-source pagelets are compatible with the custom and passthru display formats.

---

**Note.** The SOAP data type used in previous releases is now included in the functionality of the IB data type. In PeopleTools 8.50, SOAP services are consumed using Integration Broker, so use the IB data type to create pagelets going forward.

You can define your own data types on the Define Data Types page.

When using Enterprise Portal, Pagelet Wizard includes other data sources that are specific to Enterprise Portal installations.

---

### **See Also**

[Chapter 11, "Using Pagelet Wizard," Defining Display Formats, page 294](#)

[Chapter 11, "Using Pagelet Wizard," Selecting a Navigation Collection Data Source, page 317](#)

[Chapter 11, "Using Pagelet Wizard," Defining Data Types, page 296](#)

[Chapter 11, "Using Pagelet Wizard," Step 2: Selecting a Pagelet Data Source, page 312](#)

---

## **Overview of Pagelet Wizard Display Formats**

Pagelet Wizard supports the creation of pagelets based on the following display formats:

- Table
- List
- Chart
- Menu
- Search list
- Passthru
- Custom (Extensible Stylesheet Language, or XSL)

### ***Table Display Format***

The table display format is appropriate for creating a pagelet that displays data from the PeopleSoft Query and rowset data sources. Following is an example of a table pagelet displaying data from a PeopleSoft Query data source:

ORACLE

Message Sets

Set	Msg	Severity	Message
141	1	M	Personalizations
141	2	M	Explanation
141	3	M	Personalization Options
141	4	M	Explain
141	5	M	Option
141	6	M	Description
141	7	M	Enable Option
141	8	M	Allow User Value
141	9	M	Default Value
141	10	M	Override Value

Table pagelet example

List Display Format

The list display format is appropriate for creating a pagelet that displays data from the PeopleSoft Query and rowset data sources. Following is an example of a list pagelet displaying data from a PeopleSoft Query data source:

ORACLE

Message Sets

1. Personalizations
2. Explanation
3. Personalization Options
4. Explain
5. Option
6. Description
7. Enable Option
8. Allow User Value
9. Default Value
10. Override Value

[Full Query Results](#)

List pagelet example

Chart Display Format

The chart display format is appropriate for creating a pagelet that displays data from the PeopleSoft Query and rowset data sources. Following is an example of a chart pagelet displaying data from a PeopleSoft Query data source:

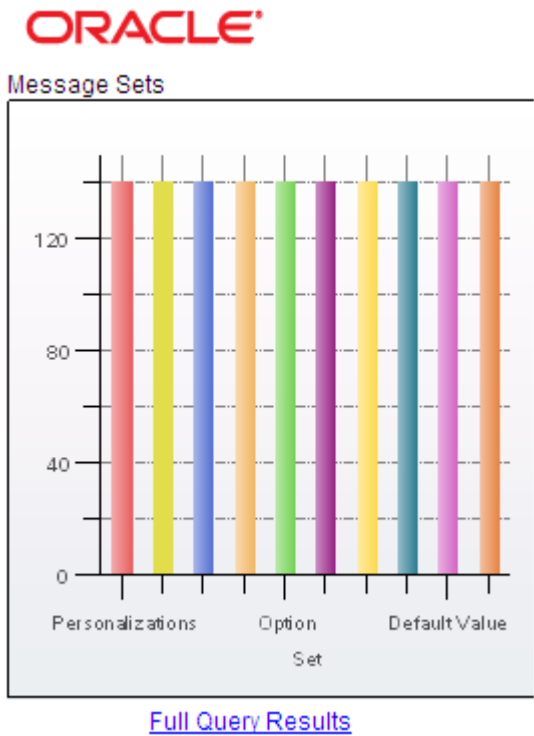


Chart pagelet example

**Menu Display Format**

The menu display format is appropriate for creating a pagelet that displays data from a Navigation Collection data source. Following is an example of a menu pagelet displaying data from a Navigation Collection data source:



Menu pagelet example

### ***Search List Display Format***

The search list display format is appropriate for creating pagelets that display data from the search record data source. Following are examples of search list pagelets displaying data from a search record data source:

**PW Headers**

**PeopleSoft®**

Pagelet Wizard headers

**Search By:** Header ID

**begins with** PAPP

Search list pagelet example (Show Results options set to N)

PW Headers	
PeopleSoft®	
Pagelet Wizard Headers	
Header ID	Description
<a href="#">PAPPBR_HEADER1_BASIC</a>	Sample header 1
<a href="#">PAPPBR_HEADER1_SIMPLE</a>	Sample header 1 simple
<a href="#">PAPPBR_HEADER2_GBI</a>	Sample GBI header
<a href="#">PAPPBR_HEADER2_GBNOTAB</a>	Sample GBI header (no tabs)
<a href="#">PAPPBR_HEADER3_IEONLY_CURVED</a>	Blue curve header - IE only
<a href="#">PAPPBR_HEADER4_TOOLS</a>	Tools 8.42 delivered header
<a href="#">PAPPBR_HEADER4_TOOLS_NOTAB</a>	Tools 8.42 delivered header
<a href="#">PAPPBR_HEADER5_TOOLSCLASSIC</a>	Tools "Classic" header
<a href="#">PAPPBR_HEADER5_TOOLSNOTAB</a>	Tools "Classic" (no tabs)

Search list pagelet example (Show Results option set to Y)

### ***Passthru Display Format***

The passthru display format is appropriate for creating a pagelet that displays data from the HTML, Content Management, IB Connector, and URL data sources. Because the passthru display format performs no further transformations on data from the data source, it should only be used with data sources that return HTML that can be directly displayed as a pagelet. Following is an example of a passthru pagelet displaying data from an HTML data source:



Passthru pagelet example

## Custom Display Format

The custom display format is appropriate for creating a pagelet using all datasources that return XML. Generally, most data types return XML, but there are potential circumstances where IBConnector, URL, and HTML datatypes will not return XML. In those instances, the custom display format would not be appropriate.

Use the custom display format when you want to use a display format other than the ones expressly listed in the Pagelet Wizard. When you select the custom display format, Pagelet Wizard enables you to select a format provided by one of the delivered XSL prototypes, or you can select an XSL template that you created on the Define XSL page.

Following is an example of a custom pagelet displaying data from a PeopleSoft Query data source:



Set	Short Desc	Description
1	Title Bar	PeopleTools Message Bar Items
2	PeopleCode	PeopleCode
3	GEN	General Tools Messages
4	HPR	Help Processor
5	HTM	Help Text Manager
6	IMP	Import Definition Manager
7	OPNQRY	Open Query API
8	PCedit	PeopleCode Editor
9	MDM	Menu Definition Manager
10	CCONTRL	Change Control

[Access Message Sets...](#)

powered by  
**PeopleSoft**

Custom pagelet example

This custom pagelet uses the same PeopleSoft Query data source as the table pagelet example shown earlier; however, this example has a custom XSL template applied.

### See Also

*Enterprise PeopleTools 8.50 PeopleBook: PeopleCode API Reference*, "Charting Classes"

Chapter 11, "Using Pagelet Wizard," Defining Pagelet Wizard XSL Prototypes, page 300

Chapter 11, "Using Pagelet Wizard," Step 4: Selecting a Pagelet Display Format, page 340

---

## Overview of Pagelet Wizard Display Sizes

In addition to the display format, pagelet sizing is also important during pagelet creation. Pagelet Wizard uses pagelet sizing that follows a basic set of rules that enable pagelets to appear properly on a PeopleSoft portal homepage. Pagelet size corresponds to the homepage layout that is specified. You can specify either a two-column layout that displays one narrow pagelet and one wide pagelet, or a three-column layout that displays three narrow pagelets.

Pagelets that you design with the Pagelet Wizard should conform to the dimensions of the narrow column pagelet and (optionally) the wide column pagelet. A narrow pagelet should be 240 pixels wide. Subtracting the border and the internal margin provides 218 pixels for content. A wide pagelet should be 490 pixels wide. Subtracting the border and the internal margin provides 468 pixels for content.

If you display a pagelet that is wider than the homepage column that contains it, an error does not occur. The column expands to accommodate the content; however, this may require horizontal scrolling on the homepage, which can be inconvenient for the end user. In accordance with general pagelet design principles, make the pagelet height as short as possible.

---

## Overview of Pagelet Wizard Pagelet Types

Pagelet Wizard supports the publication of the following pagelet types:

- Homepage pagelets.
- Template pagelets.
- Embeddable pagelets.
- Web Services for Remote Portlets (WSRP) portlets.

### ***Homepage Pagelets***

A homepage pagelet is a version of a pagelet that is available for use on the portal homepage. The layout and content of the homepage is usually configurable by the end user, and homepage pagelets can be designated as optional or required.

---

**Note.** On any PeopleSoft homepage, only one pagelet or portlet should use the calendar prompt. If more than one pagelet or portlet per homepage uses the calendar prompt, unexpected results might occur.

---

See [Chapter 6, "Administering Portal Homepages and Pagelets," Selecting the Tab Content, page 157.](#)

PeopleSoft®

Personalize [Content](#) | [Layout](#)

[Home](#) | [Worklist](#) | [MultiChannel Console](#) | [Add to Favorites](#) | [Sign out](#)

**Menu**

Search:

- My Favorites
- Data Expansion Tools
- Employee Self-Service
- Manager Self-Service
- Supplier Contracts
- Customers
- Partners
- Products
- Catalog Management
- Promotions
- Customer Contracts
- Order Management
- Pricing Configuration
- Customer Returns

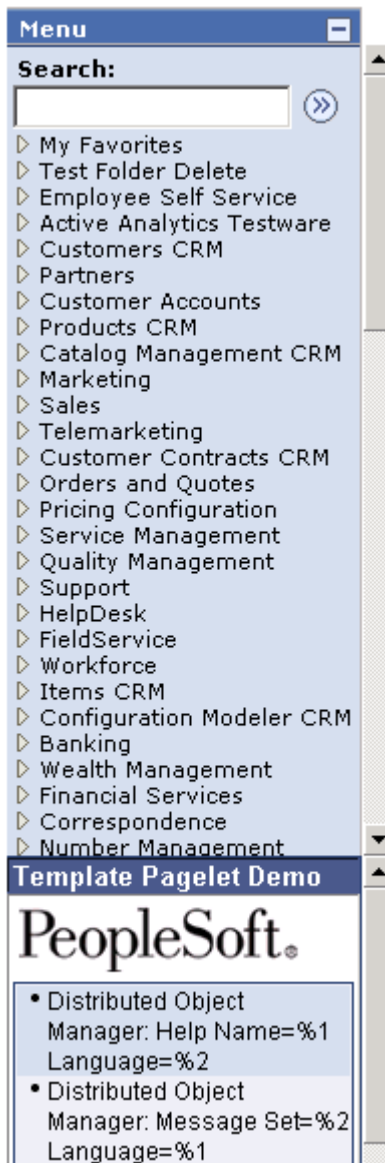
**Content References**

Portal Label	Usage Type
<a href="#">Base Navigation Page</a>	Target
<a href="#">Clean Portal Project</a>	Target
<a href="#">Copy Pagelets</a>	Target
<a href="#">Define Categories</a>	Target
<a href="#">Define Data Types</a>	Target
<a href="#">Define Display Formats</a>	Target
<a href="#">Define Footers</a>	Target
<a href="#">Define Headers</a>	Target
<a href="#">Define Transform Types</a>	Target
<a href="#">Define XSL</a>	Target

Homepage pagelet example

### Template Pagelets

A template pagelet is a version of a pagelet that can be added to a PeopleSoft template. These pagelets can extract and leverage keys specified by the target transaction for the template. Publishing a pagelet as a template pagelet is useful when the PeopleTools Context Manager is used to display context sensitive data in the template pagelet.



Template pagelet example

### ***Embeddable Pagelets***

An embeddable pagelet is a version of a pagelet that can be generated by the Pagelet Wizard application programming interface (API) for rendering within an HTML area of a PeopleSoft Pure Internet Architecture page or an iScript. For example, you might create an OBIEE-based pagelet that displays a report of sales by region to embed on a Regional Sales page.

### ***WSRP Portlet***

A WSRP portlet is a portlet that can be used with external portals that support the WSRP standard.

## See Also

Chapter 11, "Using Pagelet Wizard," Step 6: Specifying Pagelet Publication Options, page 365

Chapter 15, "Understanding Web Services For Remote Portlets (WSRP)," page 491

*Enterprise PeopleTools 8.50 PeopleBook: Using PeopleSoft Applications*

## Defining Pagelet Wizard Setup Values

This section discusses how to:

- Set up the PeopleSoft Integration Broker gateway for Pagelet Wizard.
- Set up the Pagelet Wizard IB Connector (Integration Broker) data source.
- Set up the Pagelet Wizard Integration Broker data source.

## Pages Used to Set Up Pagelet Wizard Setup Values

Page Name	Definition Name	Navigation	Usage
Gateways	IB_GATEWAY	PeopleTools, Integration Broker, Configuration, Gateways	Specify the gateway's location, update configuration settings, and register target connectors to be used with the gateway.  See <i>Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Integration Broker Administration</i> , "Managing Integration Gateways."
Define IB Connector Security - Connectors	PTPPB_IBCONN_TRANS	PeopleTools, Portal, Pagelet Wizard, Define IB Connector Security, Connectors	Apply security to a connector that you want to use with the Pagelet Wizard IB Connector data source.
Define IB Connector Security - Default Security	PTPPB_IBCON_DEFSEC	PeopleTools, Portal, Pagelet Wizard, Define IB Connector Security, Default Security	Select one or more permission lists to which you want to provide access to IB Connector. This security will be applied to connectors for which the Apply Default Security option is selected on the Define IB Connector Security - Connectors page.

<b>Page Name</b>	<b>Definition Name</b>	<b>Navigation</b>	<b>Usage</b>
Define IB Connector Security	PTPPB_IBCON_CUSTSC	Click the Custom Security button on the Define IB Connector Security - Connectors page.	Select one or more permission lists to which you want to provide access to IB Connector. This security will be applied to connectors for which you clicked the Custom Security button on the Define IB Connector Security - Connectors page.

## Setting Up the PeopleSoft Integration Broker Gateway for Pagelet Wizard

This section provides an overview of setting up the PeopleSoft Integration Broker Gateway for Pagelet Wizard and discusses how to set up the default Gateway for use with Pagelet Wizard data sources.

When you use the Pagelet Wizard PeopleTool with PeopleSoft Enterprise Portal installed, the IB Connector and Integration Broker data sources become available for use. These data sources rely on PeopleSoft Integration Broker for web interactivity. You must set up the default gateway before these data sources are usable with your system.

## Setting Up the Default Gateway for Use with Pagelet Wizard Data Sources

Access the Gateways page (PeopleTools, Integration Broker, Configuration, Gateways).

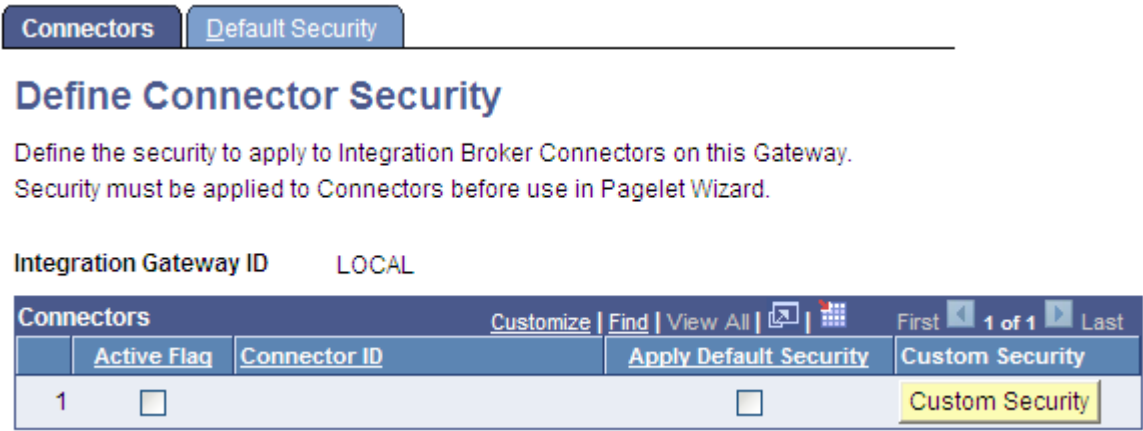
Set up the default gateway according to the instructions in the Integration Broker PeopleBook.

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Integration Broker Administration*, "Managing Integration Gateways."

1. Click Save.
2. Click Load Gateway Connectors.
3. Click Save.

## Setting Up the Pagelet Wizard IB Connector Data Source Security

Access the Define Connector Security page (PeopleTools, Portal, Pagelet Wizard, Define IB Connector Security, Connectors).



Define IB Connector Security - Connectors page

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Integration Broker Administration*, "Using Listening Connectors and Target Connectors."

- Integration Gateway ID**   Select the appropriate Integration Gateway.
- Active Flag**   Select to activate the connector and make it available for use with the Pagelet Wizard IB Connector data source.
- Apply Default Security**   Select to apply the default security settings that are defined on the Define IB Connector Security - Default Security page to the connector.
- Custom Security**   Click to access the Define Connector Security page, where you can define custom security settings for the connector.

Setting Up the Pagelet Wizard Integration Broker Data Source

- When you use the Pagelet Wizard PeopleTool with PeopleSoft Enterprise Portal installed, the Integration Broker data source becomes available for use. You must perform the following tasks before the Integration Broker data source is usable with your system.
1. Set up the Integration Broker listening connector.  
  
See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Integration Broker Administration*, "Using the Integration Broker Quick Configuration Page."
  2. Set up the appropriate service, service operation, handlers, and routings for the operation you want to use.  
  
See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Integration Broker Administration*, "Using the Integration Broker Quick Configuration Page."

## Defining Pagelet Wizard Transformation Types, Display Formats, and Data Types

This section discusses how to:

- Define transformation types.
- Define display formats.
- Define data types.

### Pages Used to Define Pagelet Wizard Transformation Types, Display Formats, and Data Types

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Define Transform Types	PTPPB_TXFRMTYP_MTC	PeopleTools, Portal, Pagelet Wizard, Define Transform Types	Define transformation methods that can be applied to Pagelet Wizard pagelet content.
Define Display Formats	PTPPB_DISPFRMT_MTC	PeopleTools, Portal, Pagelet Wizard, Define Display Formats	Define display formats that can be used to format Pagelet Wizard pagelet content.
Define Data Types	PTPPB_DATATYPE_MTC	PeopleTools, Portal, Pagelet Wizard, Define Data Types	Define data types that can be used to supply Pagelet Wizard with pagelet content.
Document Type Definition (DTD)	PTPPB_DATADTD_SBP	Click the View Document Type Definition (DTD) link on the Define Data Types page.	Access the DTD that describes the XML generated by the selected data type.

### Defining Transformation Types

Access the Define Transform Types page (PeopleTools, Portal, Pagelet Wizard, Define Transform Types).

## Define Transform Types

Define the transform types used by Pagelet Wizard

Transformation Type

XSL

\*Description

XSL

Long Description:

XSL

Supporting Application Class

Package Name

PTPPB\_PAGELET

Path

Transformer

Application Class ID

XSLTransformer

Define Transform Types page

Pagelet Wizard is delivered with the following transformation type definitions to support their associated display formats:

- PASSTHRU
- XSL

**Warning!** Do not alter delivered field values for these transformation types.

This Define Transform Types page enables you to build your own transformation types. For example, you could create a new transformation type called ABC123. You would then extend Pagelet Wizard to perform ABC123 transformations by building a supporting application class and registering it as a transformation type. You could then create builders for assembling pagelet displays that create ABC123 transformation definitions.

See [Chapter 11, "Using Pagelet Wizard," Overview of Pagelet Wizard Display Formats, page 280.](#)

<b>Transformation Type</b>	Enter the internal name that you want to assign to the transformation type.
<b>Description</b>	For reference purposes only, enter a short description of the transformation type.
<b>Long Description</b>	For reference purposes only, enter a more detailed description of the transformation type.

### Supporting Application Class

The application class specified here refers to the PeopleCode that implements the transformation type.

<b>Package Name</b>	Enter the application class package name that you want to use for the transformation type.
---------------------	--




<b>Path</b>	Enter the application class path that you want to use for the transformation type.
<b>Application Class ID</b>	Enter the name of the application class that you want to use for the transformation type. The class must exist in the application package name that you have specified.

## Defining Display Formats

Access the Define Display Formats page (PeopleTools, Portal, Pagelet Wizard, Define Display Formats).

### Define Display Formats

Define the display formats used by Pagelet Wizard

Display Format ID:	SEARCHREC		
*Description:	Search List	<input checked="" type="checkbox"/> Active	
Long Description:	Display your component's search results or a search dialog box for your component's search page		
Sort Order:	70		
Image Name:	PTPP_PROC_SEARCH_LIST_ICN		
*Transformation Type:	XSL		XSL
Page Name:	PTPPB_WIZ_DISP_SRC		Display Options - Search List

Supporting Application Class	
Package Name:	PTPPB_PAGELET
Path:	TransformBuilder
Application Class ID:	SearchRecordBuilder

Define Display Formats page

A display format performs the following tasks:

- Defines a TransformBuilder, which is code that builds a particular type of display format (chart, menu, list, and so forth).
- Identifies the proper transformation to use with pagelets created with a specific display format builder.

For example, the chart display format defines the code that builds charts, and it specifies that Pagelet Wizard must use the XSL transformation for pagelets created using this display format.

Pagelet Wizard is delivered with the following display format definitions to support their associated display formats:

- CHART

- CUSTOM
- LIST
- MENU
- PASSTHRU
- SEARCHREC
- TABLE

---

**Warning!** Do not alter delivered field values for these display formats.

---

See [Chapter 11, "Using Pagelet Wizard," Overview of Pagelet Wizard Display Formats, page 280.](#)

<b>Display Format ID</b>	Enter the name that you want to assign to the display format.
<b>Description</b>	Enter a short description of the display format. This short description appears as the display format name value on the Select Display Format page in Pagelet Wizard.
<b>Active</b>	<p>Select to activate the display format and make it a selectable value in Pagelet Wizard.</p> <p>If you deactivate a display format, the display format is no longer available for selection on the Select Display Format page when accessed for associated data types.</p> <p>Existing pagelets that have been created using a deactivated display format are unaffected; however, you can no longer use Pagelet Wizard to change the display settings for those pagelets.</p>
<b>Long Description</b>	Enter a detailed description of the display format. This long description appears as the display format description on the Select Display Format page in Pagelet Wizard.
<b>Sort Order</b>	<p>Enter a numeric value to specify the order in which you want the display format to appear as a selectable field value relative to other display formats on the Select Display Format page.</p> <p>See <a href="#">Chapter 11, "Using Pagelet Wizard," Step 4: Selecting a Pagelet Display Format, page 340.</a></p>
<b>Image Name</b>	Select the image that you want to appear with the display format name and description on the Select Display Format page.
<b>Transformation Type</b>	<p>Display formats use a transformation technology. Select the transformation type that you want to use for the display format. Available transformation types are defined on the Define Transform Types page.</p> <p>See <a href="#">Chapter 11, "Using Pagelet Wizard," Defining Transformation Types, page 292.</a></p>

**Page Name** Enter the object name of the page to be used to define display options for this display format. This page must exist in the PTPPB\_WIZARD component in PeopleSoft Application Designer.

### ***Supporting Application Class***

The application class specified here refers to the PeopleCode that implements the display format. This code is the builder that creates the definition that Pagelet Wizard uses (along with the transformation) to display the pagelet.

**Package Name** Enter the application class package name to use for the display format.

**Path** Enter the application class path to use for the display format.

**Application Class ID** Enter the name of the application class to use for the display format. The class must exist in the application package name that you specify.

## **Defining Data Types**

Access the Define Data Types page (PeopleTools, Portal, Pagelet Wizard, Define Data Types).

## Define Data Types

Define the data types used by Pagelet Wizard.

**Data Type:** HTML

**\*Description:**  ☒ Active

**Long Description:**

The HTML data source allows you to specify HTML code which you would like displayed in your pagelet.

**Supporting Application Class**

**Package Name:**

**Path:**

**Application Class ID:**

**Search Application Class**

**Package Name:**

**Path:**

**Application Class ID:**

**Display Formats to use with this Data Type**

	Display Format ID	Description		
1	<input type="text" value="CUSTOM"/>	Custom	<input type="button" value="+"/>	<input type="button" value="-"/>
2	<input type="text" value="PASSTHRU"/>	Passthru	<input type="button" value="+"/>	<input type="button" value="-"/>

Define Data Types page

Pagelet Wizard is delivered with the following data type definitions to support their associated data sources:

- HTML data type enables you to specify HTML code to display in the pagelet.
- IBCONNECTOR data type interacts directly with connectors registered to the Integration Broker and secured using Define IB Connector Security within Pagelet Wizard. This data type should only be used when it is not desirable to use the Integration Broker data type.
- IBDATASOURCE data type retrieves messages using Integration Broker. Specifically, it uses outbound synchronous messages to retrieve data from external sources.
- NAVIGATION\_COLLECTION data type enables you to display navigation collections within the context of a pagelet
- OBIEE data type interacts directly with analytics server.
- PSQUERY data type enables you to use PeopleSoft Query to create database queries on records and fields within the PeopleSoft system.
- ROWSET data type is used with embedded pagelets. Rowsets can exist within components.
- SEARCHREC data type displays the search box or the search results from a given component.

- URL data type retrieves data from external sources like the Internet. This includes sources like RSS newsfeeds.

---

**Warning!** Do not alter delivered field values for these data types.

---

See [Chapter 11, "Using Pagelet Wizard," Overview of Pagelet Wizard Data Sources, page 278.](#)

<b>Data Type</b>	Enter the internal name that you want to assign to the data type.
<b>Description</b>	Enter a short description of the data type. This short description appears as the prompt list value when you select data sources in Pagelet Wizard.
<b>Active</b>	<p>Select activate the data type and make it a selectable value in Pagelet Wizard.</p> <p>If you deactivate a data type, the data type is no longer available for selection on the Select Data Source page.</p> <p>If you deactivate a data type that is in use, associated pagelets do not display data. The pagelets display an error message instead.</p>
<b>Long Description</b>	Enter a detailed description of the data type. This long description appears as the data type description on the Select Data Source page in Pagelet Wizard.

### **Supporting Application Class**

The application class specified here refers to the PeopleCode that implements the data type. This is the code that Pagelet Wizard calls to retrieve the data for the pagelet.

This table describes the Supporting Application Class fields:

<b>Package Name</b>	Enter the application class package name that you want to use for the data type.
<b>Path</b>	Enter the application class path that you want to use for the data type.
<b>Application Class ID</b>	Enter the name of the application class that you want to use for the data type. The class must exist in the application package name that you specify.

---

**Note.** The search option of each pagelet has to be explicitly declared through this page or Pagelet Wizard to activate the search box, even for *Inline XSL* type of the search box.

---

### **Search Application Class**

The application class specified here refers to the PeopleCode that implements a default pagelet search for this data type. This is the code that Pagelet Wizard calls to build and place the search box in the pagelet; perform the search; and render and format the search results.

Changes to the data type level default search application class will appear in any pagelet of this data type automatically the next time the pagelet is rendered, with the exception of the pagelets for which you define a custom search application class. Examples of these changes include:

The *PTPPB\_PAGELET:PageletSearch:PortalSearch* sample search application class is part of a standard PeopleTools installation. This class uses the portal registry index to perform the search. You can use this sample search class to test the pagelet search functionality for any pagelet or you can associate this search class with either a pagelet data type or with a pagelet.

- Changing the code in the search class.
- Adding a new search class.
- Removing a search class.
- Replacing a search class.

---

**Note.** Pagelet search is supported for homepage and embedded pagelets only.

---

This table describes the Search Application Class fields:

<b>Package Name</b>	Enter the Search application class package name that you want to use for the data type.
<b>Path</b>	Enter the Search application class path that you want to use for the data type.
<b>Application Class ID</b>	Enter the name of the Search application class that you want to use for the data type. The class must exist in the application package name that you specify.

### ***Display Formats to use with this Data Type***

<b>Display Format ID</b>	Enter the display formats that are supported by the data type. You can select from existing display formats defined on the Define Display Formats page. See <a href="#">Chapter 11, "Using Pagelet Wizard," Defining Display Formats, page 294.</a>
<b>View Document Type Definition (DTD)</b>	Click to access the Document Type Definition (DTD) page, where you can access the DTD that describes the XML generated by the selected data type.

### ***Viewing the Document Type Definition (DTD)***

The Document Type Definition (DTD) page enables you to view the tags and attributes are used to describe content in the XML from the data source type. This information is useful when using third-party tools to generate XSL templates for the datatype.

## Document Type Definition (DTD)

A DTD is a specification that accompanies a document and identifies what the markup codes are that separate paragraphs, identify topic headings, and so forth and how each is to be processed. The following is the DTD which describes the XML outputted by the data source type.

Data Type: NAVIGATION\_COLLECTION

DTD:

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT ClearDotImg (#PCDATA)>
<!ELEMENT Contents (NavItem+)>
<!--ATTLIST Contents FolderCount CDATA #REQUIRED ShortcutCount CDATA
#REQUIRED-->
<!ELEMENT Description (#PCDATA)>
<!ELEMENT IScript_AppHP (#PCDATA)>
<!ELEMENT ImageName (#PCDATA)>
<!ELEMENT ImageURL (#PCDATA)>
<!ELEMENT Label (#PCDATA)>
<!ELEMENT MaxChildLinks (#PCDATA)>
<!--ELEMENT NavCollection (Label, Description, NavPageName, SCName,
SCPortal, ImageName, ImageURL, ShowIcons, StyleSheet, MaxChildLinks
```

Document Type Definition (DTD) page

## Defining Pagelet Wizard XSL Prototypes

This section discusses how to define XSL prototypes for use in building Pagelet Wizard pagelets.

### Page Used to Define Pagelet Wizard XSL Prototypes

Page Name	Definition Name	Navigation	Usage
Define XSL	PTPPB_XSLPROT_MTCE	PeopleTools, Portal, Pagelet Wizard, Define XSL	Create custom XSL definitions for use with Pagelet Wizard.

### Defining XSL Prototypes

Access the Define XSL page (PeopleTools, Portal, Pagelet Wizard, Define XSL).

## Define XSL

Define the XSL prototypes used by Pagelet Wizard's transformation step.

Pagelet XSL

XSL Prototype ID: TABLE\_QUERY

\*Description:

Table

☒ Active

Long Description:

Table - all rows, all columns

\*Data Type

PSQUERY

PS Query

Define XSL:

<?xml version="1.0" ?>
<!--
Description: XSL T supplied with the Pagelet Wizard
-->

Delete XSLT

Define XSL page

Pagelet Wizard is delivered with the following XSL prototypes to support their associated data sources and display formats:

<i>XSL Prototype Name</i>	<i>Data Type</i>	<i>Description</i>
NAV_SMALLIMAGES	NAVIGATION_COLLECTION	Small Images
LIST_QUERY	PSQUERY	Bullet List
NUMBERED_LIST_QUERY	PSQUERY	Numbered List
TABLE_QUERY	PSQUERY	Table
PAPP_IB_BULLET_LIST	IBDATASOURCE	Bullet List
PAPP_IB_CUSTOM_LIST	IBDATASOURCE	Customizable List
PAPP_IB_CUSTOM_TABLE	IBDATASOURCE	Customizable Table
PAPP_IB_NUMBER_LIST	IBDATASOURCE	Numbered List
PAPP_IB_NUMBER_LIST_H	IBDATASOURCE	Numbered List - Header
PAPP_IB_SKELETON	IBDATASOURCE	Skeleton XSL
PAPP_IB_TABLE	IBDATASOURCE	Table
PAPP_URL_ATOM03	URL	ATOM 0.3

<i><b>XSL Prototype Name</b></i>	<i><b>Data Type</b></i>	<i><b>Description</b></i>
PAPP_URL_ATOM10	URL	ATOM 1.0
PAPP_URL_RSS	URL	RSS
PAPP_URL_RSS091	URL	RSS 0.91-0.93
PAPP_URL_RSS10	URL	RSS 1.0
PAPP_URL_RSS20	URL	RSS 2.0
PAPP_URL_SKELETON	URL	Skeleton XSL

---

**Warning!** Do not alter the delivered XSL for these XSL prototypes.

---

<b>XSL Prototype ID</b>	Enter an ID for the XSL prototype.
<b>Description</b>	<p>Enter a description of the XSL prototype. The text that you enter here appears as available XSL template field values on the Specify Display Options page when you create a custom display format pagelet.</p> <p>See <a href="#">Chapter 11, "Using Pagelet Wizard," Specifying Custom Display Options, page 360</a>.</p>
<b>Active</b>	<p>Select to activate the XSL prototype and make it an available XSL template field value on the Specify Display Options page for custom display format pagelets.</p> <p>If you deactivate an XSL prototype, the prototype is no longer available for selection on the Specify Display Options page for the custom display format. Existing pagelets that have been created using the deactivated XSL prototype are unaffected.</p>
<b>Long Description</b>	For reference purposes only, enter a long description of the XSL prototype.
<b>Data Type</b>	Select the applicable data type for the XSL prototype. You can assign only one data type per XSL prototype.
<b>Define XSL</b>	Enter XSL code for formatting pagelet XML data in the desired pagelet format.
<b>Delete XSL</b>	Click to delete the XSL prototype definition. You will be prompted to confirm your deletion.

---

## Defining Pagelet Wizard Headers and Footers

This section discusses how to:

- Define headers.

- Define header images.
- Define footers.
- Define footer images.

## Pages Used to Define Pagelet Wizard Headers and Footers

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Define Header	PTPPB_HDR_MTCE	PeopleTools, Portal, Pagelet Wizard, Define Headers, Define Header	Create a pagelet header design. Enter HTML that you want to appear on the header.
Define Headers - Specify Image	PTPPB_HDR_IMAGE	PeopleTools, Portal, Pagelet Wizard, Define Headers, Define Header, Specify Image	Specify an image that you want to appear on the pagelet header. Specify image properties.
Define Headers - Preview	PTPPB_HDR_PREVIEW	PeopleTools, Portal, Pagelet Wizard, Define Headers, Define Header, Preview	View the fully assembled pagelet header. After previewing the pagelet header, you can adjust the HTML and image parameters as needed.
Define Footer	PTPPB_FTR_MTCE	PeopleTools, Portal, Pagelet Wizard, Define Footers, Define Footer	Create a pagelet footer design. Enter HTML that you want to appear on the footer.
Define Footers - Specify Image	PTPPB_FTR_IMAGE	PeopleTools, Portal, Pagelet Wizard, Define Footers, Specify Image	Specify an image that you want to appear on the pagelet footer. Specify image properties.
Define Footers - Preview	PTPPB_FTR_PREVIEW	PeopleTools, Portal, Pagelet Wizard, Define Footers, Preview	View the fully assembled pagelet footer. After previewing the pagelet footer, you can adjust the HTML and image parameters as needed.

## Defining Headers

Access the Define Header page (PeopleTools, Portal, Pagelet Wizard, Define Headers, Define Header).

Define Header

Specify Image

Preview

## Define Headers

Define the pagelet headers used by Pagelet Wizard. Specify the HTML to display in the header.

Pagelet Header

Header ID:

PTPP\_PEOPLESOFT\_LOGO

\*Description:

PeopleSoft Logo

☒ Active

Long Description:

PeopleSoft Logo

HTML Area:

Delete Header

Define Header page

Pagelet Header

Header ID	Enter an ID for the pagelet header.
Description	Enter a description of the pagelet header. The text that you enter here appears as the selectable field value in the Header field when users assign headers to pagelets on the Specify Display Options page.
Active	Select to activate the pagelet header and make it a selectable field value in the Header field on the Specify Display Options page.  If you deactivate a header that is in use, the header no longer appears on associated pagelets. In addition, the header is not be available for selection on the Specify Display Options page.
Long Description	For reference purposes only, enter a long description of the pagelet header.
HTML Area	Enter the HTML code that you want to use to generate the pagelet header. You can enter both HTML text and image references. Alternatively, you can include an image by selecting it on the Specify Image page.
Delete Header	Click to delete the pagelet header design. You will be prompted to confirm your deletion.

## Defining Header Images

Access the Header Image page (PeopleTools, Portal, Pagelet Wizard, Define Headers, Define Header, Specify Image).

Define Header
Specify Image
Preview

### Header Image

Specify the image to display in the header. Depending on the Image Source, select an image from the system catalog or enter an image URL.

Header ID: PTPP\_PEOPLESOFT\_LOGO

Image Source: System Catalog


Image Name: NEW\_PS\_LOGO

Alignment:

Height:

Width:

Float Over Text:



Define Headers - Specify Image page

### Header Image

- Image Source** Select the image source. Options are:
- System Catalog.* Select to specify an image from the PeopleTools system image catalog.
  - External Source.* Select to specify an image from a source external to the PeopleTools system image catalog.
- Image Name** If you select the *System Catalog* image source, click the Search button to access a list of existing images.
- Image URL** If you select the *External Source* image source, enter a URL for the desired image file. All image types are supported. Ensure that the location of the image provides reliable access.
- Alignment** Select the alignment of the image. Options are:
- Left:* Select to align the image with the left margin of the pagelet.
  - Right:* Select to align the image with the right margin of the pagelet.

Height	Enter the image height in pixels if the image needs to be resized. If you enter an image height, the system adjusts the image width to fit the image height.
Width	Enter the image width in pixels if the image needs to be resized. If you enter an image width, the system adjusts the image height to fit the image width.
Float Over Text	Enter text that you want to appear when the user moves the cursor over the image.

Defining Footers

Access the Define Footer page(PeopleTools, Portal, Pagelet Wizard, Define Footers, Define Footer) .

Define FooterSpecify ImagePreview

Define Footers

Define the pagelet footers used by Pagelet Wizard. Specify the HTML to display in the footer.

Pagelet Footer

Footer ID:PTPP\_POWEREDBY\_FOOTER

\*Description:Powered by PeopleSoft

☒ Active

Long Description:Powered by PeopleSoft

HTML Area:

Delete Footer

Define Footer page

Pagelet Footer

Footer ID	Enter an ID for the pagelet footer.
Description	Enter a description of the pagelet footer. The text that you enter here appears as the selectable field value in the Footer field when users assign footers to pagelets on the Specify Display Options page.

<b>Active</b>	<p>Select to activate the pagelet footer and make it a selectable field value in the Footer field on the Specify Display Options page.</p> <p>If you deactivate a footer that is in use, the footer no longer appears on associated pagelets. In addition, the footer is not be available for selection on the Specify Display Options page.</p>
<b>Long Description</b>	<p>For reference purposes only, enter a long description of the pagelet footer.</p>
<b>HTML Area</b>	<p>Enter that HTML code that you want to use to generate the pagelet footer. You can enter both HTML text and image references. Alternatively, you can include an image by selecting it on the Specify Image page.</p>
<b>Delete Footer</b>	<p>Click to delete the pagelet footer design. You will be prompted to confirm your deletion.</p>

Defining Footer Images

Access the Footer Image page (PeopleTools, Portal, Pagelet Wizard, Define Footers, Specify Image).

Define Footer

Specify Image

Preview

### Footer Image

Specify the image to display in the footer. Depending on the Image Source, select an image from the system catalog or enter an image URL.

Footer Image

Footer ID:

PTPP\_POWEREDBY\_FOOTER

Image Source:

System Catalog

Image Name:

PT\_PORTAL\_POWEREDBY

Alignment:

Height:

Width:

Float Over Text:

Powered by PeopleSoft

Define Footers - Specify Image page

**Footer Image****Image Source**

Select the image source. Options are:

*System Catalog*: Select to specify an image from the PeopleTools system image catalog.

*External Source*: Select to specify an image from a source external to the PeopleTools system image catalog.

**Image Name**

If you select *System Catalog* as the image source, click the Search button to access a list of existing images.

**Image URL**

If you select *External Source* as the image source, enter a URL for the desired image file. All image types are supported. Ensure that the location of the image provides reliable access.

**Alignment**

Select the alignment of the image. Options are:

*Left*: Select to align the image with the left margin of the pagelet.

*Right*: Select to align the image with the right margin of the pagelet.

**Height**

Enter the image height in pixels if the image needs to be resized. If you enter an image height, the system adjusts the image width to fit the image height.

**Width**

Enter the image width in pixels if the image needs to be resized. If you enter an image width, the system adjusts the image height to fit the image width.

**Float Over Text**

Enter text that you want to appear when the user moves the cursor over the image.

---

## Defining Pagelet Wizard Pagelet Categories

This section discusses how to define Pagelet Wizard pagelet categories.

### Page Used to Define Pagelet Wizard Pagelet Categories

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Define Categories	PTPPB_PGLTCATG_MTC	PeopleTools, Portal, Pagelet Wizard, Define Categories	Define categories used to organize pagelets created using Pagelet Wizard.

### Defining Pagelet Categories

Access the Define Categories page (PeopleTools, Portal, Pagelet Wizard, Define Categories).

Define Category

Define Categories

Define the categories used to organize the pagelets created with Pagelet Wizard.

Pagelet Category

Category ID:PTPP\_ADMINISTRATION

\*Description:Portal Administration

☒Active

Long Description:Contains pagelets for the Portal Administrator

Delete Category

Define Categories page

- Category ID

Enter a Pagelet Wizard pagelet category ID. You can use category IDs to organize Pagelet Wizard pagelets. The portal administrator can use this value when running pagelet reports and when searching for pagelets that need to be moved between different portal sites.

Categories provide another level of organization in addition to the owner ID. For example, you can use a departmental owner ID, and within that, categorize your pagelets by pagelet type.
- Description

Enter a description of the pagelet category. The text that you enter here appears as a field value in the Category ID field when users assign pagelets to categories on the Specify Pagelet Information page.
- Active

Select to activate the pagelet category and make it a selectable field value in the Category ID field on the Specify Pagelet Information page.

If you deactivate a category, the category is no longer available for selection on the Specify Pagelet Information page. Existing pagelets that have been created using the deactivated category are unaffected.
- Long Description

For reference purposes only, enter a long description of the pagelet category.
- Delete Category

Click to delete a pagelet category. You are prompted to confirm the deletion.

If you delete a category, the category is no longer available for selection on the Specify Pagelet Information page. Existing pagelets that have been created using the deactivated category still appear, but the pagelet category is removed from the pagelet definition in Pagelet Wizard.

---

## Step 1: Entering Pagelet Identifying Information

This section discusses how to specify pagelet information. This is step 1 in the Pagelet Wizard pagelet creation process.

---

**Note.** Before you start building a pagelet with Pagelet Wizard, familiarize yourself with all data sources. This enables you to evaluate the best way to retrieve the data that you want to display in the pagelet.

---

### See Also

[Chapter 11, "Using Pagelet Wizard," Overview of Pagelet Wizard Data Sources, page 278](#)

## Page Used to Enter Pagelet Identifying Information

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Specify Pagelet Information	PTPPB_WIZ_INFO	PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard	Define description and summary information for a pagelet.

## Specifying Pagelet Information

Access the Specify Pagelet Information page (PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard).

Pagelet Wizard

Step 1 of 6

123456

Next >

### Specify Pagelet Information

The following information will be used to identify and categorize your pagelet.

Pagelet Information	
Pagelet ID:	PT_1
*Pagelet Title:	<input type="text" value="New Construction Update"/>
Description:	<div></div>
Owner ID:	<div>PeopleTools</div>
Category ID:	<div>Portal Administration</div>
Help URL:	<input type="text" value="http://help.oracle.com/construction.html"/>

Specify Pagelet Information page

<b>Pagelet ID</b>	<p>Enter an ID for the pagelet. Each pagelet that you build with Pagelet Wizard must have a unique ID.</p> <hr/> <p><b>Note.</b> Pagelet IDs can only contain the characters A-Z, a-z, and _. All other characters put into a pagelet ID will be removed by the system. If all characters are removed from the user entered pagelet ID, then the Registry Object Prefix will be used instead. If, after removing illegal characters from the pagelet ID, the name is found to match an existing pagelet ID, then Pagelet Wizard will append numbers to the end of the pagelet ID to make the new pagelet ID unique.</p> <hr/>
<b>Pagelet Title</b>	<p>Enter a unique title for the pagelet. This value appears in the pagelet title bar, which is the solid bar at the top of the pagelet. If you change the pagelet title value for a published pagelet, you must unpublish and republish the pagelet for the change to take effect.</p>
<b>Description</b>	<p>Enter additional descriptive information to identify the pagelet. This reference text is used in Verity and search record searches.</p>
<b>Owner ID</b>	<p>Select the value that you want to use to identify pagelets created by the organization. The portal administrator can use this value when searching for pagelets through PeopleSoft Application Designer or SQL queries on the portal registry table.</p> <p>Do not select a delivered PeopleSoft owner ID. You can define unique owner IDs for the organization by entering field translate values for the OBJECTOWNERID field using PeopleSoft Application Designer.</p>

**Category ID**

To organize Pagelet Wizard pagelets, select a category assignment for the pagelet. You create pagelet categories on the Define Categories page. The portal administrator can use this value when running pagelet reports and when searching for pagelets that need to be moved between different portal sites.

Categories provide another level of organization in addition to the owner ID. For example, you can use a departmental owner ID, and within that, categorize pagelets for that ID by pagelet type.

See [Chapter 11, "Using Pagelet Wizard," Defining Pagelet Wizard Pagelet Categories, page 308.](#)

**Help URL**

Enter the URL of the help document to associate with the pagelet. Entering a URL causes the Help button to appear in the pagelet header bar.

When the pagelet is published as a homepage pagelet, a PTPPB\_PAGELET\_HELP\_LINK content reference attribute is added to the pagelet content reference definition corresponding to the pagelet. The value of this content reference attribute will be the help URL entered by the user.

---

**Note.** The help button appears only when the pagelet is on the homepage inside the PeopleSoft Portal; it will not appear when the pagelet is displayed as a WSRP portlet.

---



---

**Note.** The pagelet Help URL takes precedence over the Help ID, when both attributes are set for a pagelet.

---



---

## Step 2: Selecting a Pagelet Data Source

This section discusses selecting a pagelet data source. This is step 2 in the Pagelet Wizard pagelet creation process. Specifically, this section discusses how to :

- Select an HTML data source.
- Select an IB connector data source.
- Select an Integration Broker data source.
- Select a navigation collection data source.
- Select an OBIEE report data source.
- Select a PeopleSoft Query data source.
- Select a rowset data source.
- Select a search record data source.
- Select a URL data source.

You can define other data sources by using the Define Data Types page.

**See Also**

Chapter 11, "Using Pagelet Wizard," Overview of Pagelet Wizard Data Sources, page 278

Chapter 11, "Using Pagelet Wizard," Defining Data Types, page 296

**Page Used to Select a Pagelet Data Source**

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Select Data Source	PTPPB_WIZ_DATASRC	PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard  Click the Next button or the icon for step 2.	Select the type of data source for the pagelet.

**Selecting an HTML Data Source**

Access the Select Data Source page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Click button 2).

## Pagelet Wizard

## Step 2 of 6



&lt; Previous

Next &gt;

## Select Data Source

Select the type of data and specify the source which contains the data you want displayed in your pagelet.

## New Construction Update

\*Data Type

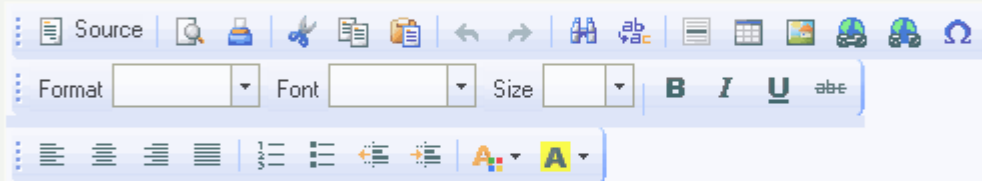
HTML

## ▼ Description

The HTML data source allows you to specify HTML code which you would like displayed in your pagelet.

## Data Source

## \*HTML Text:



Select Data Source page (HTML data source)

Select *HTML* in the Data Type field.

When you use HTML as the data source for a pagelet, the pagelet does not provide configurable parameters that enable criteria selection. Consequently, there is also no end user personalization for HTML data source pagelets.

**Note.** Source code mode is disabled in the Rich Text Editor on the Step 2 page of the Pagelet Wizard for HTML type of pagelets. This is for security reasons, as malicious code could be inserted in Source code mode. If you want to enable the Source mode for some special reasons, please use Application Designer to open the PTPPB\_WIZ\_DATASRC page, double click the PTPPB\_WORKREC.PTPPB\_HTML\_AREA\_01 long edit box field on the page, select the Options tab, select PT\_RTE\_CFG\_PTPPB from the Configuration Setting Id drop down list box, and then save the changes.

When creating an HTML data source pagelet, you skip step 3 and proceed to step 4, in which you select the pagelet display format.

## Data Source

### HTML Text

Enter the HTML code that you want to use as the data source for the pagelet. You can enter an HTML document or fragment. This HTML can include embedded JavaScript.

Ensure that the HTML that you enter generates content that is appropriately sized for pagelet display.

See [Chapter 11, "Using Pagelet Wizard," Defining Data Types, page 296.](#)

## Selecting an IB Connector Data Source

Access the Select Data Source page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 2).

Select *IB Connector* in the Data Type field.

**Pagelet Wizard**
**Step 2 of 6**

1
2
3
4
5
6

< Previous
Next >

### Select Data Source

Select the type of data and specify the source which contains the data you want displayed in your pagelet.

Unit IBCON Test

**\*Data Type** IB Connector

**Description**

The IB Connector DataType can interact directly with connectors registered to the Integration Broker, and secured using "Define IB Connector Security" within Pagelet Wizard. This DataType should only be used when it is not desirable to use the Integration Broker DataType.

**Data Source**

**Integration Gateway ID:** LOCAL

**Connector ID:** HTTPTARGET

Select Data Source page (IB Connector data source)

## Data Source

**Integration Gateway ID** Select the Integration Broker gateway that contains the connector that you want to use to supply content to the pagelet.

**Connector ID** Select the connector that you want to use to supply content to the pagelet.

## Selecting an Integration Broker Data Source

Access the Select Data Source page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 2).

Select *Integration Broker* in the Data Type field.

**Pagelet Wizard** Step 2 of 6

1 2 3 4 5 6 < Previous Next >

### Select Data Source

Select the type of data and specify the source which contains the data you want displayed in your pagelet.


test for facilities


**\*Data Type**

**Description**

The Integration Broker DataType retrieves messages using Integration Broker. Specifically, it can use outbound synchronous messages to retrieve data from external sources.

**Data Source**

**Service Operation:**  

**Receiver Node Name:**  

Select Data Source page (Integration Broker data source)

You can use an outbound synchronous message defined in the local database as a data source for a Pagelet Wizard pagelet. Outbound, synchronous messages are messages for which a request is made to a target system and a response is expected in real time.

The message can be either structured or unstructured. If the message is structured, then a record structure has been associated with the outbound message and that record structure is translated into data source parameters during step 3. If the message is unstructured, then no record structure has been associated with the outbound message, so no data source parameters can be supplied and Pagelet Wizard bypasses step 3.

Typically, because outbound synchronous messages are designed to run in real time, they are associated with transactions with good performance records. As with all pagelet data sources, you should understand that pagelet performance, and homepage performance are tied to the performance of the underlying Integration Broker message and target transaction. You should review the performance of the target transaction before creating an Integration Broker message-based pagelet.

## Data Source

**Service Operation** Select the requesting service operation that you want to be the data source for the pagelet.

**Receiver Node Name** Select the node that is associated with the selected requesting service operation.

## Data Source Details

If the selected message is structured, the Data Source Details group box displays input and output records and fields that are defined in the message structure.

## Selecting a Navigation Collection Data Source

Access the Select Data Source page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 2).

Select *Navigation Collection* in the Data Type field.

**Pagelet Wizard** Step 2 of 6

1 2 3 4 5 6 < Previous Next >

### Select Data Source

Select the type of data and specify the source which contains the data you want displayed in your pagelet.

New Construction Update

\*Data Type Navigation Collection

**Description**

Navigation Collections are collections of links and folders which can be used for customized navigation. The Navigation Collection data source allows you to display these collections within the context of a pagelet.

**Data Source**

\*Portal Name: EMPLOYEE

\*Collection Name: PT\_PTPP\_PORTAL\_ROOT

Select Data Source page (Navigation Collection data source)

The Navigation Collection that you use as the data source must have already been created using the Navigation Collections feature. However, the Navigation Collection does not need to exist in its final form. Any changes that you make to the underlying Navigation Collection used as the data source are immediately reflected on any pagelets created from that Navigation Collection.

---

**Note.** Although it is possible to create a Navigation Collection pagelet directly from the Navigation Collections feature, pagelets created using this feature cannot be maintained in Pagelet Wizard. Only Navigation Collection pagelets created specifically with Pagelet Wizard can be maintained in Pagelet Wizard.

---

See [Chapter 4, "Working With Navigation Pages," Understanding Navigation Collections, page 48.](#)

**Data Type**                      Select *Navigation Collection*.

### **Data Source**

**Portal Name**                      Select the portal that contains the navigation collection for the pagelet.

**Collection Name**                Select the name of the Navigation Collection that you want to use to create the pagelet.

## **Selecting an OBIEE Report Data Source**

Access the Select Data Source page. (PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 2)

Select *OBIEE Report* in the Data Type field.

Pagelet Wizard

Step 2 of 6

1

2

3

4

5

6

< Previous

Next >

### Select Data Source

Select the type of data and specify the source which contains the data you want displayed in your pagelet.

Market by Region

\*Data Type

Description

The OBIEE Report Data Type displays a report from an Oracle Business Intelligence Server.

Data Source

Report ID:

Report Name: OBIEE Server1 > MARKET\_REGION

Report View ID:

Pagelet Wizard-Select Data Source page

## Data Source

- Report ID** Select the Report that you want to be the data source. Only reports that have been loaded into the PeopleSoft application appear in this list.  
See [Chapter 14, "Configuring OBIEE Report Pagelets," Loading OBIEE Report Definitions, page 482.](#)
- Report Name** This field displays the OBIEE server name and the report name. This field is display-only.
- Report View ID** Select the view or report type as created in the OBIEE application.

## Selecting a PeopleSoft Query Data Source

Access the Select Data Source page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 2).

Select *PS Query* in the Data Type field.

**Pagelet Wizard** Step 2 of 6

1 2 3 4 5 6 < Previous Next >

### Select Data Source

Select the type of data and specify the source which contains the data you want displayed in your pagelet.


New Construction Update

\*Data Type

**Description**

PeopleSoft Query allows users to create database queries on records and fields within the PeopleSoft system. These queries can include customized parameters, or bind variables. Query Manager is the utility which allows users to create, customize and secure these queries.

**Data Source**

\*Query Name:  

**Data Source Details**

Inputs	
.MAXROWS	Max Rows

Outputs	
MESSAGE_SET_NBR	Set
DESCRSHORT	Short Desc
DESCR	Description

Select Data Source page (PS Query data source)

Queries typically return a number of columns and rows. Queries for pagelets should be selected based on available display space on the homepage. In addition, avoid selecting long-running queries that can delay the display of the homepage.

Keep in mind that public query results are available to portal viewers only if they have appropriate access to the underlying data record. Permission lists in the users' role definitions must contain query access groups that include the records in the query definition.

**Data Type** Select *PS Query*.

### **Data Source**

**Query Name** Select a query to serve as the data source for the pagelet. You can select any public query in the local database.

PeopleSoft queries provide SQL access to any table recognized by the PeopleSoft system. This includes external, remote tables that are recognized by the PeopleSoft system by way of a database link.

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Query*, "Creating and Running Simple Queries."

---

**Note.** Even though a query is marked as public, PeopleSoft Query still applies security to the underlying records. Ensure that the users of this pagelet have query security access to the underlying records to avoid runtime errors.

---

### **Data Source Details**

**Inputs** Displays the data source parameter fields that correspond to the query's input prompts. Also displays the required system-supplied .MAXROWS query data source parameter that you can use to limit the amount of data returned to the pagelet.

**Outputs** Displays the data source parameter fields that correspond to the query's output.

## **Selecting a Rowset Data Source**

Access the Select Data Source page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 2).

Select *Rowset* in the Data Type field.

**Pagelet Wizard**
**Step 2 of 6**

1

2

3

4

5

6

< Previous

Next >

## Select Data Source

Select the type of data and specify the source which contains the data you want displayed in your pagelet.

**New Construction Update**

**\*Data Type** Rowset v

**Description**

Rowsets can exist within components. This data type is geared toward embedded pagelets.

**Data Source**

**\*Record Name:** ACCESS\_GRP\_TBL 🔍

Select Data Source page (Rowset data source)

A rowset is a record that collects rows of data and can contain data from one or more records. The rowset data source is geared towards supplying data to embedded pagelets. This is because embedded pagelets require that a populated rowset be passed into the data source for it to be able to display the data as a chart or table, for example.

Embedded pagelets do not allow for customizable parameters that enable criteria selection. Consequently, there is also no end user personalization for rowset data source pagelets.

When creating a rowset data source pagelet you skip step 3 and go on to step 4 in which you select your pagelet display format.

### **Data Source**

**Record Name** Select the record that contains the layout of the rowset you want to use to supply your pagelet with data. Records available for selection are records that are tables or views.

## **Selecting a Search Record Data Source**

Access the Select Data Source page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 2).

Select *Search Record* in the Data Type field.

## Pagelet Wizard

## Step 2 of 6



&lt; Previous

Next &gt;

## Select Data Source

Select the type of data and specify the source which contains the data you want displayed in your pagelet.

Facilities Page

**\*Data Type**

**Description**  
The Search Record Data Source displays the search box or the search results from a given component.

**Data Source**

**\*Portal Name:**

**\*Content Reference:**

**\*Access Type:**

**Data Source Details**

Inputs	
MESSAGE_SET_NBR	Message Set Number
.MAXROWS	Max Rows
SHOWRESULTS	Show Results

Outputs	
Message Set Number	
Description	

Select Data Source page (search record data source)

Search records typically return a number of columns and rows. When you select search records to be displayed on pagelets, consider the available display space on the homepage.

**Data Type**                      Select *Search Record*.

**Data Source**

**Portal Name**                      Select the portal in which the search record for the pagelet resides.

**Content Reference**              Select the content reference for the search record.

**Access Type**

Select the type of access for the search record data. Options are:

*Basic Search Record:* Select to have the search record pagelet retrieve data based on the basic search record defined in the selected portal.

*CI: <component interface name>:* Select when the basic search record does not provide the required functionality; for example, when security or PeopleCode considerations need to be taken into account. This option is available if the selected search record has an associated component interface. Select to have the search record pagelet retrieve data by way of the component interface.

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Component Interfaces*, "Developing Component Interfaces."

**Data Source Details****Inputs**

Displays search record key input fields, as well as other system-required input fields that are used to derive data that is displayed on the search record pagelet.

**Outputs**

Displays search record key output fields, as well as other output fields for which search results are displayed on the search record pagelet. These output fields are derived from the drop-down list box fields on the search record.

If the SHOWRESULTS data source parameter field for the pagelet is set to *N*, the search key output fields appear as Search By drop-down list values on the pagelet.

**Selecting a URL Data Source**

Access the Select Data Source page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 2).

Select *URL* in the Data Type field.

Pagelet Wizard
Step 2 of 6

1
2
3
4
5
6

< Previous
Next >

## Select Data Source

Select the type of data and specify the source which contains the data you want displayed in your pagelet.

New Construction Update

\*Data Type URL ▼

▼ Description

The URL DataType retrieves data from external sources specified by a URL.

Data Source

URL: http://search.google.com/srch

Select Data Source page (URL data source)

**URL**

Enter any HTTP-accessible URL as the data source for the pagelet. The URL may or may not require query string parameters that can be specified and personalized as end-user prompts within Pagelet Wizard.

For example, the Yahoo! search page URL takes the form `http://search.yahoo.com/search?p=searchterm`. If you want to make `searchterm` a user-specified data parameter, enter a URL field value of `http://search.yahoo.com/search`. In step 3 on the Specify Data Source Parameters page, add a data source parameter Field Name value of `p` and set its Usage Type field value to *User Specified* with a default value. Pagelet Wizard then adds the `?p=searchterm` value, with the `searchterm` value able to be personalized by the user, to the URL before it retrieves the content to the pagelet.

---

## Step 3: Specifying Pagelet Data Source Parameters

This section provides an overview of variables supported as data source parameters, lists common elements, and discusses specifying pagelet data source parameters. This is step 3 in the Pagelet Wizard pagelet creation process. Specifically, this section discusses how to:

- Specify HTML data source parameter values.
- Specify IB Connector data source parameters.

- Specify Integration Broker data source parameters.
- Specify Navigation Collection data source parameters.
- Specify OBIEE data source parameters.
- Specify query data source parameters.
- Specify rowset data source parameters.
- Specify search record data source parameters.
- Specify URL data source parameters.

## Understanding System Variables Supported as Data Source Parameters

When you set the Usage Type field to *System Variable* or *User Specified* on the Specify Data Source Parameters page, you can enter a system variable such as *%UserId*, which the system substitutes at runtime with the actual value represented by the variable. The following system variables are supported as data source parameters.

- %AuthenticationToken
- %ClientDate
- %ClientTimeZone
- %ContentID
- %ContentType
- %Copyright
- %Currency
- %Date
- %DateTime
- %DbName
- %DbServerName
- %DbType
- %EmailAddress
- %EmployeeId
- %HPTabName
- %IsMultiLanguageEnabled
- %Language
- %Language\_Base
- %LocalNode

- %Market%MaxMessageSize
- %NavigatorHomePermissionList
- %Node
- %PasswordExpired
- %Portal
- %PrimaryPermissionList
- %ProcessProfilePermissionList
- %RowSecurityPermissionList
- %RunningInPortal
- %ServerTimeZone
- %Time
- %UserDescription
- %UserId

Common Elements Used in This Section

Description	Displays a description of the data source parameter.
-------------	--

## Usage Type

Select the type of end-user accessibility that you want to grant for the data source parameter field when it appears on the pagelet. Options are:

- *Not Used*: Select if it will not be used on the pagelet.
- *Fixed*: Select to enter a fixed value for the data source parameter that the pagelet end user cannot modify.
- *System Variable*: Select to assign a system variable as the data source parameter value. The value of the system variable is automatically inserted into the parameter when the pagelet appears. When you select this option, the pagelet end user cannot modify the data source parameter.

For example, suppose that you specify *%UserId* as the system variable for a User data source parameter field. When the pagelet appears on a user's homepage, the User field is populated by the *%UserId* system variable, which is the user ID used to access the pagelet.

- *User Specified*: Select to enable end users to specify a data source parameter value for this field. When a pagelet contains a user-specified field, the Customize button appears on the pagelet title bar.

End users can click this button to access a personalization page, where they can select a data source parameter value that they want to appear by default on the pagelet when they access it. They can select a value from a prompt, or they can manually enter their own value if no prompt values are available.

If you change the usage type from or to *User Specified* for a data source parameter on a published homepage pagelet, you must unpublish and republish the pagelet.

- *Admin Specified*: Enables users with administrative privileges to specify variables for this field, as well as access the Configure link on the published pagelet and select from those parameters for users.
- *Context Sensitive*: Enables users to specify context sensitivity for OBIEE and Embedded pagelets to be used with Context Manager and Related Content Framework.

**Default Value**

If you select *User Specified* or *Admin Specified* as the usage type and the data source parameter is a required value, you must enter the default value that should appear before a user enters a value. If the data source parameter is not a required value, you do not have to enter a default value.

If you select a default value when defining available user-specified values on the Specify Data Source Parameter Values page, that default value populates this field.

If you select *User Specified* as the usage type, you can enter a system variable as the default value. For example, to make the current date the default value for a user-specified parameter, enter a default value of *%Date*.

If you select *System Variable* as the usage type, you must enter the system variable to use as the data source parameter value.

See [Chapter 11, "Using Pagelet Wizard," Understanding System Variables Supported as Data Source Parameters, page 325](#).

If you select *Fixed* as the usage type, you must enter the fixed value.

You can enter a value that includes the % and \* wildcards at the beginning or end of a value in the Default Value field.



If you select *Fixed* as the usage type and the data source parameter has associated prompt values, you can use the Look up Value button to search for available data source parameter values.

If you select *System Variable* as the usage type, you can use the Look up Value button to access a list of valid system variables.

If you select *User Specified* as the usage type, this button does not appear.

**User Values**

If you select *User Specified* as the usage type, click to access the Specify Data Source Parameter Values page.

See [Chapter 11, "Using Pagelet Wizard," Step 3: Specifying Pagelet Data Source Parameters, page 324](#).

## Page Used to Specify Pagelet Data Source Parameters

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Specify Data Source Parameters	PTPPB_WIZ_DATAPRMS	<p>PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard</p> <p>Click the Next button on the Select Data Source page, or click the icon for step 3.</p>	<p>Configure the data source parameters required for the data to be displayed on the pagelet. For instance, when entering data source parameters for a query pagelet, you specify input data that extracts output data to the pagelet. The parameters vary depending on the data source selected on the Select Data Source page.</p> <p>This page is accessible only when building a pagelet with a data source that allows modification of associated parameters. These data sources include PeopleSoft Query, search record, and Navigation Collections.</p> <p><b>Note.</b> This step is not applicable when creating pagelets with an HTML data source.</p>
Specify Data Source Parameter Values	PTPPB_WIZ_PRMVALS	Click the Values link on the Specify Data Source Parameters page.	<p>Enter values for the end user to select from when personalizing user-specific parameters on a pagelet.</p> <p><b>Note.</b> This page is accessible only for data source parameters for which you have selected the <i>User Specified</i> or <i>Admin Specified</i> usage types.</p>

## Specifying HTML Data Source Parameters

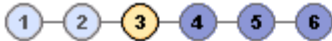
When creating an HTML data source pagelet you skip step 3 and go on to step 4 in which you select your pagelet display format.

## Specifying IB Connector Data Source Parameters

After selecting *IB Connector* as the data type on the Select Data Source page, access the Specify Data Source Parameter page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 3).

### Pagelet Wizard

### Step 3 of 6


[< Previous](#)
[Next >](#)

## Specify Data Source Parameters

Specify the parameters and their associated options specific to the data source you have selected for your pagelet. Rows showing a selected 'Required' require a Default Value.

Facilities

Data Source Parameters					
Field Name	Description	*Usage Type	Required	Default Value	
CLIENTCERT	CLIENTCERT	Not Used	<input type="checkbox"/>		<a href="#">Values</a>
DIRECTORY	DIRECTORY	Not Used	<input type="checkbox"/>		
ENCODING	ENCODING	Not Used	<input type="checkbox"/>		
FILENAME	FILENAME	Not Used	<input type="checkbox"/>		
FTPS	FTPS	Fixed	<input checked="" type="checkbox"/>	N	
HOSTNAME	HOSTNAME	Fixed	<input checked="" type="checkbox"/>		
METHOD	METHOD	Fixed	<input checked="" type="checkbox"/>	PUT	
PASSWORD	PASSWORD	Fixed	<input checked="" type="checkbox"/>		
PORT	PORT	Not Used	<input type="checkbox"/>	21	

Specify Data Source Parameters (IB Connector data source) (1 of 2)

**Personalization Instructions**

Specify the text that should appear on the personalization page for this pagelet.

**Text:**

[Reset to Default](#)

Specify Data Source Parameters (IB Connector data source) (2 of 2)

Define the data source parameters for the IB Connector data source. Click the User Values button to set the default values for users.

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Integration Broker*, "Understanding PeopleSoft Integration Broker."

See [Chapter 11, "Using Pagelet Wizard," Specifying User-Defined Data Source Parameters, page 339.](#)

## Specifying Integration Broker Data Source Parameters

After selecting *Integration Broker* as the data type on the Select Data Source page, access the Specify Data Source Parameter page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 3)..

### Pagelet Wizard

### Step 3 of 6


[< Previous](#)
[Next >](#)

## Specify Data Source Parameters

Specify the parameters and their associated options specific to the data source you have selected for your pagelet. Rows showing a selected 'Required' require a Default Value.

Facilities

Data Source Parameters					
Field Name	Description	*Usage Type	Required	Default Value	
MCFEM_REQ.MCF_USER		System Variable	<input checked="" type="checkbox"/>		<a href="#">Values</a>
MCFEM_REQ.MCF_UID_LIST		User Specified	<input type="checkbox"/>		
MCFEM_REQ.MCF_SERVER		Not Used	<input type="checkbox"/>		
MCFEM_REQ.MCF_PORT		Not Used	<input type="checkbox"/>		
MCFEM_REQ.MCF_PROTOCOL		Not Used	<input type="checkbox"/>		
MCFEM_REQ.MCF_NUMROWS		Admin Specified	<input checked="" type="checkbox"/>	10	
MCFEM_REQ.MCF_EMSZ_PART		Not Used	<input type="checkbox"/>		
MCFEM_REQ.MCF_EMSZ_CONN		Not Used	<input type="checkbox"/>		
MCFEM_REQ.MCF_EMAIL_LANG_CD		Not Used	<input type="checkbox"/>		
MCFEM_REQ.MCF_PASSWORD		Not Used	<input type="checkbox"/>		
MCFEM_REQ.MCF_METHODNAME		Not Used	<input type="checkbox"/>		
MCFEM_REQ.MCF_ATTROOT		Not Used	<input type="checkbox"/>		

Specify Data Source Parameters (Integration Broker data source)

This step is applicable for the Integration Broker data source if the service operation is based on a structured message or if the message is structured but contains a schema. The structure that is associated with the message is translated into data source parameters on this page.

If the message is unstructured, then no schema structure has been associated with the message. Therefore, no data source parameters can be supplied and Pagelet Wizard bypasses this step.

You must supply values for all rows with the Required option selected.

## Personalization Instructions

Use the Personalization Instructions text box to enter custom personalization instructions for the pagelet. These personalization instructions appear on the Personalization page of the pagelet.

**Note.** Personalization instructions must be translatable.

## Specifying Navigation Collection Data Source Parameters

After selecting *Navigation Collection* as the data type on the Select Data Source page, access the Specify Data Source Parameter page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 3).

**Pagelet Wizard** **Step 3 of 6**

1 2 3 4 5 6 < Previous Next >

### Specify Data Source Parameters

Specify the parameters and their associated options specific to the data source you have selected for your pagelet. Rows showing a selected 'Required' require a Default Value.  
New Construction Update

Field Name	Description	*Usage Type	Required	Default Value	
.INCLUDECHILDREN	Include Children	Fixed	<input checked="" type="checkbox"/>	Y	<a href="#">Values</a>
.MAXCHILDLINKS	Maximum Child I	Fixed	<input checked="" type="checkbox"/>	10	

[Reset to Default](#)

Specify Data Source Parameters (Navigation Collection data source)

### Data Source Parameters

Pagelet Wizard analyzes the selected query and retrieves a list of predefined parameters and any related prompt list values. You cannot add to the list of parameters.

<b>Field Name</b>	Displays the following data source parameters required by the Navigation Collection data source.
	.INCLUDECHILDREN: Determines whether the pagelet displays Navigation Collection child links.
	.MAXCHILDLINKS: If the pagelet is defined to display child links, determines the maximum number of Navigation Collection child links to display.

- Required** This option is selected if the data source parameter must contain a value on the pagelet to retrieve data successfully. For example, the .INCLUDECHILDREN field is required.
- If this option is selected and display-only, the value is set in the application class at the API level.
- Reset to Default** Click to reset the values on this page to the default values specified by the Navigation Collection data source API.
- Values** Click the [Values](#) link to set the default values for users.
- See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Integration Broker*, "Understanding PeopleSoft Integration Broker."

### Personalization Instructions

Use the Personalization Instructions text box to enter custom personalization instructions for the pagelet. These personalization instructions appear on the Personalization page of the pagelet.

---

**Note.** Personalization instructions must be translatable.

---

## Specifying OBIEE Data Source Parameters

After selecting *OBIEE Report* as the data type on the Select Data Source page, access the Specify Data Source Parameter page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 3).

**Pagelet Wizard** **Step 3 of 6**

1 — 2 — **3** — 4 — 5 — 6 < Previous    Next >

### Specify Data Source Parameters

Specify the parameters and their associated options specific to the data source you have selected for your pagelet. Rows showing a selected 'Required' require a Default Value.

Market by Region

▼ **Data Source Parameter Details**

**Report Filter Expression:**

Markets.Region is prompted

Data Source Parameters					Find	First	1 of 1	Last
Field Name	Description	*Usage Type	Required	Default Value				
1) Markets.Region	Markets.Region	Fixed	<input checked="" type="checkbox"/>	C				

Reset to Default

Specify Data Source Parameter page (OBIEE Report data source)

## Data Source Parameters

Pagelet Wizard analyzes the selected OBIEE report and retrieves a list of predefined parameters and any related prompt list values. You cannot add to the list of parameters.

<b>Field Name</b>	Displays the data source parameters that correspond to the OBIEE prompts associated with the selected OBIEE report. Also displays the required system-supplied .MAXROWS query data source parameter that you can use to limit the amount of data returned on the pagelet.
<b>Description</b>	This field contains a description that identifies the purpose or location of the OBIEE field.
<b>Required</b>	<p>This option is selected if the data source parameter must have a value entered on the pagelet to retrieve data successfully. For example, the .MAXROWS field is required.</p> <p>If this option is selected and display-only, the value is set in the application class at the application programming interface (API) level.</p> <p>If you leave this option cleared, the data source parameter value can be clear at runtime, or it can be entered to refine the data selection criteria.</p>
<b>Reset to Default</b>	Click to reset the values on this page to the default values specified by the OBIEE data source API.

## Specifying Query Data Source Parameters

After selecting *PS Query* as the data type on the Select Data Source page, access the Specify Data Source Parameter page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 3).

### Pagelet Wizard

### Step 3 of 6


[< Previous](#)
[Next >](#)

## Specify Data Source Parameters

Specify the parameters and their associated options specific to the data source you have selected for your pagelet. Rows showing a selected 'Required' require a Default Value.

New Construction Update

Data Source Parameters				
Field Name	Description	*Usage Type	Required	Default Value
ATTRIBUTE_NUM	Attribute Name	Fixed	<input checked="" type="checkbox"/>	
.MAXROWS	Max Rows	Fixed	<input checked="" type="checkbox"/>	10

[Reset to Default](#)

Specify Data Source Parameters page (PeopleSoft Query data source)

## Data Source Parameters

Pagelet Wizard analyzes the selected query and translates the query prompts into the list of pagelet parameters shown in this step. You cannot add to the list of parameters. Most prompts will be marked as required; however, any prompts that are used exclusively in the IN clause of the query will not be marked as required.

---

**Important!** Query Prompts that are based on dynamic views should not be used as User Specified data source parameters. Unexpected results will occur.

---

It is possible to attain advanced functionality with proper design of the query and the Pagelet Wizard pagelet. For example, you might want to create a pagelet that will show open purchase orders for a list of vendors, and allow each user to specify up to five vendors. To achieve this functionality, you would first create a query to retrieve the open purchase orders for up to five vendors. When you create this query, ensure that the five vendor ID prompts are used only in the IN clause of the query. You can then create a pagelet using the PS Query data type and choose the query you have created. The five vendor ID prompts will be shown as data source parameters. However, they will be marked as not required. It will then be possible for end users to specify any number of vendors (up to five) when they personalize the pagelet. Finally, update the personalization instructions on the pagelet to explain to the end users that they may supply up to five vendor IDs.

<b>Field Name</b>	Displays the data source parameters that correspond to the query prompts associated with the selected query. Also displays the required system-supplied .MAXROWS query data source parameter that you can use to limit the amount of data returned on the pagelet.
<b>Required</b>	<p>This option is selected if the data source parameter must have a value entered on the pagelet to retrieve data successfully. For example, the .MAXROWS field is required.</p> <p>If this option is selected and display-only, the value is set in the application class at the application programming interface (API) level.</p> <p>If you leave this option cleared, the data source parameter value can be clear at runtime, or it can be entered to refine the data selection criteria.</p> <p>Prompts tied to the IN clause of the query will not be marked as required.</p>
<b>Reset to Default</b>	Click to reset the values on this page to the default values specified by the query data source API.
<b>Values</b>	Click the <a href="#">Values</a> link to set the default values for users.

## Personalization Instructions

Use the Personalization Instructions text box to enter custom personalization instructions for the pagelet. These personalization instructions appear on the Personalization page of the pagelet.

---

**Note.** Personalization instructions must be translatable.

---

## Specifying Rowset Data Source Parameters

When creating a rowset data source pagelet you skip step 3 and go on to step 4 in which you select your pagelet display format.

## Specifying Search Record Data Source Parameters

After selecting *Search Record* as the data type on the Select Data Source page, access the Specify Data Source Parameter page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 3).

**Pagelet Wizard** **Step 3 of 6**

1 2 3 4 5 6 < Previous Next >

### Specify Data Source Parameters

Specify the parameters and their associated options specific to the data source you have selected for your pagelet. Rows showing a selected 'Required' require a Default Value.  
New Construction Update

Field Name	Description	*Usage Type	Required	Default Value	
OPRID	User ID	Not Used	<input type="checkbox"/>		
.MAXROWS	Max Rows	Fixed	<input checked="" type="checkbox"/>	20	
SHOWRESULTS	Show Results	Fixed	<input checked="" type="checkbox"/>	Y	<a href="#">Values</a>

[Reset to Default](#)

Specify Data Source Parameters (search record data source)

### Data Source Parameters

Pagelet Wizard analyzes the selected search record and retrieves a list of search keys required for the search record. You cannot add to the list of data source parameters.

For fields that are search keys but that are not displayed on the search prompt page or in the result set for the search record (for example, the setID, Business Unit, Operator ID, and Operclass fields), Pagelet Wizard assimilates the PeopleTools behavior and does not display the keys during design or at runtime. It simply binds in the same values that PeopleTools would use. These values are derived from the default field values established by the user in the PSRECDEFN table.

<b>Field Name</b>	<p>Displays the data source parameters that correspond to the search keys required for the search record. Also displays the required system-supplied .MAXROWS and SHOWRESULTS search record data source parameters.</p> <p>Use the .MAXROWS field to limit the amount of data returned to the pagelet.</p> <p>Use the SHOWRESULTS field to determine whether search results appear on the pagelet.</p> <p>If this field value is set to <i>Y</i>, search record results appear directly on the pagelet with links to transaction data.</p> <p>In addition, if this field value is set to <i>Y</i>, the search results grid displays results based on the supplied data source parameter default key field values. If no defaults are supplied, all results appear, up to the .MAXROWS value defined for the pagelet.</p> <p>If this field value is set to <i>N</i>, the pagelet displays a search prompt for the selected search record. The Search By drop-down list values available on the pagelet are the search keys for the search record.</p> <p>If data source parameter default key field values have been defined, when the pagelet is initially accessed, the default value appears in the begins with pagelet field.</p>
<b>Required</b>	<p>This option is selected if the data source parameter must have a value entered on the pagelet to retrieve data successfully. For example, the .MAXROWS and SHOWRESULTS fields are required.</p> <p>If this option is selected and display-only, the value is set in the application class at the API level.</p> <p>Search key values are not required because the search record pagelet supports partially populated keys so that you can define the pagelet to return all possible results.</p> <p>If you leave this option clear, the data source parameter value can be blank at runtime, or the value can be provided to refine the data selection criteria.</p>
<b>Reset to Default</b>	<p>Click to reset the values on this page to the default values specified by the search record data source API.</p>
<b>Values</b>	<p>Click the Values link to set the default values for users.</p>

### ***Personalization Instructions***

Use the Personalization Instructions text box to enter custom personalization instructions for the pagelet. These personalization instructions appear on the Personalization page of the pagelet.

---

**Note.** Personalization instructions must be translatable.

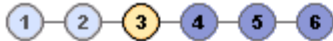
---

## **Specifying URL Data Source Parameters**

After selecting *URL* as the data type on the Select Data Source page, access the Specify Data Source Parameter page.(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 3).

## Pagelet Wizard

Step 3 of 6



&lt; Previous

Next &gt;

## Specify Data Source Parameters

Specify the parameters and their associated options specific to the data source you have selected for your pagelet. Rows showing a selected 'Required' require a Default Value.

Facilities Page

*Field Name	Description	*Usage Type	Required	Default Value		
.TIMEOUT_MS	Time Out (Millisec)	Not Used	<input type="checkbox"/>		+	-
.PASSCOOKIES	Do not pass cookies	Fixed	<input checked="" type="checkbox"/>	False	+	-

**Personalization Instructions**  
Specify the text that should appear on the personalization page for this pagelet.  
**Text:**

Reset to Default

Specify Data Source Parameters page (URL data source)

The data source parameters that you can define on this page correspond to URL query parameters, such as an HTTP GET request.

An HTTP GET request is a URL type for which parameters are placed on the query string separated by ? and &, such as <http://search.yahoo.com/search?p=searchterm>.

For example, the Yahoo! search page URL takes the form: <http://search.yahoo.com/search?p=searchterm>. If you want to make searchterm a user-specified data parameter, enter a URL field value of <http://search.yahoo.com/search> in step 2 on the Select Data Source page. In step 3 on this page, add a data source parameter Field Name value of *p* and set its Usage Type field value to *User Specified* with a default value. Pagelet Wizard then adds the *?p=searchterm* value, with the *searchterm* value able to be personalized by the user, to the URL before it retrieves the content to the pagelet.

You can specify a parameter for the number of milliseconds that a data source will wait for a response before timing out on the homepage. When you set *.timeout\_ms* you can then enter the number of milliseconds that the system will wait.

**Passing Cookies**

When the URL request is in the AuthenticationToken domain, then the URL data source will pass all cookies on the request to retrieve the content. (The only exception to this rule is if the request is being made back to the local webserver. In that instance, only the PS\_TOKEN cookie will be passed). Any requests to servers outside of the AuthTokenDomain will not have any cookies passed.

To turn off the passing of cookies other than PS\_TOKEN, add the parameter .passcookies as a Fixed usage type with a default value of False. Adding this parameter with a value of False will instruct pagelet wizard to only pass the PS\_TOKEN cookie (and no other cookies) for requests within the AuthenticationToken domain. Adding the .passcookies parameter with a value of True is not recommended, since this is the same as the default behavior.

---

**Note.** Make sure to examine any security impact that passing cookies might present in your configuration before setting this parameter.

---

### Personalization Instructions

Use the Personalization Instructions text box to enter custom personalization instructions for the pagelet. These personalization instructions appear on the Personalization page of the pagelet.

---

**Note.** Personalization instructions must be translatable.

---

## Specifying User-Defined Data Source Paramters




Click the User Values button to display the Specify Data Source Parameters page.

### Pagelet Wizard

#### Specify Data Source Parameter Values

List the parameter values available for a user to select for the Data Source Parameter Name.

**Field Name:** LANGUAGE\_CD      Lang Cd

Parameter Values			
		Find   View All   	First  1-25 of 26  Last
	Parameter Value	Description	Default
1	ARA	Arabic	<input type="checkbox"/>
2	CFR	Canadian French	<input type="checkbox"/>
3	CZE	Czech	<input type="checkbox"/>
4	DAN	Danish	<input type="checkbox"/>
5	DUT	Dutch	<input type="checkbox"/>
6	ENG	English	<input type="checkbox"/>
7	ESP	Spanish	<input type="checkbox"/>
8	FIN	Finnish	<input type="checkbox"/>
9	FRA	French	<input checked="" type="checkbox"/>
10	GER	German	<input type="checkbox"/>
11	GRK	Greek	<input type="checkbox"/>
12	HEB	Hebrew	<input type="checkbox"/>

Specify Data Source Parameter Values page

Select the parameter value to be displayed to the user as the default value for the field. Users can select from this list of valid values when they personalize the pagelet.

---

## Step 4: Selecting a Pagelet Display Format

This section discusses how to select a display format. This is step 4 in the Pagelet Wizard pagelet creation process.

### Page Used to Select a Pagelet Display Format

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Select Display Format	PTPPB_WIZ_DISPFRMT	PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard  Click the Next button on the Specify Data Source Parameters page, or click the icon for step 4.	Specify the data transformation method and display format for the pagelet.

### Selecting a Display Format

Access the Select Display Format page (PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 4).

## Pagelet Wizard

## Step 4 of 6







&lt; Previous

Next &gt;

## Select Display Format

Select the format in which you would like your pagelet data rendered.

Welcome to the page

Specify Display Options		First	1-4 of 4	Last
	Name	Description		
<input checked="" type="radio"/>	 <b>Table</b>	Display your pagelet data in tabular format, with customizable columns, visual display and ordering		
<input type="radio"/>	 <b>List</b>	Display your pagelet data as a numbered or bulleted list		
<input type="radio"/>	 <b>Chart</b>	Display your pagelet data as line, bar, pie or histogram chart, complete with customizable display options and drilldown capabilities		
<input type="radio"/>	 <b>Custom</b>	Specify your own custom display transformation (XSL template) for your pagelet		

### Select Display Format page

Select the format to use for the pagelet. The display formats available for selection on this page depend on the data source you have selected on the Select Data Source page. Options are:

**Table** Available for query and rowset data source pagelets.

**List** Available for query and rowset data source pagelets.

**Chart** Available for query and rowset data source pagelets.

**Menu** Available for Navigation Collection data source pagelets.

**Search List** Available for search record data source pagelets.

**Passthru** Available for HTML and OBIEE source pagelets.

**Custom** Available for HTML, Navigation Collection, query, rowset, and search record data source pagelets.

### See Also

[Chapter 11, "Using Pagelet Wizard," Step 5: Specifying Pagelet Display Options, page 342](#)

---

## Step 5: Specifying Pagelet Display Options

This section lists common elements and discusses specifying pagelet display options. This is step 5 in the Pagelet Wizard pagelet creation process. Specifically, this section discusses how to:

- Specify table display options.
- Specify passthru display options.
- Specify search list display options.
- Specify menu display options.
- Specify list display options.
- Specify chart display options.
- Specify custom display options.
- Specify data link details.
- Specify URL and newsfeed options.

### Common Elements Used in This Section

This section discusses common elements used in this section.



Click the Link Options icon to access the Link Details page, where you can enter values that link elements in the pagelet to pages (components or URLs) that provide details about the data.

See [Chapter 11, "Using Pagelet Wizard," Specifying Data Link Details, page 362.](#)

#### ***Additional Text***

The Additional Text group box contains options to add headers, footers, opening text, and closing text to a pagelet.

#### **Header**

Select a pagelet header. You can select from headers that are defined on the Define Header page.

See [Chapter 11, "Using Pagelet Wizard," Defining Headers, page 303.](#)

**Opening Text**

Select the type of opening text that you want to appear on the pagelet. Options are:

*Freeform Text/HTML:* The Text field appears. Enter the opening text that appears on the pagelet.

*Message Catalog Entry:* The Message Set and Number fields appear. Select the message set and message number that appear as opening text on the pagelet. You can select from the messages defined in the database.

See *Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*, "Using PeopleTools Utilities," Message Catalog.

**Show "View Source Data" Link**

Displays when *PS Query* is selected as the data source for a pagelet. Select to display the Full Query Results link on the pagelet. Click the Full Query Results link to access the full results of the query used for the pagelet.

**Closing Text**

Select the type of closing text that appears on the pagelet. Options are:

*Freeform Text/HTML:* The Text field appears. Enter the closing text that appears on the pagelet.

*Message Catalog Entry:* The Message Set and Number fields appear. Select the message set and message number that appear as closing text on the pagelet. You can select from the messages defined in the database.

See *Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*, "Using PeopleTools Utilities," Message Catalog.

**Footer**

Select a pagelet footer. You can select from footers that are defined on the Define Footer page.

See Chapter 11, "Using Pagelet Wizard," Defining Footers, page 306.

**Search Options**

The Search Options group box contains the options to override the default search functionality for the current pagelet only.

See Chapter 11, "Using Pagelet Wizard," Search Application Class, page 298.

**Search Box**

Select a value from these options to determine the location of the search box:

- *None* – Do not show the search box. This is the default option.
- *Top Left* – The search box appears at the top left corner of the pagelet, above the pagelet content but below any header and opening text.
- *Top Right* – The search box appears at the top right corner of the pagelet, above the pagelet content but below the header and opening text.
- *Bottom Left* – The search box appears at the bottom left corner of the pagelet, below the pagelet content but above any footer and closing text.
- *Bottom Right* – The search box appears at the bottom right corner of the pagelet, below the pagelet content but above any footer and closing text.
- *Inline XSL* – A PSSEARCHBOX node must be written into the transformation XSL to indicate where the search box appears. If there are multiple PSSEARCHBOX nodes in the pagelet content after the transformation, only the first node is replaced by the Search box. The PASSTHRU display format does not support this type of search box. If the display format is PASSTHRU, no search box appears in the pagelet.

**Package**

Enter the application class package name that you want to use to define and build the search for the pagelet.

**Path**

Enter the application class subpackage name that you want to use to define and build the search for the pagelet.

**Class ID**

Enter the application class name that you want to use to define and build the search for the pagelet. The class must exist in the application package and subpackage that you specify.

---

**Note.** If you have not defined default Search Option values for the data type on the Define Data Types page (PeopleTools, Portal, Pagelet Wizard, Define Data Types), the Package, Path, and Class ID fields have no default values. You must enter values if you want to activate the pagelet search feature.

---

**Customization**

The Customization group box contains the option to customize your pagelet.

## Customize

Click to customize the pagelet by applying a custom XSL template, or by directly editing the XSL resulting from the pagelet display options you have defined so far. When you click Customize, you switch from a non-custom display format selected on the Select Display Format page, to a custom format of your definition.

The XSL field is populated based upon preexisting display options set for the original non-custom display format. You can edit the XSL directly in the XSL field and retain as much of the original non-custom display options as you wish. Keep in mind that if you apply an XSL template using the XSL Template field, the XSL is regenerated based on the selected template and original non-custom display options you have set may be overridden. Making any edits to the XSL, either directly or by assigning a template, disconnects the pagelet definition from the original non-custom display format.


When you click the Customize button, the current pagelet definition is saved for restoration purposes. You can revert to this pre-customization pagelet definition by clicking the Undo Customizations button.

Clicking this button assigns your pagelet to the Custom design format on the Select Display Format page - Step 4. When you return to the Specify Publishing Options page - Step 5, you will access the version of the page for the Custom display format.

See [Chapter 11, "Using Pagelet Wizard," Specifying Custom Display Options, page 360.](#)

## Pages Used to Specify Pagelet Display Options

<b>Page Name</b>	<b>Definition Name</b>	<b>Navigation</b>	<b>Usage</b>
Specify Display Options (table format)	PTPPB_WIZ_DISP_TBL	PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard  Select <i>Table</i> as the display format and click the Next button on the Select Display Format page, or click the icon for the step 5.	Enter pagelet table formatting details and header and footer options. Preview the pagelet.
Set Thresholds	PTPPB_WIZ_TRSHVALS	Select the Set Thresholds button in the Table Options of Step 5 in Pagelet Wizard.	Specify the default threshold options for tables in pagelets.
Specify Display Options (passthru format)	PTPPB_WIZ_DISP_PST	PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard  Select <i>Passthru</i> as the display format and click the Next button on the Select Display Format page, or click the icon for step 5.	Enter pagelet passthru formatting details and header and footer options. Preview the pagelet.

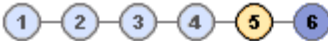
<b>Page Name</b>	<b>Definition Name</b>	<b>Navigation</b>	<b>Usage</b>
Specify Display Options (search list format)	PTPPB_WIZ_DISP_SRC	PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard  Select the <i>Search List</i> as the display format and click the Next button on the Select Display Format page, or click the icon for step 5.	Enter pagelet search list formatting details and header and footer options. Preview the pagelet.
Specify Display Options (menu format)	PTPPB_WIZ_DISP_NAV	PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard  Select <i>Menu</i> as the display format and click the Next button on the Select Display Format page, or click the icon for step 5.	Enter pagelet menu formatting details and header and footer options. Preview the pagelet.
Specify Display Options (list format)	PTPPB_WIZ_DISP_LST	PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard  Select <i>List</i> as the display format and click the Next button on the Select Display Format page, or click the icon for step 5.	Enter pagelet list formatting details and header and footer options. Preview the pagelet.
Specify Display Options (chart format)	PTPPB_WIZ_DISP_CHT	PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard  Select <i>Chart</i> as the display format and click the Next button on the Select Display Format page, or click the icon for step 5.	Enter pagelet chart formatting details and header and footer options. Preview the pagelet.
Specify Display Options (custom format)	PTPPB_WIZ_DISP_CUS	PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard  Select <i>Custom</i> as the display format and click the Next button on the Select Display Format page, or click the icon for step 5.	Enter pagelet custom formatting details and header and footer options. Preview the pagelet.
Link Details	PTPPB_WIZ_LINK_DTL	Click the Link Options  icon on the Specify Display Options page.	Enter linking details that convert pagelet column data into drill-down links to pages that provide details about the pagelet data.

## Specifying Table Display Options

After selecting the *Table* option on the Select Display Format page, select the Specify Display Options page (PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 4).

## Pagelet Wizard

## Step 5 of 6



&lt; Previous

Next &gt;

## Specify Display Options

Specify the visual options related to the display format for your pagelet.

## New Construction Update

Set	Short Desc	Description
1	Title Bar	PeopleTools Message Bar Items
2	PeopleCode	PeopleCode
3	GEN	General Tools Messages
4	HPR	Help Processor
5	HTM	Help Text Manager
6	IMP	Import Definition Manager
7	OPNQRY	Open Query API
8	PCEdit	PeopleCode Editor
9	MDM	Menu Definition Manager
10	CCONTROL	Change Control

Specify Display Options page (table display format)

**Table Options**

Select options that will display grid lines, alternate row shading, and row numbers on the table-formatted pagelet.

## Columns

**Column Name** Displays possible column headings provided by the selected data source parameters. Use the drop-down list boxes to set the order in which the columns appear on the table-formatted pagelet. Use the Add button or the Delete button to add or remove columns.

**Alignment** Specify the alignment of the text in the column. Options are *Center*, *Left*, and *Right*.



### Link Options

Click the Link Options icon to display the Link Details page where you can set paths, parameters, and behavior of links.

See [Chapter 11, "Using Pagelet Wizard," Specifying Data Link Details, page 362.](#)

## Setting Thresholds

Access the Set Thresholds page by clicking the Set Thresholds button.

### Pagelet Wizard

## Set Thresholds

Specify the default threshold options. Check "Allow User Thresholds" to allow users of this pagelet to individually set threshold values. Settings for 'Allowed Comparators' will affect the comparator values that are available for the Default Thresholds and end user's personalization pages.

▼ Allowed Comparators

Select 'Number Only' or 'Text Only' in order to limit the available comparator options.

Column Name	*Allowed Comparators
Users	All
Portal Label	All

☐ Allow User Thresholds

Default Thresholds

	*Column Name	*Comparator	*Value	*Highlight	*Color				
1						↑	↓	+	-

OK

Cancel

Set Thresholds page

Use the Set Thresholds page to set the column alignment and threshold options.

**Allowed Comparators** The system displays the column names and enables you to set the comparators for that column based on numbers, text, or both.

**Allow User Thresholds**     Select this check box to enable users to set their own threshold values for the table from the Personalization page.

Set the display order and attributes of each column. Use the up and down arrows to change the display order of the columns, if necessary. Each column threshold can have a unique color associated with it—duplicates are allowed. If more than one threshold rule applies for a given value, the rule that is positioned highest in the list is the one that is applied.


**Column Name**                     Select the column name from the drop-down list box.

**Comparator**                     Select a comparator value for each column

**Value**                             Enter the threshold value for the column. When this value is reached, the highlight and color attributes become active.

**Highlight**                        If you choose to have highlighting for the threshold, select either the entire row or just a single cell.

**Color**                             Select the color for the column.

                             Use the arrows to move each individual column up or down in the table.

**Pagelet Preview**

Displays a preview of the pagelet based on most display option settings on this page.

**Specifying Passthru Display Options**

After selecting the *Passthru* option on the Select Display Format page, select the Specify Display Options page(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 5).

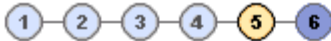
---

**Note.** For OBIEE Reports, *Passthru* is the only display format option.

---

## Pagelet Wizard

Step 5 of 6



&lt; Previous

Next &gt;

## Specify Display Options

Specify the visual options related to the display format for your pagelet.

## Market by Region

Additional Text	Pagelet Preview																	
<p>Header <input type="text"/></p> <p>Opening Text <input type="text" value="Freeform Text/HTML"/></p> <p>Text <input type="text" value="Text entered here appears directly ABOVE pagelet details."/></p> <p>Closing Text <input type="text" value="Message Catalog Entry"/></p> <p>Message Set <input type="text" value="21000"/> <input type="button" value="Search"/> Number <input type="text" value="1"/></p> <p>Footer <input type="text"/></p>	<p>Text entered here appears directly ABOVE pagelet details.</p>																	
<p>Search Options</p> <p>Search is supported for homepage pagelets and embeddable pagelets only.</p> <p>*Search <input type="text" value="No Search Box"/></p> <p>Box <input type="button" value="Custom Search Class"/></p>	<p><b>Market Detail</b></p> <table border="1"> <thead> <tr> <th>Region</th> <th>Market</th> </tr> </thead> <tbody> <tr><td rowspan="12">CENTRAL REGION</td><td>CHICAGO</td></tr> <tr><td>CINCINNATI</td></tr> <tr><td>COLUMBUS</td></tr> <tr><td>DES MOINES</td></tr> <tr><td>DETROIT</td></tr> <tr><td>GRAND RAPIDS</td></tr> <tr><td>INDIANAPOLIS</td></tr> <tr><td>KANSAS CITY</td></tr> <tr><td>LOUISVILLE</td></tr> <tr><td>MILWAUKEE</td></tr> <tr><td>MINNEAPOLIS</td></tr> <tr><td>OKLAHOMA CITY/TULSA</td></tr> <tr><td>OMAHA</td></tr> <tr><td>ST. LOUIS</td></tr> </tbody> </table>	Region	Market	CENTRAL REGION	CHICAGO	CINCINNATI	COLUMBUS	DES MOINES	DETROIT	GRAND RAPIDS	INDIANAPOLIS	KANSAS CITY	LOUISVILLE	MILWAUKEE	MINNEAPOLIS	OKLAHOMA CITY/TULSA	OMAHA	ST. LOUIS
Region	Market																	
CENTRAL REGION	CHICAGO																	
	CINCINNATI																	
	COLUMBUS																	
	DES MOINES																	
	DETROIT																	
	GRAND RAPIDS																	
	INDIANAPOLIS																	
	KANSAS CITY																	
	LOUISVILLE																	
	MILWAUKEE																	
	MINNEAPOLIS																	
	OKLAHOMA CITY/TULSA																	
OMAHA																		
ST. LOUIS																		
	<p>This closing text is derived from from Message Set 21000, Message Number 1.</p>																	

Specify Display Options page (passthru display format)

**Additional Text****Header**

Select a value from this field to display a header below the Title bar, but above the Opening Text.

**Opening Text**

Select a value to include opening text *above* the pagelet details.

Select from these values:

- Freeform Text/HTML

Enter freeform text or paste in HTML in the Text field to display text or to render HTML in the pagelet.

- Message Catalog

Select *Message Catalog* to enter a message set and message number to derive the opening text from the message catalog.

**Text****Closing Text**

Select a value to include opening text *below* the pagelet details.

Select from these values:

- *Freeform Text/HTML*

Select this value to enter freeform text or paste in HTML in the Text field to display text or to render HTML in the pagelet.

- *Message Catalog*

Select this value to enter a message set and message number to derive the closing text from the message catalog.

**Footer**

Select a value from this field to display a footer above the Footer bar, but below the Closing Text.

***Pagelet Preview***

Displays a preview of the pagelet based on most display option settings on this page.

**Specifying Search List Display Options**

After selecting the *Search List* option on the Select Display Format page, select the Specify Display Options page(PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 4).

## Pagelet Wizard

Step 5 of 6



&lt; Previous

Next &gt;



## Specify Display Options

Specify the visual options related to the display format for your pagelet.

Search Pagelet

Additional_text	
Header	PeopleSoft Logo
Opening Text	
Closing Text	
Footer	Powered by PeopleSoft
Customization	
Customize	

Pagelet Preview	
	
Access Group	
<a href="#">APPLICATION ENGINE</a> <a href="#">APPLICATION MSG</a> <a href="#">BUSINESS COMPONENTS</a> <a href="#">BUSINESS INTERLINK</a> <a href="#">BUSPROC_DEFINITION</a> <a href="#">CHANGE CONTROL</a> <a href="#">COBOL_DEFINITION</a> <a href="#">CONTENT_DEFINITION</a> <a href="#">EDI MANAGER</a> <a href="#">EO_BAM_FRAMEWORK</a>	
<small>powered by</small> 	

Specify Display Options page (search list display format)

**Pagelet Preview**

Displays a preview of the pagelet based on most display option settings on this page.

**Specifying Menu Display Options**

After selecting the *Menu* option on the Select Display Format page, select the Specify Display Options page (PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 4).

## Pagelet Wizard

Step 5 of 6



&lt; Previous

Next &gt;

## Specify Display Options

Specify the visual options related to the display format for your pagelet.

NC Pagelet

Menu Options	Pagelet Preview
<input checked="" type="checkbox"/> Top Level Title <input type="checkbox"/> Top Level Description *Maximum Columns: 1 *Display Order: Left/Right <input type="checkbox"/> Alternate Row Shading *Parent Images: Large Images <input checked="" type="checkbox"/> Parent Descriptions *Child Display Type: List with Images	<b>PeopleSoft®</b> <b>Tax Center</b> <b>Tax Setup</b> Set up tax structures Sales and Use Tax Value Added Tax Intrastat Excise and Sales Tax IND <b>Related Setup</b> Set up tax-related structures in applications Asset Management Common Expenses General Ledger Procurement Sales Treasury <b>Tax Processing</b> Perform tax processing and reporting Sales and Use Tax Value Added Tax Intrastat Excise and Sales Tax IND
<b>Additional_text</b> Header: PeopleSoft Logo Opening Text: Closing Text: Footer:	
<b>Customization</b> <div>Customize</div>	

Specify Display Options page (menu display format)

**Menu Options****Top Level Title**

Select to have the menu pagelet display the top-level title defined in the underlying Navigation Collection.

**Top Level Description**

Select to have the menu pagelet display the top-level description defined in the underlying Navigation Collection.

- Maximum Columns** Select the number of columns to display (options are *1* through *4*) on the Navigation Collection menu pagelet. Narrow pagelets display contents in one column. Wide pagelets can display contents in up to four columns.
- See [Chapter 11, "Using Pagelet Wizard," Overview of Pagelet Wizard Display Sizes, page 286.](#)
- Display Order** Select the order in which you want Navigation Collection elements to appear on the menu pagelet. When you create a Navigation Collection, you define the order of folders and links in the collection using options in the Placement Properties group box on the Edit Folder page and the Edit Link page. Options are:
- Left to Right:* Select to display the elements from left to right on the menu pagelet.
- Top to Bottom.* Select to display the elements from top to bottom on the menu pagelet.
- Alternate Row Shading** Select to display alternate row shading on the menu-formatted pagelet.
- Parent Images** Select the type of images you want to display for parent-level Navigation Collection elements (folders) on the menu pagelet. When you create a Navigation Collection, the collection definition includes specified large and small images that appear for parent-level elements in the Navigation Collection. Options are:
- Large Images:* Select to have the menu pagelet display the large folder image defined for the underlying Navigation Collection.
- No Images:* Select to prevent parent-level images from appearing on the menu pagelet.
- Small Images:* Select to have the menu pagelet display the small folder image defined for the underlying Navigation Collection.
- Parent Descriptions** Select to have the menu pagelet display parent-level folder descriptions defined in the underlying Navigation Collection.
- Child Display Type** Select the display type for child links on the menu pagelet. Options are:
- Comma Delimited:* Displays the maximum number of child links separated by commas.
- List with Images:* Displays the maximum number of child links accompanied by icons. This icon is the small content image selected in the Navigation Collection system or registry options for the underlying Navigation Collection.
- Simple List:* Displays the maximum number of child links, one child link per row. No bullets, numbers, or icons appear in the list.

### **Pagelet Preview**

Displays a preview of the pagelet based on most display option settings on this page.

## See Also

Chapter 4, "Working With Navigation Pages," Navigation Collections, page 48

## Specifying List Display Options

After selecting the *List* option on the Select Display Format page, select the Specify Display Options page (PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 4).

**Pagelet Wizard**
**Step 5 of 6**

1

2

3

4

5

6

< Previous

Next >

### Specify Display Options

Specify the visual options related to the display format for your pagelet.

Query Pagelet

**▼ List Options**

\*List Display Type Bulleted List

List Field Name MESSAGE\_TEXT

☒ **Alternate-Row Shading**

**▼ Additional\_text**

Header PeopleSoft Logo

Opening Text

☒ **Show "View Source Data" Link**

Closing Text

Footer Powered by PeopleSoft

**▼ Customization**

Customize

**▼ Pagelet Preview**

PeopleSoft.

- Invalid name.
- Not all selected definitions were added to the "Included Definitions" list.
- Not all selected definitions made it into the "Excluded Definitions" list.

[Full Query Results](#)

powered by  
**PeopleSoft**

Specify Display Options page (list display format)

### List Options

#### List Display Type

Select the list format for displaying query results on the list pagelet. Options are:

*Bulleted List:* Displays query results in a bulleted list.

*Numbered List:* Displays query results in a numbered list.

*Simple List:* Displays query results in a list with no bullets or numbers.

**List Field Name**      Select the field whose values are displayed in the list on the pagelet.

**Alternate-Row Shading**      Select to use alternate-row shading on the list pagelet.

### ***Pagelet Preview***

Displays a preview of the pagelet based on most display option settings on this page.

## **Specifying Chart Display Options**

After selecting the *Chart* option on the Select Display Format page, select the Specify Display Options page (PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 4).

Specify Display Options

Specify the visual options related to the display format for your pagelet.

Popular Pagelet Chart

Chart Options

Chart Type

2D Bar Chart

X-Axis Field

PORTAL\_LABEL

X-Axis Title

X Label Angle

90

Y Axis Data

in Columns

Data is in multiple columns.

Y-Axis Fields

Y-Axis Field	Color			
COUNT(*)	Default		+	-

Y-Axis Title

Y Title Angle

Advanced Options

3D Angle

Legend

Height

Width

218

Pagelet Preview

PeopleSoft®

Popular pagelet usage.

Pagelet	Usage
Company News	1.0
custom	1.0
Enterprise Menu	1.0
Message Sets	1.0
Pagelet Wizard Home	1.0

[Full Query Results](#)

powered by PeopleSoft

Specify Display Options page (chart display format) (1 of 2)

**Overlay Field**

**Additional Text**

**Header** PeopleSoft Logo

**Opening Text** Freeform Text/HTML

**Text** Popular pagelet usage.

☒ **Show "View Source Data" Link**

**Closing Text**

**Footer** Powered by PeopleSoft

**Customization**

Customize

Specify Display Options page (chart display format) (2 of 2)

To generate a chart pagelet using Pagelet Wizard, you include information related to the chart's *x* axis and *y* axis. The *y* axis is the axis that contains the query data. In most charts, the *y* axis is the vertical axis. However, in a horizontal bar chart, the *y* axis is the horizontal axis. The *x* axis is the axis against which the *y*-axis data is measured.

For example, a chart pagelet may have a *y* axis that displays the number of times that a pagelet has been accessed. This access count can be displayed against an *x* axis that shows the name of the pagelet accessed.

## Chart Options

<b>Chart Type</b>	Select the type of chart used to display the query results on the pagelet. Available chart formats are those supported by the PeopleCode Chart class.  See <i>Enterprise PeopleTools 8.50 PeopleBook: PeopleCode API Reference</i> , "Charting Classes."
<b>X-Axis Field</b>	Select the field used to derive <i>x</i> -axis values for the chart. Available values are determined by the data source's output fields.
<b>X-Axis Label</b>	Select the field you want to use to derive the labels for the <i>x</i> -axis values for the chart. Available values are determined by the data source's output fields.
<b>X Label Angle</b>	Enter the angle at which you want the text of the <i>x</i> -axis labels to appear along the <i>x</i> -axis on the pagelet.

**Y Axis Data**

Select to indicate whether the y-axis data is stored in rows or columns in the selected query. This enables the Pagelet Wizard to read query data and generate the chart.

*in Columns:* Select if the y-axis data is stored in columns.

For example, you might have query data stored as shown below:

<b>Quarter</b>	<b>Revenue</b>	<b>Profit</b>
Q1	3562	1010
Q2	3473	1290
Q3	4568	1490
Q4	5668	2769

**Y Axis Data** (continued) *in Rows:* Select if the y-axis data to be used to generate the chart is stored in rows.

For example, you might have query data stored as shown below:

<b>Quarter</b>	<b>Type</b>	<b>Value</b>
Q1	Revenue	3562
Q1	Profit	1010
Q2	Revenue	3473
Q2	Profit	1290
Q3	Revenue	4568
Q3	Profit	1490
Q4	Revenue	5668
Q4	Profit	2769

**Y-Axis Field**

Select the field that you want to use to derive y-axis values for the chart. Available values are determined by the data source's output fields.

If you set the Y Axis Data field to *in Columns*, you can select up to four y-axis fields to create a chart pagelet that displays data for multiple fields. For example, if you track sales figures for several departments over multiple years, you can assign each department its own y-axis value.

**Color** Select the color for the y-axis data. If you select multiple y-axis fields, selecting a different color for each field improves pagelet usability. This option is available only if the y-axis data is in stored columns.

**Y-Axis Series** This field is available if the Y Axis Data field is set to in *Rows*. Select the field used to generate a color-coded series of y-axis data. Colors are automatically assigned, but you can override them by supplying custom XSL.

### ***Advanced Options***

**3D Angle** If you select a 3D chart type, enter the angle at which you want the 3D data to be displayed.

**Legend** Select the area of the pagelet where the chart legend appears. Options are:

- *Bottom*
- *Left*
- *None*
- *Right*
- *Top*

**Height** Enter the height of the chart in pixels.

**Width** Enter the width of the chart in pixels.

If you do not enter height or width values, the chart is automatically sized based on the pagelet position and size (narrow or wide).

See Chapter 11, "Using Pagelet Wizard," Overview of Pagelet Wizard Display Sizes, page 286.

**Overlay Fields** Select up to three fields for which you want to display data using a line drawn over the background chart. Overlays apply to all charts types except 2D and 3D pie charts.

### ***Pagelet Preview***

Displays a preview of the pagelet based on most display option settings on this page.

## **Specifying Custom Display Options**

After selecting the *Custom* option on the Select Display Format page, select the Specify Display Options page (PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 4).

## Pagelet Wizard

## Step 5 of 6



&lt; Previous

Next &gt;

## Specify Display Options

Specify the visual options related to the display format for your pagelet.

Custom Pagelet

Custom Options		Pagelet Preview																									
<b>XSL Template</b> <span>Table</span> <span>Generate</span>		<b>PeopleSoft®</b> <b>Set Msg Severity Message</b>																									
<b>XML</b> <pre>&lt;?xml version="1.0" standalone="yes"?&gt; &lt;queryresult&gt;&lt;queryproperties&gt;&lt;querylink &gt;&lt;URL&gt;http://adas0112/psp/pa890dvl/EMP LOYEE/EMPL/n/? </pre>		<table border="1"> <thead> <tr> <th></th> <th></th> <th></th> <th></th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td>M</td> <td>Distributed Object Manager: Help Name=%1 Language=%2</td> </tr> <tr> <td>1</td> <td>2</td> <td>M</td> <td>Distributed Object Manager: Message Set=%2 Language=%1</td> </tr> <tr> <td>1</td> <td>3</td> <td>M</td> <td>Distributed Object Manager: Program=%1</td> </tr> <tr> <td>1</td> <td>4</td> <td>M</td> <td>Distributed Object Manager: Page=%1 Language=%2</td> </tr> <tr> <td>1</td> <td>5</td> <td>M</td> <td>Distributed Object Manager: Record=%1 Language=%2</td> </tr> </tbody> </table>						1	1	M	Distributed Object Manager: Help Name=%1 Language=%2	1	2	M	Distributed Object Manager: Message Set=%2 Language=%1	1	3	M	Distributed Object Manager: Program=%1	1	4	M	Distributed Object Manager: Page=%1 Language=%2	1	5	M	Distributed Object Manager: Record=%1 Language=%2
1	1	M	Distributed Object Manager: Help Name=%1 Language=%2																								
1	2	M	Distributed Object Manager: Message Set=%2 Language=%1																								
1	3	M	Distributed Object Manager: Program=%1																								
1	4	M	Distributed Object Manager: Page=%1 Language=%2																								
1	5	M	Distributed Object Manager: Record=%1 Language=%2																								
<b>XSL</b> <pre>&lt;?xml version="1.0" ?&gt; &lt;!-- Description: XSL T supplied with the </pre>		<a href="#">Click here for more...</a> <a href="#">Full Query Results</a> <p>powered by <b>PeopleSoft</b></p>																									
<b>Additional_text</b> <b>Header</b> <span>PeopleSoft Logo</span> <b>Opening Text</b> <span></span> <input checked="" type="checkbox"/> Show "View Source Data" Link <b>Closing Text</b> <span></span> <b>Footer</b> <span>Powered by PeopleSoft</span>																											

Specify Display Options page (custom display format)

**Custom Options****XSL Template**

Select the XSL template used to generate the custom format pagelet. Available XSL templates are defined on the Define XSL page. Only XSL templates defined for the selected data source are displayed.

See [Chapter 11, "Using Pagelet Wizard," Defining Pagelet Wizard XSL Prototypes, page 300.](#)

**Generate**

Click to generate custom XSL for the pagelet based on the selected XSL template. This XSL is displayed in the XSL text box and is applied to the XML in the XML text box to generate the pagelet preview.

**Pagelet Preview**

Displays a preview of the pagelet based on most display option settings on this page.

## Specifying Data Link Details

Access the Link Details page (Click the Link Options icon).

### Pagelet Wizard

#### Link Details

Specify the details of the Link.

**Link Type:** ☐ Content Reference  
☒ External URL

**Base URL:** Fixed Text Entry

**Link Paths** First 1 of 1 Last

*Path Source	Parameter Value
Fixed Text Entry <input type="text"/>	<input type="text"/>

**Link Parameters** First 1 of 1 Last

*Name Source	Parameter Name	Parameter Value
Fixed Text Entry <input type="text"/>	<input type="text"/>	<input type="text"/>

**Link Suffix:** Fixed Text Entry

**Link Behavior:** ☒ Display in the portal  
☐ Open in a new window

Reset to Default

OK Cancel

Link Details page

You can access this page for field or column names that are displayed on the Specify Display Options page when you access the page for the table, rowset, list, and chart formats. Entering information on the Link Details page enables you to link data that appears on the pagelet to additional information.

Pagelet Wizard builds the column data link as follows using link path, parameter, and suffix values:

`http://baseURL/linkpath1/linkpath2/.../linkpathN/?parm1=val1&parm2=val2&...&parmN=valNlinksuffix`

#### Link Type

Select the type of link destination to define for the data. Options are:

*Content Reference:* Select to define data links to a content reference.

*External URL:* Select to define data links to an external URL.

#### Portal Name

Select the portal in which the content reference resides. This field appears when you select *Content Reference* as the link type.

#### Content Reference

Select the content reference to which the link points. This field appears when you select *Content Reference* as the link type.

**Base URL**

Enter the URL to which you want the link to point. This field appears when you select *External URL* as the link type. Options are:

*Specified by Data:* Select to have the system dynamically generate the link information based on the selected data field.

*Fixed Text Entry:* Select to hardcode the link information by entering a value.

**Link Paths****Path Source**

Select the method used to determine the link destination. Options are:

*Specified by Data:* Select to have the system dynamically generate the link information based on the selected data field.

*Fixed Text Entry:* Select to hardcode the link information by entering a value.

**Link Path**

This field appears if you select *Specified by Data* as the path source. Select the data field used to generate the link.

**Parameter Value**

This field appears if you select *Fixed Text Entry* as the path source. Enter a parameter value.

**Link Parameters****Name Source**

Select the method used to determine the link destination. Options are:

*Specified by Data:* Select to have the system dynamically generate the link information based on the selected data field.

*Fixed Text Entry:* Select to hardcode the link information by entering a value.

**Parameter Name**

Enter a value that corresponds to the value in the Parameter Value field.

**Parameter Value**

If you have selected *Specified by Data* as the name source, select a data field.

If you have selected *Fixed Text Entry* as the name source, enter a parameter value.

**Link Suffix**

Select the method used to determine the link destination.

*Specified by Data:* Select to have the system dynamically generate the link information based on the selected data field.

*Fixed Text Entry:* Select to hardcode the link information by entering a value.

If you have selected *Specified by Data* as the link suffix, select a data field.

If you have selected *Fixed Text Entry* as the link suffix, enter a parameter value.

**Link Behavior**

Options are:

*Display in portal:* Select if you want the content accessed by the data link to be displayed within the portal.

*Open in a new window:* Select if you want the content accessed by the data link to be displayed in a new window.

**Reset to Default**

Click to clear the values on this page and remove the link.

***PeopleTools Portal Technologies XSL Templates***

PeopleTools Portal Technologies delivers seven XSL templates that enable you to transform content from internet news sources that use industry-standard formats such as RSS and ATOM. When selecting the URL data source and the Custom Display Format, the following additional templates appear in the drop-down list box in Step 5 of Pagelet Wizard.

These include:

- ATOM 0.3
- ATOM 1.0
- RSS
- RSS 0.91 - 0.93
- RSS 1.0
- RSS 2.0
- Skeletal XSL

Select the appropriate XSL template from the list and click the Generate button. The formatted news articles will appear in the pagelet preview on the right side.

## Pagelet Wizard

## Step 5 of 6




&lt; Previous

Next &gt;

## Specify Display Options

Specify the visual options related to the display format for your pagelet.

URL Test

Custom Options	Pagelet Preview
<b>XSL Template</b> <span>RSS 0.91-0.93</span> <input type="button" value="Generate"/>	<a href="#">Wired News</a> 
<b>XML</b> <pre>&lt;?xml version="1.0" ?&gt;&lt;!DOCTYPE rss PUBLIC "-//Netscape Communications//DTD RSS 0.91//EN" "http://mx.netscape.com/publish/f</pre>	<p>Wired News, a pioneer in online journalism, has been at the forefront of daily technology news coverage since its launch in 1996. The site's mission is to provide an original, lively and timely chronicle of how technology affects our lives, for better or worse.</p>
<b>XSL</b> <pre>&lt;?xml version="1.0"?&gt; &lt;!-- RSS 0.91, 0.92, 0.93 XSLT --&gt; &lt;xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Tr</pre>	<ul style="list-style-type: none"> <li>• <a href="#">Careful Where You Put That Tree</a> Think you're doing the Earth a favor by planting a tree? Not so fast -- new research shows forest locations could make or break efforts to combat global warming. By Elizabeth Svoboda.</li> <li>• <a href="#">All the World's a Podcast</a> It's incredibly easy to create and syndicate your own audio content. Here's how. By Michael Calore.</li> </ul>
<b>Additional_text</b>	
<b>Header</b> <input type="text"/>	
<b>Opening Text</b> <input type="text"/>	
<b>Closing Text</b> <input type="text"/>	
<b>Footer</b> <input type="text"/>	

Step 5 of Pagelet Wizard with formatted newsfeed

## Step 6: Specifying Pagelet Publication Options

This section discusses common elements used in this section and how to specify pagelet publication options. This is step 6 in the Pagelet Wizard pagelet creation process.

### Common Elements Used in This Section

This section discusses common elements used in this section.

#### Enable Caching

Select to enable caching for the pagelet. Enabling caching improves performance because once the pagelet is stored in the cache memory, the next time the pagelet is accessed, it is retrieved and displayed from cache memory instead of requiring a trip to the server. Selecting this option displays the Caching Options group box.

See Add link to the Caching Options section below

**WSRP Portlet**

Select this option if you intend this pagelet to be produced by PeopleSoft as WSRP portlet.

Web Services for Remote Portlets (WSRP) is a standard that defines the way in which remote web services should plug into the pages of a portal. WSRP standards allow you to easily embed a remote web service into a portal. The portal then displays interactive content and services that are dynamically updated from the remote servers.

When creating WSRP portlets in the Pagelet Wizard, consider the following aspects of WSRP portlet publishing:

- Pagelet rendering.

If the pagelet contains links, the links may transfer to the PeopleSoft portal, or stay in-line within the portlet, or consuming portal. If the links point to the PeopleSoft portal, then access to the PeopleSoft system by the browser is necessary. Therefore, in many cases, remote portals consuming WSRP portlets must employ single signon between the consuming portal and the PeopleSoft portal for them to be fully functional.

- Pagelet personalization.

Edit mode as defined by WSRP standards is equivalent to personalization mode in PeopleSoft portals. If the Pagelet being WSRP produced can be personalized (as in, it has user-specified parameters), then the produced WSRP portlet will support edit mode. When in edit mode, invalid entries on the personalization page invoke error messages by way of WSRP.

- WSRP production.

Selecting the WSRP Portlet option on the Specify Publishing Options page will cause the pagelet to be registered for WSRP production.

See [Chapter 15, "Understanding Web Services For Remote Portlets \(WSRP\)," page 491](#) and [Chapter 17, "Using WSRP to Consume and Produce Remote Portlets," page 517](#).

**Caching Options**

Entering caching options creates a content reference attribute for the new pagelet content reference. The content reference attribute contains the following information:

- Name: *PSCACHECONTROL*
- Label: *PSCache-Control*
- Attribute value: <Caching type field value>;max-age=<Minutes to Cache field value>

Once the pagelet is saved and registered, you can use Pagelet Wizard to alter caching options in the same way you can alter other registration information. However, altering caching option values will require that you re-register the pagelet.

See [Chapter 5, "Administering Portals," Administering Content References, page 109](#).

**Caching Type**

*Private.* Select to enable user-based caching. For example, pagelet caching is stored per user. If user A accesses the pagelet, only user A will access the pagelet from the cache.

*Public.* Select to enable application-based caching. For example, pagelet caching is stored per web server instance. If a user accesses the pagelet using web server instance 1, only other users accessing the pagelets using web server instance 1 will access the pagelet from the cache.

*Role.* Select to enable role-based caching. For example, pagelet caching is stored per role combination. If user A has roles 1, 2, and 3, only another user with all roles 1, 2, and 3 will access the pagelet from the cache.

**Minutes to Cache**

Enter the number of minutes for which you want the pagelet to remain in the cache. If you do not enter a value, a default value of 30 minutes is used.

**Pages Used to Specify Pagelet Publication Options**

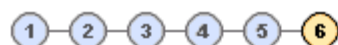
<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Specify Publishing Options	PTPPB_WIZ_PUBOPT	PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard  Click the Next button on the Specify Display Options page, or click the icon for step 6.	Specify the type of pagelet that you want to publish. Provide registration, caching, and security details, and register the pagelet.
Pagelet Creation Confirmed	PTPPB_WIZ_FINISH	Click the Finish button on the Specify Publishing Options page.	Confirm that the creation of the pagelet is complete.

**Specifying Publication Options**

Access the Specify Publishing Options page (PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard; then click the icon for step 6..

## Pagelet Wizard

## Step 6 of 6



&lt; Previous

Finish

## Specify Publishing Options

Specify the manner in which your pagelet is published.

## New Construction Update

☒
**Homepage Pagelet**

Publishing as a Homepage Pagelet allows this pagelet to be placed on a user's Homepage tab. Homepage Pagelets are organized by pagelet folders.

Folder:

**Advanced Options**

☐ Hide Pagelet Title Bar
 ☐ Enable Caching

☐ Hide Pagelet Border

☐ Requires Personalization

**WSRP Options**

Publishing as a WSRP Portlet allows this pagelet to be consumed and displayed by another portal that is WSRP compliant. When using OC4J, WSRP Portlets are also automatically published as JPKD Portlets for consumption and display within the Oracle Portal. OC4J is the "Oracle Containers for Java" web application server. JPKD stands for "Java Portal Development Kit", an exclusive Oracle Portal specification.

☐ WSRP Portlet

☐
**Template Pagelet**

Publishing as a Template Pagelet allows this pagelet to be used with any template. For the Context Manager template, this pagelet can be context sensitive to the target transaction.

**Advanced Options**

**Behavior if Keys not Present**

☒ Use Specified Defaults
 ☐ Display Message
 ☐ Hide Pagelet

☐ Enable Caching

**WSRP Options**

Publishing as a WSRP Portlet allows this pagelet to be consumed and displayed by another portal that is WSRP compliant. When using OC4J, WSRP Portlets are also automatically published as JPKD Portlets for consumption and display within the Oracle Portal. OC4J is the "Oracle Containers for Java" web application server. JPKD stands for "Java Portal Development Kit", an exclusive Oracle Portal specification.

☐ WSRP Portlet

Specify Publishing Options page (1 of 2)

☐ **Embeddable Pagelet**

Publishing as an Embeddable Pagelet allows this pagelet to be rendered on a target transaction page. The target transaction executes this pagelet from the Pagelet Wizard API.

---

**▼ Pagelet Security**

\*Security Type:  ☒ **Author Access**

**Selected Security** [Customize](#) | [Find](#) | [View All](#) | [Print](#) | [Grid](#) | [First](#) | [1 of 1](#) | [Last](#)

*Type	Name	Description		
<input type="text" value="Permission List"/>	<input type="text" value="PTPT1100"/>	Security Administrator	<input type="button" value="+"/>	<input type="button" value="-"/>

### Specify Publishing Options page (2 of 2)

When you access a pagelet definition on this page, the pagelet's registration option (Homepage Pagelet, Template Pagelets, Embeddable Pagelet, or WSRP Portlet) is reflected only as it exists in the portal you are currently logged in to. Select the registration option to publish the pagelet to the portal where you are currently logged in.

For example, if you access ABC123 pagelet, the Homepage Pagelet option is selected only if the portal where you are currently logged in contains a CREF for the ABC123 pagelet. If there is no CREF for this pagelet in the current portal, even if there is a CREF for this pagelet in another portal on the same database, the Homepage Pagelet option will be clear. Select the Homepage Pagelet option to publish the pagelet to the current portal.

### Homepage Pagelet

Select this option to publish the pagelet as a homepage pagelet that can be added to a user's homepage.

**Folder** Select the pagelet folder to which you want to assign the pagelet. The pagelet appears in this pagelet category on the Personalize Content page, which users access to add or remove pagelets from their homepages.

*See Enterprise PeopleTools 8.50 PeopleBook: Using PeopleSoft Applications.*

### Advanced Options

**Hide Pagelet Title Bar** Select to display the pagelet without a title bar.

**Hide Pagelet Border** Select to display the pagelet without a border.

**Requires Personalization** Select to specify that the pagelet requires end-user personalization. The pagelet displays a personalization button that accesses a personalization page. With this option selected, you are requiring that the end user access the personalization page to personalize and specify data source parameter values before it displays data. A message notifying the user of the personalization requirement appears on the pagelet. If this option is clear, the pagelet displays data using the default data source parameter values provided.

## ***Template Pagelet***

Select this option to publish the pagelet as a template pagelet that can be added to any template. Template pagelets can extract and leverage keys specified by the target transaction for the template. Publishing a pagelet as a template pagelet is useful when you use the PeopleTools Context Manager to display the template pagelet.

When a pagelet is published as a template pagelet, any user-specified parameters are automatically available as parameters that can be set by key passing in Context manager. For the keys to pass correctly, the key name in Context Manager (the key name on the page) must be identical to the name of the parameter in the pagelet. If the names match, then any key values passed are used by the template pagelet in place of the default values. If the names are not an exact match, the default values appear. Since users cannot customize template pages, user-specified parameters can only show the default values or the values that come through Context Manager key passing.

## ***Advanced Options***

### **Behavior if Keys not Present**

Select one of the following options to indicate how pagelet data is displayed if target transaction keys are not present to populate the pagelet's parameters. Options are:

*Use Specified Defaults:* If target transaction keys are not present, the pagelet will display data using the defaults specified in Step 3: Specify Pagelet Data Source Parameters.

*Display Message:* If target transaction keys are not present, the pagelet does not display data but instead displays a message.

*Hide Pagelet:* If target transaction keys are not present, the pagelet does not appear.

## ***WSRP Portlet***

Select this option to make your pagelet available for use with external portals that support the WSRP standard.

See [Chapter 11, "Using Pagelet Wizard," Specifying Publication Options, page 367.](#)

## ***Embeddable Pagelet***

Select this option to make your pagelet available as an embeddable pagelet that can be generated by the Pagelet Wizard API for rendering within an HTML area of a PeopleSoft Pure Internet Architecture target page or using an iScript.

## ***Pagelet Security***

If you change any pagelet security options for a registered pagelet, you must re-register the pagelet.

<b>Security Type</b>	<p>Select the type of security that you want to assign to the pagelet. Options are:</p> <p><i>Public Access:</i> Select to designate that all users can access the pagelet.</p> <p><i>Select Security Access:</i> Select to assign role- or permission list-based security to the pagelet. The Selected Security scroll area appears.</p>
<b>Author Access</b>	<p>Select to provide the pagelet author with access to the pagelet regardless of any security restrictions assigned to the pagelet. This access is granted based on the author's user ID.</p>
<b>Update Security</b>	<p>This button displays when Pagelet Wizard determines that there is an inconsistency between the security defined for the pagelet on this page and the security on one or more of the content references defined for the pagelet. When you click this button, the content reference security is updated for all of the content references for Homepage Pagelet, Template Pagelet, and WSRP Portlet options.</p>
<b><i>Selected Security</i></b>	
<b>Type</b>	<p>Select the type of security assigned to the pagelet. Specified security is applied to the content reference entries when the pagelets are registered. Options are:</p> <p><i>Permission List:</i> Select to assign permission list-based security to the pagelet. Select a permission list in the Name field.</p> <p><i>Role:</i> Select to assign role-based security to the pagelet. Select a role in the Name field.</p>
<b>Name</b>	<p>Prompt to select or enter the name of the permission list or role to which to authorize access to the pagelet.</p>
<b><i>Buttons</i></b>	
<b>Previous</b>	<p>Click to move backwards one step in pagelet creation process in the Pagelet Wizard.</p>
<b>Finish</b>	<p>Click to complete the creation of the pagelet. The Pagelet Creation Confirmed page appears.</p> <p>After you click the Finish button for a pagelet, the button no longer appears when you access the pagelet definition in Pagelet Wizard. Once a pagelet has been created, you can apply any modifications to the pagelet definition by clicking the Save button.</p>

---

## Using Pagelet Wizard Post-Transformation Processing

This section provides an overview of Pagelet Wizard post-transformation processing and discusses how to:

- Insert message catalog text.
- Format Date, Time, DateTime, Currency, and Number elements.
- Insert a content-reference-based link.
- Insert a content reference URL as text.
- Generate charts.
- Insert pagelet headers and footers in specific locations.

## Understanding Pagelet Wizard Post-Transformation Processing

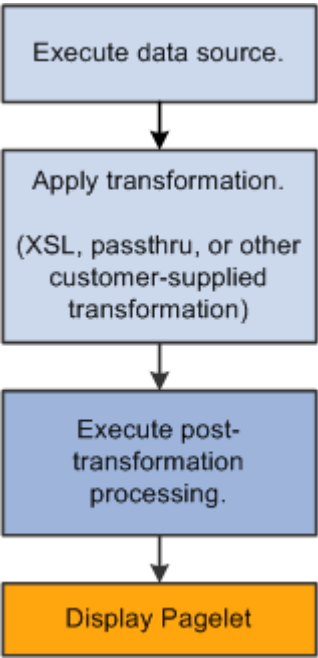
Pagelet Wizard enables you to use delivered Pagelet Wizard builder transformation tasks in custom and passthru transformations.

Listed in the order in which they are executed in post-transformation processing, the following transformation tasks are not automatically generated for custom and passthru transformations, but they can be added using embedded tags:

1. Insert Message Catalog text.
2. Format Date, Time, DateTime, Currency, and Number.
3. Insert content-reference-based links.
4. Insert content reference URLs as text.
5. Generate charts.
6. Insert pagelet headers and footers in specific locations.

These tasks are handled by the Post Transformation step, which is built into Pagelet Wizard.

The following diagram illustrates the point at which the Post Transformation step runs in the process flow for displaying a Pagelet Wizard pagelet.



Pagelet Wizard pagelet-display processing flow

The Post Transformation step runs after the completion of the Transformation step. Following the Post Transformation step, the Pagelet Wizard pagelet appears for the user. Pagelet Wizard uses embedded tags to identify the type of Post Transformation processing that occurs.

Tags that perform these post-transformation tasks can be placed in the HTML for HTML pagelets using the passthru transformation, or in the XSL for pagelets using a custom transformation. For all other transformations—such as chart, list, table, and so forth—Pagelet Wizard automatically generates the tags when necessary. These tags are primarily meant to be used by pagelet designers who want to create custom pagelets using functionality similar to what our delivered Pagelet Wizard builders provide.

Inserting Message Catalog Text

Use the following syntax to insert the text of a message catalog entry. Variables that are user-supplied are italicized.

**Note.** This processing step occurs only if the output of the Transformation step is XHTML-compliant.

See <http://www.w3.org/TR/xhtml1/>.

Item to Insert	XHTML Tag	HTML After Processing
Message Catalog Text	<PSMSGTEXT MSGSET="message set" MSGNBR="message number" DEFAULT="defaulttext"/>	Message text.
Message Catalog Explain Text	<PSMSGEXPLAINTEXT MSGSET="message set" MSGNBR="message number" DEFAULT="defaulttext"/>	Message explanation text.

## Formatting Date, Time, DateTime, Currency, and Number Elements

Use the following syntax to format a Date, Time, DateTime, Currency, or Number elements. Variables that are user-supplied are set in italics.

**Note.** This processing step only occurs if the output of the Transformation step is XHTML compliant.

See <http://www.w3.org/TR/xhtml1/>.

<i>Item to Insert</i>	<i>XHTML Tag</i>	<i>HTML After Processing</i>
Date	<pre>&lt;PSFORMAT TYPE="DATE" [FORMAT="format string "] [FORMAT_3 character language code="format string "]&gt;DateToFormat&lt;/PSFORMAT&gt;</pre> <p><b>Note.</b> FORMAT has the same syntax as the DateTimeToLocalizedString() PeopleCode Function.</p>	<p>Formatted Date.</p> <p>Language-specific Formatted Date.</p>
Time	<pre>&lt;PSFORMAT TYPE="TIME" [SOURCETZ="Source Time Zone "] [DESTTZ="Destination Time Zone "] [FORMAT="format string "]&gt;TimeToFormat&lt;/PSFORMAT&gt;</pre> <p><b>Note.</b> SOURCETZ and DESTTZ have the same format as the DateTimeToTimeZone() PeopleCode function. FORMAT has the same syntax as the DateTimeToLocalizedString() PeopleCode Function.</p>	Formatted Time.
DateTime	<pre>&lt;PSFORMAT TYPE="DATETIME" [SOURCETZ="Source Time Zone "] [DESTTZ="Destination Time Zone "] [FORMAT="format string "]&gt;DateTimeToFormat&lt;/PSFORMA T&gt;</pre> <p><b>Note.</b> SOURCETZ and DESTTZ have the same format as the DateTimeToTimeZone() PeopleCode function. FORMAT has the same syntax as the DateTimeToLocalizedString() PeopleCode Function.</p>	Formatted DateTime.

<i>Item to Insert</i>	<i>XHTML Tag</i>	<i>HTML After Processing</i>
Number	<pre>&lt;PSFORMAT TYPE="NUMBER" FORMAT="format string" [WIDTH="width"] [PRECISION="precision"]&gt;NumberToFormat&lt;/PSFORMAT&gt;</pre> <p><b>Note.</b> FORMAT, WIDTH, and PRECISION have the same format as the NumberToDisplayString() PeopleCode Function.</p>	Formatted Number.
Currency	<pre>&lt;PSFORMAT TYPE="CURRENCY" FORMAT="format string" [WIDTH="width"] [PRECISION="precision"] [CURRENCY_CODE="CODE "] [CURRENCY_DECORATION="SYMBOL/CODE/BOTH"]&gt;NumberToFormat&lt;/PSFORMAT&gt;</pre> <p><b>Note.</b> FORMAT, WIDTH, and PRECISION have the same format as the NumberToDisplayString() PeopleCode Function. CURRENCY_CODE can be one of the supported PeopleSoft currency codes, or if left blank, defaults to the user's currency code. CURRENCY_DECORATION can have the value of <i>SYMBOL</i> (currency symbol appears before the number), <i>CODE</i> (currency code appears after the number), or <i>BOTH</i> (the default).</p>	Formatted Number with currency symbol and/or currency code.

## Disabling WSRP Style Substitution

Use the following post-processing tag to disable WSRP style substitution. <PSNOWSRPSTYLELESSUBST/> takes no parameters and signals that the style classes referenced in the pagelet are already WSRP compliant. Including this tag can have a slight positive improvement in performance.

## Inserting a Content-Reference-Based Link

Use the following syntax to insert a content-reference-based link into a Pagelet Wizard pagelet. Variables that are user-supplied are set in italics.

---

**Note.** This processing step only occurs if the output of the Transformation step is XHTML compliant.

---

See <http://www.w3.org/TR/xhtml1/>.

<i>Item to Insert</i>	<i>XHTML Tag</i>	<i>HTML After Processing</i>
Content Reference Link	<PSREGISTERDLINK PORTAL=" <i>portal name</i> " CREF=" <i>content reference name</i> " target=" <i>target</i> " APPEND=" <i>append</i> ">Text of Link</PSREGISTEREDLINK>	<A HREF="http...URL of content reference...[?/&]append" target="target">Text of Link</A>

## Inserting a Content Reference URL as Text

Use the following syntax to insert a content reference URL as text into a Pagelet Wizard pagelet. Variables that are user-supplied are set in italics.

---

**Note.** This processing step only occurs if the output of the Transformation step is XHTML compliant.

---

See <http://www.w3.org/TR/xhtml1/>.

<i>Item to Insert</i>	<i>XHTML Tag</i>	<i>HTML After Processing</i>
Content Reference URL	<PSREGISTERDURL PORTAL=" <i>portal name</i> " CREF=" <i>content reference name</i> " APPEND=" <i>append</i> " />	http...URL of content reference... [?/&]append

## Generating Charts

Embedded charts are generated in the Post Transformation step.

For details about chart generation and its syntax, refer to the ChartXSLBuilder included in the EOPPB\_PAGELET Application Class. You can access the ChartXSLBuilder application class code using PeopleSoft Application Designer.

## Inserting Pagelet Headers and Footers in Specific Locations

Pagelet Wizard automatically inserts a pagelet's header in front of the output of the Transformation step, and it inserts the pagelet's footer after the output of the Transformation step.

To alter this behavior, you must use special HTML comment tags. Here is the syntax of these tags:

<i>Item to Insert</i>	<i>HTML Tag</i>	<i>XSL Equivalent</i>
Header	<!--PWHEADER-->	<xsl:comment>PWHEADER</xsl:comment>
Footer	<!--PWFOOTER-->	<xsl:comment>PWFOOTER</xsl:comment>

---

## Administering Pagelets

This section discusses the following:

- Deleting and copying Pagelet Wizard pagelets.
- Generating export and import scripts for Pagelet Wizard.
- Publishing pagelets to multiple portals.
- Publishing multiple pagelets to a portal
- Reviewing Pagelet Wizard pagelet details

## Common Elements Used in This Chapter

This section discusses common elements used in this chapter.

### ***Filter Pagelets***

If appropriate, use the options in the Filter Pagelets group box to narrow the range of pagelets displayed in the Pagelets scroll area.

#### **Data Type**

Select the data type of the pagelet that you want to appear in the Pagelets scroll area. Options are:

- *All Data Types.*
- *Collaborative Workspace Links.*
- *Content Management.*
- *HTML.*
- *Java.*
- *Navigation Collection.*
- *News Publications.*
- *PS Query.*
- *Rowset.*
- *Search Record.*
- *Shortcut Collection.*
- *URL.*

<b>Published Type</b>	<p>Select the pagelet type assigned to the pagelet that you want to appear in the Pagelets scroll area. Options are:</p> <ul style="list-style-type: none"> <li>• <i>All Pagelets.</i></li> <li>• <i>Embeddable Pagelet.</i></li> <li>• <i>Homepage Pagelet.</i></li> <li>• <i>Non-published Pagelet.</i></li> <li>• <i>Template Pagelet.</i></li> <li>• <i>WSRP.</i></li> </ul>
<b>Portal Name</b>	The portal name that you select here does not filter results, but rather serves to inform you which pagelets displayed in the Pagelet scroll area have been published in the selected portal. Pagelets are automatically published to the portal where the pagelet author is logged in when creating the pagelet.
<b>Pagelet Keyword</b>	Enter keywords to narrow the range of pagelets that appear in the Pagelets scroll area. Pagelet elements that are used as keywords used include pagelet ID, title, and description.
<b>Search</b>	Click to perform a pagelet search using the criteria entered in the Filter Pagelets group box.
<b>Reset</b>	Click to clear the Pagelet Keyword field.

### ***Pagelets***

The Pagelets scroll area provides the following pagelet information.

<b>Pagelet ID</b>	Displays the pagelet ID. Click to access the pagelet definition in the Pagelet Wizard component.
<b>Pagelet Title</b>	Displays the pagelet title.
<b>Homepage</b>	If selected, indicates that the pagelet has been published as a homepage page.
<b>Template</b>	If selected, indicates that the pagelet has been published as a template pagelet.
<b>WSRP</b>	If selected, indicates that the pagelet has been published as a WSRP portlet.
<b>Embeddable</b>	If selected, indicates that the pagelet has been published as an embeddable pagelet.

## Pages Used to Administer Pagelet Wizard Pagelets

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Delete Pagelets	PTPPB_ADMIN_DELETE	PeopleTools, Portal, Pagelet Wizard, Delete Pagelets	Delete one or more pagelets.  Deleting a pagelet removes the pagelet definition from Pagelet Wizard, as well as deletes associated metadata and registry entries from the database and portal registry.
Copy Pagelets	PTPPB_ADMIN_SAVEAS	PeopleTools, Portal, Pagelet Wizard, Copy Pagelets	Create a new pagelet that is a clone of an existing pagelet. The new pagelet has a unique pagelet ID.  The pagelet copy inherits all aspects of the original pagelet, with the exception of registration information.
Export/Import Pagelets	PTPPB_ADMIN_EXPIMP	PeopleTools, Portal, Pagelet Wizard, Export/Import Pagelets	Exporting and importing a pagelet generates the Data Mover script text to export and import the Pagelet Wizard defined pagelets.
Review Pagelets	PTPPB_ADMIN_REPORT	PeopleTools, Portal, Pagelet Wizard, Review Pagelets	Review the existing Pagelet Wizard defined pagelets.

## Understanding Administering Pagelets

This chapter lists common elements used in administering pagelets and discusses how to:

- Delete and copy Pagelet Wizard pagelets.
- Generate export and import scripts for pagelets.
- Review Pagelet Wizard pagelet details.

## Deleting and Copying Pagelet Wizard Pagelets

This section discusses how to:

- Delete pagelets.
- Copy pagelets.

## Deleting Pagelets

Access the Delete Pagelets page (PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard; Delete Pagelets).

### Delete Pagelets

Deleting a pagelet removes the definition from Pagelet Wizard as well unpublishing the pagelet.

**Additional Instructions**

**Filter Pagelets**

\*Data Type: All Data Types

\*Published Type: All Pagelets

\*Portal Name: CUSTOMER Customer-facing registry content

Pagelet Keyword:

**Pagelets**
Customize | Find | View All | First 1-10 of 29 Last

General Published

Select	Pagelet ID	Pagelet Title
<input type="checkbox"/>	<a href="#">DEMO FAC BUILDING</a>	New Construction Update
<input type="checkbox"/>	<a href="#">EOPP CREF BY DATE ID</a>	Content References
<input type="checkbox"/>	<a href="#">EOPP POPULAR PGLT LNG SCR</a>	Popular Pagelets by Language
<input type="checkbox"/>	<a href="#">EOPP POPULAR PGLT SRC</a>	Popular Pagelets
<input type="checkbox"/>	<a href="#">NEW EMPLOYEE UPDATE</a>	For New Employees
<input type="checkbox"/>	<a href="#">PAPI CFO INSIGHTS TOOLS</a>	Investor Insights & Tools
<input type="checkbox"/>	<a href="#">PAPP BRANDING HOMEPAGE</a>	Branding Center
<input type="checkbox"/>	<a href="#">PAPP COMPANY NEWS</a>	Company News
<input type="checkbox"/>	<a href="#">PAPP CONTENT MANAGER HOMEPAGE</a>	Content Manager Center
<input type="checkbox"/>	<a href="#">PAPP CONTEXT MANAGER HOMEPAGE</a>	Context Manager Center

☒ Select All
☐ Clear All

Delete Selected Pagelets

Delete Pagelets page

Deleting a pagelet removes the pagelet definition from Pagelet Wizard. In addition, associated metadata and registry entries are deleted from the database and portal registry.

**Select** Select the pagelets to delete.

**Delete Selected Pagelets** Click to delete selected pagelets. The system prompts you to confirm the deletion.

The Published tab shows information about how the pagelet has been published.

<b>Homepage</b>	If selected, indicates that the pagelet has been published as a homepage page.
<b>Template</b>	If selected, indicates that the pagelet has been published as a template pagelet.
<b>WSRP</b>	If selected, indicates that the pagelet has been published as a WSRP portlet.
<b>Embeddable</b>	If selected, indicates that the pagelet has been published as an embeddable pagelet.

### Copying Pagelets

Access the Copy Pagelets page (PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard; Copy Pagelets).

## Copy Pagelets

Copying a pagelet clones an existing definition from Pagelet Wizard.

► **Additional Instructions**

▼ **Filter Pagelets**

\***Data Type:**

\***Published Type:**

\***Portal Name:**  Customer-facing registry content

**Pagelet Keyword:**

**Pagelets** Customize | Find | View All |  First 1-10 of 29 Last

**General** **Published**

<u>Pagelet ID</u>	<u>Pagelet Title</u>
<input type="radio"/> <a href="#">DEMO FAC BUILDING</a>	New Construction Update
<input type="radio"/> <a href="#">EOPP CREF BY DATE ID</a>	Content References
<input type="radio"/> <a href="#">EOPP POPULAR PGLT LNG SCR</a>	Popular Pagelets by Language
<input type="radio"/> <a href="#">EOPP POPULAR PGLT SRC</a>	Popular Pagelets
<input type="radio"/> <a href="#">NEW EMPLOYEE UPDATE</a>	For New Employees
<input type="radio"/> <a href="#">PAPI CFO INSIGHTS TOOLS</a>	Investor Insights & Tools
<input type="radio"/> <a href="#">PAPP BRANDING HOMEPAGE</a>	Branding Center
<input type="radio"/> <a href="#">PAPP COMPANY NEWS</a>	Company News
<input type="radio"/> <a href="#">PAPP CONTENT MANAGER HOMEPAGE</a>	Content Manager Center
<input type="radio"/> <a href="#">PAPP CONTEXT MANAGER HOMEPAGE</a>	Context Manager Center

\***New Pagelet ID**

☐ **Include Personalization Data**

Copy Pagelets page

The pagelet copy inherits all aspects of the original pagelet, with the exception of registration information.

<b>Select</b>	Select the pagelet to copy.
<b>New Pagelet ID</b>	Enter a unique pagelet ID for the new pagelet that will be copied from the selected existing pagelet. <hr/> <b>Note.</b> Pagelet IDs can only contain the characters A-Z, a-z, and _. All other characters put into a pagelet ID will be removed by the system. If all characters are removed from the user entered pagelet ID, then the Registry Object Prefix will be used instead. If, after removing illegal characters from the pagelet ID, the name is found to match an existing pagelet ID, then Pagelet Wizard will append numbers to the end of the pagelet ID to make the new pagelet ID unique. <hr/>
<b>Include Personalization Data</b>	Select to copy any end-user personalizations defined for the existing pagelet along with the new pagelet.
<b>Save New Pagelet</b>	Click to create the cloned pagelet with the new pagelet ID. Once you create the copied pagelet, you must publish it before it can be used as a homepage pagelet or a template pagelet. You can access the new pagelet in the Pagelet Wizard and publish the pagelet during step 6 on the Specify Publishing Options page.

The Published tab shows information about how the pagelet has been published.

<b>Homepage</b>	If selected, indicates that the pagelet has been published as a homepage page.
<b>Template</b>	If selected, indicates that the pagelet has been published as a template pagelet.
<b>WSRP</b>	If selected, indicates that the pagelet has been published as a WSRP portlet.
<b>Embeddable</b>	If selected, indicates that the pagelet has been published as an embeddable pagelet.

## Generating Export and Import Scripts for Pagelet Wizard Pagelets

This section provides overviews of export and import scripts for Pagelet Wizard pagelets and export and import scripts for Pagelet Wizard setup data, and discusses how to generate export and import pagelet scripts.

### *Understanding Export and Import Scripts for Pagelet Wizard Pagelets*

You can use these export and import scripts to copy Pagelet Wizard pagelet definitions from one database to another.

When you export or import a pagelet, the imported pagelet is not automatically published. You must access the pagelet definition in Pagelet Wizard in the target portal in the target database and publish the pagelet on the Specify Publishing Options page in step 6.

Alternatively, you can run the generated export and import Data Mover scripts (.DMS scripts) and copy the pagelet CREF by placing it into a PeopleSoft Application Designer project.

**Note.** These export and import scripts do not include the ability to export or import Pagelet Wizard setup data associated with the pagelet definitions being exported/imported.

## Generating Export and Import Pagelet Scripts

Access the Export/Import Pagelets page (PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard; Export/Import Pagelets).

## Export/Import Pagelets

Exporting and importing a pagelet generates the Data Mover script text to export and import the Pagelet Wizard defined pagelets.

Additional Instructions

Filter Pagelets

\*Data Type: All Data Types

\*Published Type: All Pagelets

\*Portal Name: EMPLOYEE Employee-facing registry content

Pagelet Keyword: Search Reset

Pagelets Customize | Find | View All First 1-10 of 29 Last

General Published

Select	Pagelet ID	Pagelet Title
<input type="checkbox"/>	DEMO FAC BUILDING	New Construction Update
<input type="checkbox"/>	EOPP_CREF_BY_DATE_ID	Content References
<input type="checkbox"/>	EOPP_POPULAR_PGLT_LNG_SCR	Popular Pagelets by Language
<input type="checkbox"/>	EOPP_POPULAR_PGLT_SRC	Popular Pagelets
<input type="checkbox"/>	NEW_EMPLOYEE_UPDATE	For New Employees
<input type="checkbox"/>	PAPI_CFO_INSIGHTS_TOOLS	Investor Insights & Tools
<input type="checkbox"/>	PAPP_BRANDING_HOMEPAGE	Branding Center
<input type="checkbox"/>	PAPP_COMPANY_NEWS	Company News
<input type="checkbox"/>	PAPP_CONTENT_MANAGER_HOMEPAGE	Content Manager Center
<input type="checkbox"/>	PAPP_CONTEXT_MANAGER_HOMEPAGE	Context Manager Center

☒ Select All
 ☐ Clear All
 ☐ Include Personalization Data

Generate Scripts

Export/Import Pagelets page

**Select** Select the pagelets for which you want to generate export and import scripts.

<b>Include Personalize Data</b>	Select to include any end-user personalizations defined for the selected pagelets in the export and import.
<b>Generate Scripts</b>	<p>Click to generate export and import scripts for the selected pagelets. The scripts appear on the Data Mover Scripts page.</p> <p>Copy the generated scripts into PeopleSoft Data Mover to export the selected pagelet definitions from a source database and import them into a target database.</p> <p>See <i>Enterprise PeopleTools 8.50 PeopleBook: Data Management</i>, "Using PeopleSoft Data Mover"</p>

---

**Note.** There are other Pagelet Wizard-related Data Mover scripts delivered with PeopleSoft Enterprise Components for users who need more customizable scripting for pagelet migration between databases. You can find these scripts delivered in the pshome scripts directory.

---

### ***Generating Export/Import Scripts for Pagelet Wizard Setup Data***

We deliver the following Data Mover scripts to export/import Pagelet Wizard setup data:

- EOPPB\_SETUP\_DEFN\_IMPORT.DMS
- EOPPB\_SETUP\_DEFN\_EXPORT.DMS

Use these scripts to copy the following types of Pagelet Wizard setup data from one database to another; from a development database to a production database, for example:

- Footers.
- Headers.
- Categories.
- XSL prototype templates.

You can find these scripts delivered in the PS\_HOME scripts directory.

## **Reviewing Pagelet Wizard Pagelet Details**

Access the Review Pagelets page (PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard; Review Pagelets).

## Review Pagelets

Review the existing Pagelet Wizard defined pagelets.

**Additional Instructions**

Use the criteria to narrow the search for a pagelet. The Portal Name criteria informs you if the pagelet is published in the selected portal. It does not filter the results. The Pagelet Keyword criteria filters the results for the Pagelet ID, Title or Description that contains the entered value.

**Filter Pagelets**

**\*Data Type:**

**\*Published Type:**

**\*Portal Name:**  Employee-facing registry content

**Pagelet Keyword:**

**Pagelets**
Customize | Find | View All | 
First 1-25 of 29 Last

General | Type | Published | **Audit**

Pagelet ID	Created	Created By	Updated	Updated By
<a href="#">DEMO FAC BUILDING</a>		VP1	01/13/2006 11:23:57AM	VP1
<a href="#">EOPP CREF BY DATE ID</a>	02/09/2004 2:30:02.000000PM	PS	02/11/2004 3:02:21PM	PS
<a href="#">EOPP POPULAR PGLT LNG SCR</a>		VP1	03/14/2004 1:15:29AM	VP1

Review Pagelets page

On any of the tabs for this component, you can click on the Pagelet ID to open Pagelet Wizard and the definition of the pagelet.

### General Pagelet Information

Click the Pagelet ID link to open Pagelet Wizard and the definition of the pagelet.

### Pagelet Type Information

<b>Category</b>	Displays the category ID from step 1 of Pagelet Wizard.
<b>Data Type</b>	Displays the pagelet data type.
<b>Display Format ID</b>	Displays the format ID of the pagelet.

### Pagelet Publishing Information

<b>Homepage</b>	If selected, indicates that the pagelet has been published as a homepage page.
<b>Template</b>	If selected, indicates that the pagelet has been published as a template pagelet.

<b>WSRP</b>	If selected, indicates that the pagelet has been published as a WSRP portlet.
<b>Embeddable</b>	If selected, indicates that the pagelet has been published as an embeddable pagelet.
<b>Cache</b>	The system displays a check box if caching is enabled for this pagelet.
<b>Time</b>	The time the pagelet is held in the cache.

### ***Pagelet Audit Information***

This tab includes the timestamp and user ID of the person who created the pagelet as well as information on any updates to the pagelet.

## Chapter 12

# Building Pagelets

This chapter provides overviews of pagelets, PeopleSoft portal architecture, pagelet development, and single signon and pagelets. This chapter also discusses how to:

- Develop pagelets using PeopleSoft Pure Internet Architecture.
- Develop pagelets using iScripts.
- Develop pagelets using Java.
- Develop contextual embeddable pagelets.
- Administer pagelets in the portal registry.

---

## Prerequisites

This document provides a practical guide for technical users and programmers who develop pagelets (or portlets) for operation within a PeopleSoft portal. To take full advantage of the information covered in this document, we recommend that you have a basic understanding of the PeopleSoft Pure Internet Architecture and portal technology. Extensive information can be found in the PeopleBooks documentation. Many of the fundamental concepts related to the portal are discussed in the following PeopleSoft documents:

- *Enterprise PeopleTools 8.50 Hardware and Software Requirements*
- *Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*

---

## Understanding Pagelets

This section provides overviews of:

- Pagelets.
- Pagelet extensions.
- Pagelet personalizations.

## Describing Pagelets

Pagelets are small pages that provide display-only snapshots of useful and relevant content from PeopleSoft or non-PeopleSoft applications, or websites. In other portal products, pagelets are sometimes referred to as portlets, gadgets, or modules.

Homepage pagelets are designed to appear on the user's portal homepage. They can be added to the homepage by way of the Personalize Content link on the homepage. Homepage pagelets typically present general data and usually provide links to access more detailed information or a target transaction.

Template pagelets are small, narrow-format pagelets that are designed to appear in the left frame, much like the menu. They are usually only added to the area by an administrator creating a template that includes it. The administrator could also include the pagelet by configuring it to appear in the area by way of the PeopleTools Context Manager. Template pagelets are usually designed to display data that is related in some way to the target transaction.

Pagelets follow a basic set of rules so that they can be displayed properly on a PeopleSoft portal homepage or in the left frame. Homepage pagelet size corresponds to the homepage layout specified by the user. A user can specify either a two-column layout of one narrow pagelet and one wide pagelet, or a three-column layout of three narrow pagelets. Template pagelet size should use the narrow column format of a pagelet. Any pagelet that you design must conform to the dimensions of the narrow column and optionally, of the wide column. Column width is the only restriction on pagelet dimensions, although design principles suggest keeping pagelets as short as possible.

Which homepage pagelets display on the portal home depends on a few factors, including the type of portal you have implemented, the permissions granted to the user, and how the user has customized her homepage.

PeopleSoft pagelets are simply URL-accessible HTML documents that are designed to be size-appropriate for insertion onto a homepage. Pagelets can also originate from remote, non-PeopleSoft systems such as static HTML files or dynamically generated HTML from a variety of technologies.

## Describing Pagelet Development

Pagelets can be developed in PeopleTools as smaller-than-normal pages, just as you would any other PeopleSoft Pure Internet Architecture page. PeopleSoft portals are delivered with a suite of pagelets, primarily built using PeopleTools technology. However, you can also base the design of a pagelet on PeopleCode internet scripts (iScripts). Use iScripts only when you can't accomplish the same task with a PIA page; for example, if you are developing a pagelet that uses external content (content that does not originate in a PeopleSoft application). You can also create pagelets using other leading web-enabling technology such as JavaServer pages (JSP), Java servlets, Microsoft Active Server Pages (ASP), Perl scripts, and common gateway interface (CGI) programs.

A compelling pagelet serves the following primary functions:

- Summarize data that is relevant, personalized, and actionable. Screen real estate is valuable; pagelets must do more than provide "eye candy."
- Provide links to more detailed information, such as content in another application.

In addition, pagelets should also:

- Aggregate data from a variety of sources.

- Be simple and intuitive.
- Be sized appropriately.

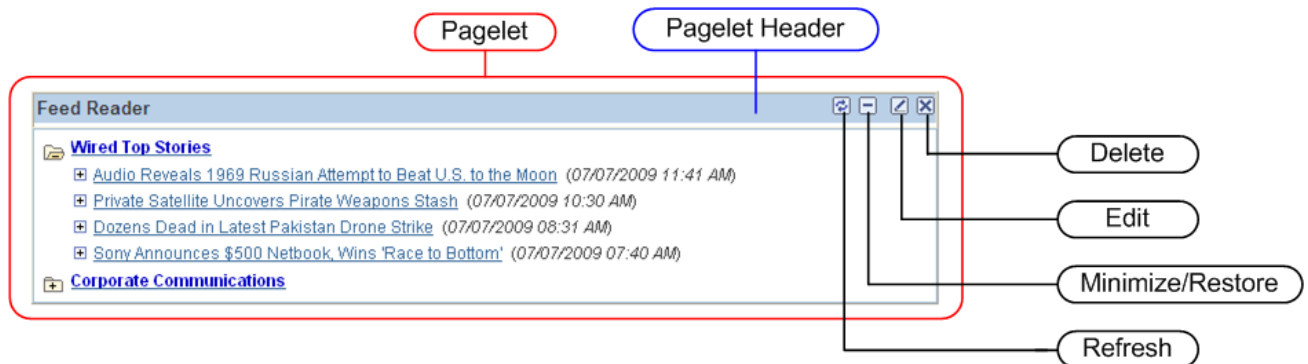
The size of pagelets corresponds to the homepage layout specified by users. Any pagelet you design must first conform to the dimensions of the narrow column and optionally, of the wide column. The only specific dimensions that must be adhered to are the width of the pagelet. The length of the pagelet can vary, but good design principles suggest keeping them as short as possible.

- *Not* use scrollable grids.

Scrollable grids displayed in a pagelet default to the original grid. If the grid appears in a pagelet and in the PIA page, Oracle recommends that you use original grid in a pagelets to insure consistent user experiences

Let's take a look at an example. The Tasks pagelet is delivered with the PeopleSoft Enterprise Portal. It aggregates worklist items from various PeopleSoft applications and allows the user to add personal tasks items. It summarizes relevant, personalized data by:

- Displaying only open (not completed) tasks for the current user.
- Displaying the number of tasks the user requests.
- Marking overdue items with an icon.
- Allowing the user to add more tasks or see other tasks not displayed.



Feed Reader pagelet




The buttons on the pagelet header provide functions that are common to most pagelets. The Remove button deletes the pagelet from your homepage. The Minimize button temporarily hides the pagelet content by displaying just the pagelet header, not the body of the pagelet. When minimized, this button is replaced by the Restore button. Clicking it displays the full pagelet, as shown above. The Edit (Customize) button is used to personalize the pagelet. The Help button opens another browser window to display either PeopleBooks or your own help pages.

## Describing Pagelet Extensions

Some pagelets also utilize pagelet extensions, which are the supporting or supplementary components for a pagelet. They are not displayed on your homepage, but are accessible by way of links and buttons on pagelets. Pagelet extensions are optional. They are a means of providing additional information or editing capability that is too cumbersome to display on the pagelet.

Clicking the name of a task on the Tasks pagelet navigates to the worklist item in another database, or to the personal task defined in the portal. In the latter case, the page that is used to display personal task information is known as an enhanced page. It extends the pagelet by showing more detail. This page also provides the user with the option to perform further actions, such as deleting the task.

## Task Details

<b>*Task:</b>	<input type="text" value="Finish Quarterly Report"/>
<b>Due Date:</b>	<input type="text" value="08/09/2002"/> 
<b>Status:</b>	<input type="text" value="In Process"/> 
<b>Priority:</b>	<input type="text" value="High"/> 
<b>Task Description:</b>	
<div><div>Finish Q3 report and turn it in for review</div><div></div></div>	
<b>Attachment</b>	
<b>File Description:</b>	<div><div></div><div></div></div>
<div>Add File</div>	
<div>SaveDelete</div>	
<a href="#">Return to Tasks</a>	

Task Details page

Another enhanced page is accessed by clicking the More link on the Tasks pagelet. The enhanced page displays all tasks for the user. Note that the More link is displayed on the pagelet, since it can (and should) only display the most relevant data. If there were dozens of open tasks, which is likely over time, the pagelet would grow to be too long and possibly crowd out relevant, actionable data on other pagelets. The enhanced page is a more appropriate vehicle for displaying details as it is wider than the pagelet and can therefore display more columns of information.




## Tasks

6 Task(s)

View:

All

Refresh

Task	Due Date	From	Status	Priority
 <a href="#">Meeting at Hills</a>	08/05/2002	Vice President of Finance	On Hold	Medium
 <a href="#">Call James Dorf</a>	08/06/2002	Vice President of Finance	New	Low
 <a href="#">Finish Quarterly Report</a>	08/09/2002	Vice President of Finance	In Process	High
<a href="#">Cancel newsletter subscription</a>	08/23/2002	Vice President of Finance	New	Low
<a href="#">Clear Q2 worklist</a>	08/23/2002	Vice President of Finance	In Process	Low
<a href="#">Update project status</a>	08/30/2002	Vice President of Finance	New	High

[Add a Task](#)

[Routed Task Details](#)

Tasks page

## Describing Pagelet Personalizations

Like homepages, pagelets can be personalized in different ways. The data displayed can be automatically filtered to show only data relevant to a particular user. The user can also explicitly personalize the data shown by setting options on a personalization page.

Personalization pages are another type of pagelet extension. However, personalization page pagelet extensions are different from previously discussed pagelet extensions in the following ways:

- They are not accessed by clicking a button or link on the pagelet. Instead, clicking the Customize button on the pagelet header take you to a personalization page where you can define user options specific to that pagelet.
- Setting values on the personalization page will change the way data is displayed on the pagelet for that particular user. Detail pages usually display data in read-only mode or allow you to change application data.

---

**Note.** Any text that the pagelet designer entered during Step 3 in Pagelet Wizard appear as personalization instructions on the Personalize page.

---

## Personalize Tasks

Use this page to specify the default values used in displaying your tasks.

**Display Preferences**

**Tasks Displayed:** In Process

**Max Number of Displayed Rows:** 05

Save    [Return to Home](#)

Personalize Tasks page

The Personalize Tasks personalization page for the Tasks pagelet can be used to change the type and number of tasks shown on the pagelet. Clicking Save stores these values and returns you to the homepage. The Tasks pagelet now reflects these changes.

**Tasks**

In Process Task(s) sorted by Due Date

Task	Due
<a href="#">Finish Quarterly Report</a>	08/09/2002
<a href="#">Clear Q2 worklist</a>	08/23/2002

Add a Task
[Show All/Enhanced...](#)

Tasks pagelet

---

## Understanding PeopleSoft Portal Architecture

This section provides overviews of:

- PeopleSoft portal architecture.
- Client access.
- The web server.
- The application server.
- The database server.
- The portal registry.

- Portal components.
- Page assembly.

## Describing PeopleSoft Portal Architecture

When developing pagelets, it is important to understand the overall architecture of the PeopleSoft portal, as well as the PeopleSoft Pure Internet Architecture. This will enable you to integrate your pagelets in the most efficient manner.

The PeopleSoft Pure Internet Architecture is the server-centric execution architecture used to deploy PeopleSoft internet applications to end users who access the applications through a web browser. This next generation architecture leverages a number of internet technologies and concepts to deliver simple, ubiquitous access to PeopleSoft applications and enable the open flow of information between systems.

Using PeopleSoft Pure Internet Architecture as the foundation, you are able to provide a wide range of end users with access to PeopleSoft applications over the web, as well as more easily integrate your PeopleSoft applications with existing internal systems and external trading partner systems.

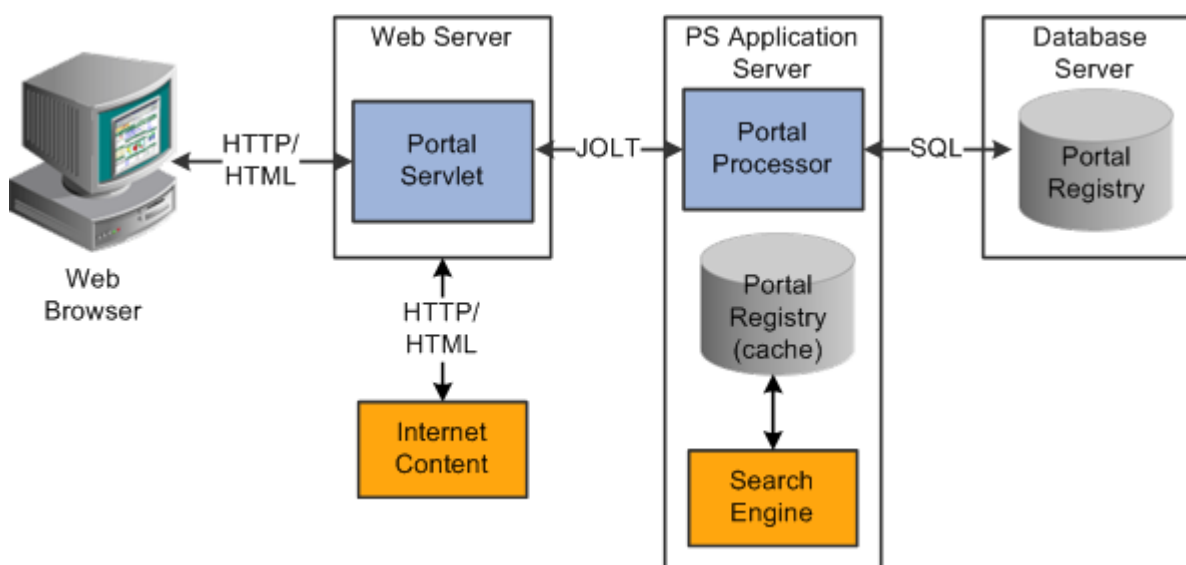
This diagram highlights the primary portal components of the PeopleSoft Pure Internet Architecture.

- Web browser.
- Web server.
- Application server.
- Database server.

---

**Warning!** Always refer to the Enterprise PeopleTools Hardware and Software Requirements Guide for supported configurations. Versions and products supported can change frequently.

---



Primary PeopleSoft Pure Internet Architecture portal components

## Describing Client Access

PeopleSoft Pure Internet Architecture is a completely server-based architecture. Clients to this architecture can be nearly any kind of internet access device:

- Web browser running on supported operating system.
- Wireless device or cell phone.
- External or third-party system with extensible markup language (XML)/hypertext transfer protocol (HTTP) protocols.

There are no PeopleSoft executables on the client, thus the client can be any internet device that uses standard internet technologies such as HTTP, hypertext markup language (HTML), and XML to communicate with the PeopleSoft internet application server.

A web browser running on a PC is the most common internet client. The PeopleSoft internet application server simply serves HTML and JavaScript to the web browser and the end user navigates through the PeopleSoft application as if they are navigating any other website.

A key concept of PeopleSoft Pure Internet Architecture is that there is no complex client software installation. The internet client device accessing the internet architecture already has all of the software and configuration it needs. No additional software must be installed on the client for interaction with PeopleSoft applications. For example, no Java applets, Windows .DLLs, or browser plug-ins are needed.

## Describing the Web Server

The web server acts as the front end of PeopleSoft Pure Internet Architecture. When a client connects to the server by way of a URL, the system displays a signon screen sent to the browser in HTML. The web server manages communication with the browser.

The following web servers products can be configured to deploy your PeopleSoft applications:

- Oracle WebLogic server.
- IBM WebSphere server.

There are two key PeopleSoft servlets running on the web server – the presentation relay servlet and portal servlet.

### ***Presentation Relay Servlet***

The presentation relay servlet is used to process all inbound and outbound HTTP requests for PeopleSoft transactions and queries. This very thin servlet acts as a relay between the client device and the core back-end services on the application server. It receives and serves HTML, XML, and wireless markup language (WML) requests over HTTP.

## **Portal Servlet**

The portal servlet is a Java servlet that runs on the portal web server. It intercepts user requests for HTML pages, retrieves the requested page, wraps additional content around it, and then sends the resulting page to the user's browser. The servlet acts like an invisible browser that sits between the user's browser and requested content.

The portal servlet:

- Provides a consistent user interface.

The portal servlet checks properties associated with each content reference, including the name of a portal template. When a user accesses content through the portal, the portal servlet wraps the target page with the portal template specified in the content reference. This template provides a consistent user interface.

- Ensures that PeopleSoft-specific tags are processed correctly.

Developers create portal pages using a template-based layout system. In addition to traditional HTML tags, templates can contain PeopleSoft-specific tags that a normal browser cannot interpret. At runtime, the portal servlet can interpret these PeopleSoft-specific tags when constructing templates, as well as any other HTML content. The portal servlet then sends the resulting page to a browser as a single HTML document.

One of the most important aspects of portal technology is its role in integrating content from a wide variety of data sources and displaying that content on a single page in a coherent, understandable, and presentable way. We refer to this complex process as "page assembly and proxying." Portal processing assembles the page to be displayed based on the content retrieved from various data sources. It uses portal templates to wrap the contents of the assembled document into a single page that fits into the context of the site.

- For page-based templates, the portal servlet assembles pages for the browser.

It ensures that all URL references in the HTML on the assembled page are references back to the portal servlet itself. In some cases, each URL in the HTML document assembled by the portal servlet must be rewritten to reference the portal servlet, not the originally requested URL. This process of redirecting URLs so that they point to the portal servlet is called proxying.

- For frame-based templates, the portal server updates the "src" tags in the frameset with the target content and sends it to the browser.

When working with a frame-based template, the portal servlet inserts a URL into each frame in the "src" tag and sends it to the browser, rather than retrieving documents for the browser as it does with page-based templates.

## **Describing the Application Server**

The application server is the core of PeopleSoft Pure Internet Architecture; it executes business logic and issues SQL to the database server. The application processing logic that ran on the client in previous releases now runs on the application server. The application server consists of numerous PeopleSoft services and server processes that handle transaction requests. These include requests to:

- Authenticate users.
- Build application pages.

- Save application pages.
- Run some PeopleCode.
- Execute SQL (prompts, page-specific SQL, validations, and so forth).

The application server is responsible for maintaining the SQL connection to the database for the browser requests and the Windows development environment. PeopleSoft uses Tuxedo to manage database transactions and Jolt, Tuxedo's counterpart, to facilitate transaction requests issued from the internet. Both Tuxedo and Jolt are products of Oracle Systems.

The Portal Processor executes as an application service of the PeopleSoft application server. It executes with the other application services, like the Component Processor, Security Manager, and SQL Access Manager. Portal Processor tasks include:

- Fetching content templates from the database.
- Fetching content references from the database portal registry and caching them in the application server portal registry.
- Processing personalizations.
- Interacting with other application services (lightweight directory access protocol (LDAP), role-based security, and so forth).

## Describing the Database Server

The PeopleSoft database is the repository for all information managed by PeopleSoft applications. Not only is application data stored in the database, but PeopleSoft metadata is also maintained in the database. PeopleSoft Application Designer enables you to define and maintain this metadata that the system uses to drive the run-time architecture. The application server executes business logic based on the PeopleSoft metadata.

At execution time, the application server fetches the most recent application object definitions from the metadata repository, compiles and caches the application object into memory, and executes the business rules based on the definition.

---

**Note.** In general, PeopleSoft Enterprise Portal 9.x can retrieve content from any RDBMS-based application as long as the content that is being retrieved is URL-accessible and is HTML-based. Always refer to the Enterprise PeopleTools Hardware and Software Guide for your release and the Supported Platforms database on Customer Connection for supported RDBMS products.

---

## Describing the Portal Registry

The portal registry is a key administrative component within the metadata of a PeopleSoft database. A portal registry is a hierarchical structure in which URLs accessed by way of the portal are organized, classified, and registered. Each portal registry consists of the following objects:

- Folders.

Folders group and organize content references into a hierarchy. With the exception of the root folder, each folder has a parent folder. Each folder can contain content references as well as other folders.

Every portal registry contains a root folder and a Portal Objects folder. The Portal Objects folder contains administrative objects specific to the portal and includes the following folders: Templates, Pagelets, and Homepage.

In addition to these standard folders, there are typically several folders located directly under the root folder: one folder for PeopleTools (administrative references) and other main folders for each PeopleSoft application. These main application folders contain the folders and content references associated with each PeopleSoft application you've licensed.

- Content references.

Content references are URLs that have been registered in a portal registry. They can be PeopleSoft application pages or external web pages. Content references fall into four main categories: pagelets, target content, templates, and homepage tabs.

In addition to specifying a URL, each content reference includes additional information such as its creator, effective dates, associated template, search keywords, and so forth. Registry URLs can point to any website that responds to HTTP requests with an HTML response – in other words, a static or dynamic web page.

For example, a content reference could be a URL pointing to a page in a PeopleSoft application. Or a content reference could point to static HTML pages on an intranet site, such as a procurement or expense policy document, or dynamic pages created by an analytic or reporting system. Access to content references is controlled by security permission lists assigned to each content reference. Any portal content can be limited to a specified group of users, or made public and available to any authorized portal users.

- Nodes.

Nodes refer to a source of HTML content. They primarily identify the universal resource indicator (URI). A URI is a subset of the URL that points to the location of the resource. It does not include the content information, such as the target file or application and any parameters passed to that resource.

The portal registry's hierarchical structure enables a portal administrator to more easily create a classification and navigation system in which content references can be registered and managed for all portal users. PeopleSoft provides a Portal Administration web page for this purpose. Additionally, the portal includes a registry application programming interface (API) for maintaining the portal registry from PeopleCode.

To improve performance, a copy of the registry is also stored on the application server in cache files. This is how the portal servlet accesses the registry at runtime.

## Describing Portal Components

The portal technology framework includes the following services:

- Personalization.

Pagelets can optionally be assigned a user personalization settings page. This allows the user of the pagelet to specify selections, settings, or attributes specific to the pagelet's content. These personalizations commonly alter the content of the pagelet. An example of this would be setting the city for which a weather pagelet displays forecast information. A personalization page is accessible to the user by clicking on the Customization button on the pagelet's frame header.

- Search.

The portal uses a search engine to quickly search for registered content in the portal registry. This is a popular means of portal navigation. Verity, the industry leading search engine, is packaged with the portal technology. PeopleSoft's search capabilities were built to assume multi-language support, including double-byte languages.

- Navigation.

PeopleTools base portal technology provides a set of navigation components based on the portal registry. These components are the drop-down menu and Favorites. Navigation has been engineered to provide rapid access to complex information based upon the role of the user.

- Menu navigation.

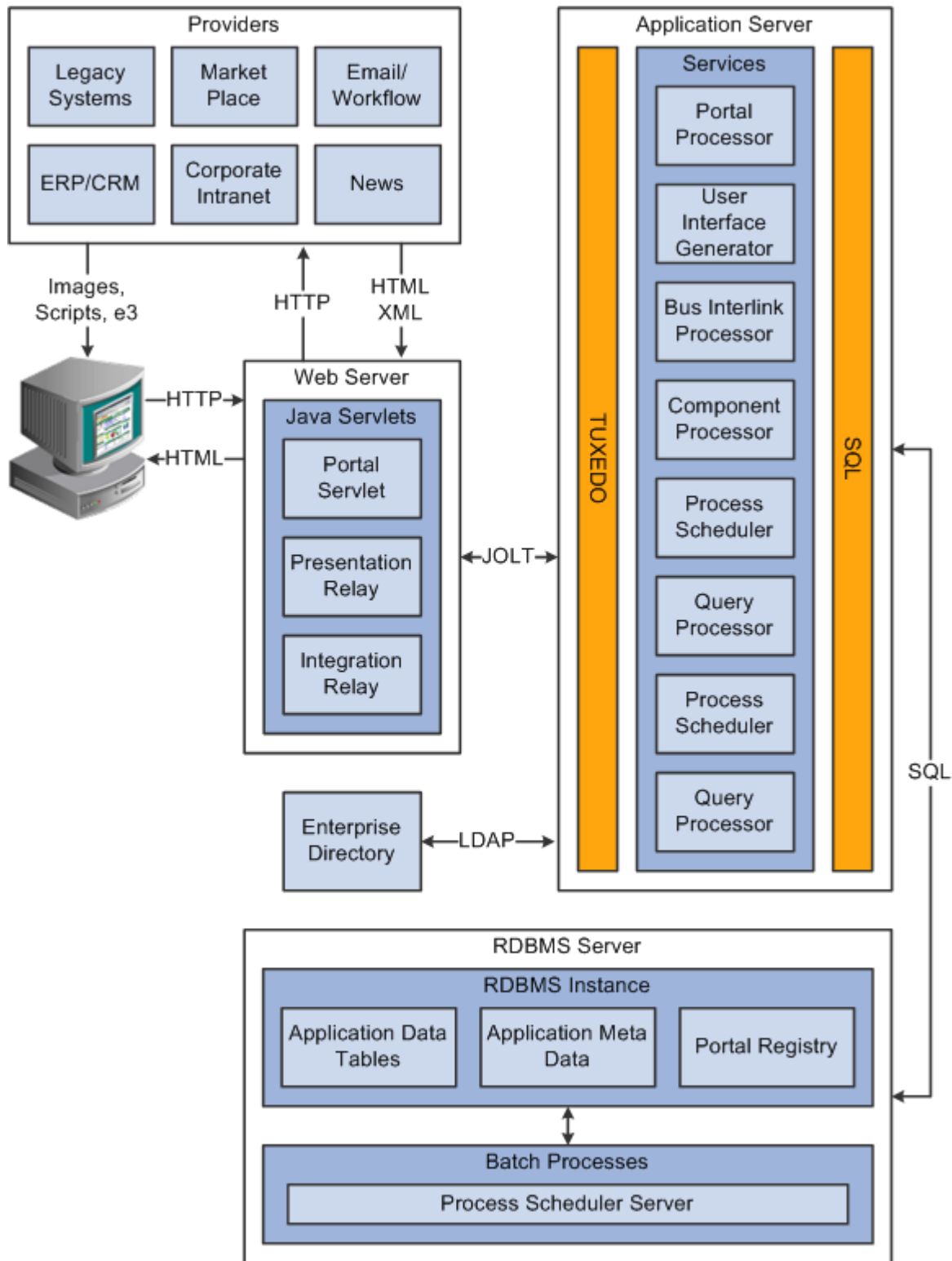
PeopleTools base portal menu navigation provides a consistent method of content access, categorization, and organization. The menu navigation presents a dynamic hierarchy-based view of the folders and references within the portal registry. The menu is available both as a pagelet on homepage tabs and as a fixed entry in the left frame of associated templates.

- Favorites.

The PeopleTools base portal menu contains a Favorites folder that is used for shortcut, or bookmark-style, storage of frequently accessed links.

## Describing Page Assembly

Now that we have looked at the individual components of PeopleSoft Pure Internet Architecture, let's take a look at the big picture.



PeopleSoft Pure Internet Architecture processing flow

1. An HTTP request is sent from the user's browser to the portal's web server, which invokes the portal servlet. Let's say that the browser is requesting a PeopleSoft application page.

2. The portal servlet makes a request to the Portal Processor to fetch the content reference for the hyperlink selected by the user. The Portal Processor fetches the content reference from the portal registry and passes a partially completed HTML document back to the portal servlet.
3. The content reference could be pointing to any or several of the content providers (specified by a node). Each content reference is referenced in a partially completed HTML document. The portal servlet issues the HTTP request to the appropriate web server and retrieves the HTML document.
4. The portal servlet then completes the page assembly process with the template, merging the content from the HTML document, and then serves the completed page back to the user's web browser.

---

## Understanding Pagelet Development

This section provides overviews of:

- Pagelet development guidelines.
- Pagelet development considerations.
- Pagelet development options.
- Size considerations.
- HTML design considerations.
- Branding considerations.

## Describing Pagelet Development Guidelines

Follow these guidelines when developing new pagelets:

- Adhere to object naming standards.
- The pagelet must display without requiring any search interaction. Do not use search keys or default search key values. You can accomplish this in one of two ways:

Assign a search record to the component with no keys. Therefore there is only one row of data.

Add SearchInit PeopleCode that would put in default key values and call the function SetSearchDialogBehavior() to bypass the search interaction.

---

**Note.** If the user ID is needed for the pagelet, or template pagelet, then an alternative such as OPERID\_VW can be used.

---

- Use the *Unlimited Occurs* option in all grids and scroll areas.

Control the number of retrieved rows with the PeopleCode function *StopFetching()* in the RowSelect event of the selected record. Match the *StopFetching()* function when the user-specified or default value of the maximum displayed number of rows is reached. Pagelets should display no more than five rows by default.

- Use deferred processing.

FieldChange/Prompts/FieldEdits other than button or link page transfers are not allowed for pagelets. These types of processing require trips to the application server, which causes the portal to redisplay any pagelet as a target page (wrapped in the default template). Therefore, such processing would result in the pagelet being displayed outside the context of the homepage. To work around this limitation you need to disable two options:

- Deselect the Cache Generated HTML check box in the On the Browser group box of the Web Profile Configuration - Caching page. PeopleSoft page caching is enabled by default.

See [Chapter 10, "Configuring the Portal Environment," On the Browser, page 236.](#)

- Deselect the Cache Target Content check box in the on the Web Server group box of the Web Profile Configuration - Caching page.

See [Chapter 10, "Configuring the Portal Environment," On the Web Server, page 238.](#)

- Do not provide any filter, sort, or refresh buttons as they also require trips to the application server.

Because the portal servlet only performs page assembly and proxying on blocks of HTML, a pagelet must be:

- URL-accessible.

The portal servlet will issue an HTTP request to the registered URL.

- HTML 4.0 compliant.

The HTML returned will be combined with that from other pagelets to form the entire web page.

- Sized appropriately.

The pagelet's width should conform to the narrow or wide guidelines discussed in the Sizing section in this document.

See [Chapter 12, "Building Pagelets," Describing Size Considerations, page 412.](#)

- JavaScript namespacing.

If you write custom JavaScript code, ensure that JavaScript from multiple portlets can coexist on the same page. There is only one namespace that is shared by all portlets on a portal page. For example, if portletA and portletB, both use the GETURL function, there will be only one definition for GETURL defined by the most recently processed portlet. If you need a different GETURL function for each portlet, then you must uniquely name the GETURL function for each portlet. As in, portletA calls GETURL1 and portletB calls GETURL2. If the same GETURL function is called by both portlets, then you don't need to rename GETURL as the most recent definition of GETURL will be used by both functions.

Pages created by way of PeopleSoft Pure Internet Architecture conform to these requirements. However, these requirements are not exclusive to PeopleSoft. As mentioned, pagelets can also be from remote, non-PeopleSoft systems, such as static HTML files or dynamically generated HTML from a variety of technologies such as JavaServer Pages (JSPs), Java Servlets, Microsoft Active Server Pages (ASPs), Perl scripts, and CGI programs.

## Describing Pagelet Development Considerations

When developing pagelets, you should also take the following topics into consideration.

## Single Signon

PeopleSoft delivers a component interface named PRTL\_SS\_CI that enables external applications to support a single signon solution with PeopleSoft portal applications. This allows users to sign on once when entering the portal and not have additional signons for each system that is referenced in your portal.

See [Chapter 12, "Building Pagelets," Understanding Single Signon and Pagelets, page 417.](#)

## Layout

Use the following rules for layouts:

- Avoid horizontal scrolling.
- Do not use page titles.

The name of the content reference in the portal registry is used as the default for the pagelet's title.

- Pagelet instructions should not be necessary.
- Avoid group boxes whenever possible.

---

**Note.** They may be necessary to separate sections.

---

- Left-align text wherever possible.
- Clear the *Show Borders* option for group boxes and scroll areas.
- Select *View, Internet Options* in the PeopleSoft Application Designer to ensure that you have access to all the styles available.
- Leave more than 20 percent spacing between field labels and field edit boxes because the rendered styles are larger in the browser than they appear in PeopleSoft Application Designer.
- Before bringing your pagelet into the portal view it in PeopleSoft Application Designer. Select *Layout, View in Browser*.

## Look and Feel

Objects on pagelets, such as text, hyperlinks, tables, and so forth, should be modified using the PeopleSoft style classes to retain a consistent look and feel between your pagelet and the rest of the portal. PeopleSoft utilizes Cascading Style Sheets (CSS). Style sheets are useful because they enable developers to change page attributes across multiple pages quickly and easily.

Each style sheet is composed of individual classes, which affect the formatting of each page. Classes control a broad range of characteristics, including font size, spacing, alignment, border width, weight, and color.

When creating the page, select the *Use Default Style Sheet* option. For any controls on the page select the *Use Default Style* option.

All style sheets used in each pagelet on the page are referenced on the final page that is assembled by the portal. Therefore, a pagelet should not redefine any class that might be shared by other pagelets on the assembled page. If a class needs to be changed, define a new one.

The order of precedence for style sheets is:

1. Template.
2. Target component.
3. Pagelets - in the order that they are assigned.

---

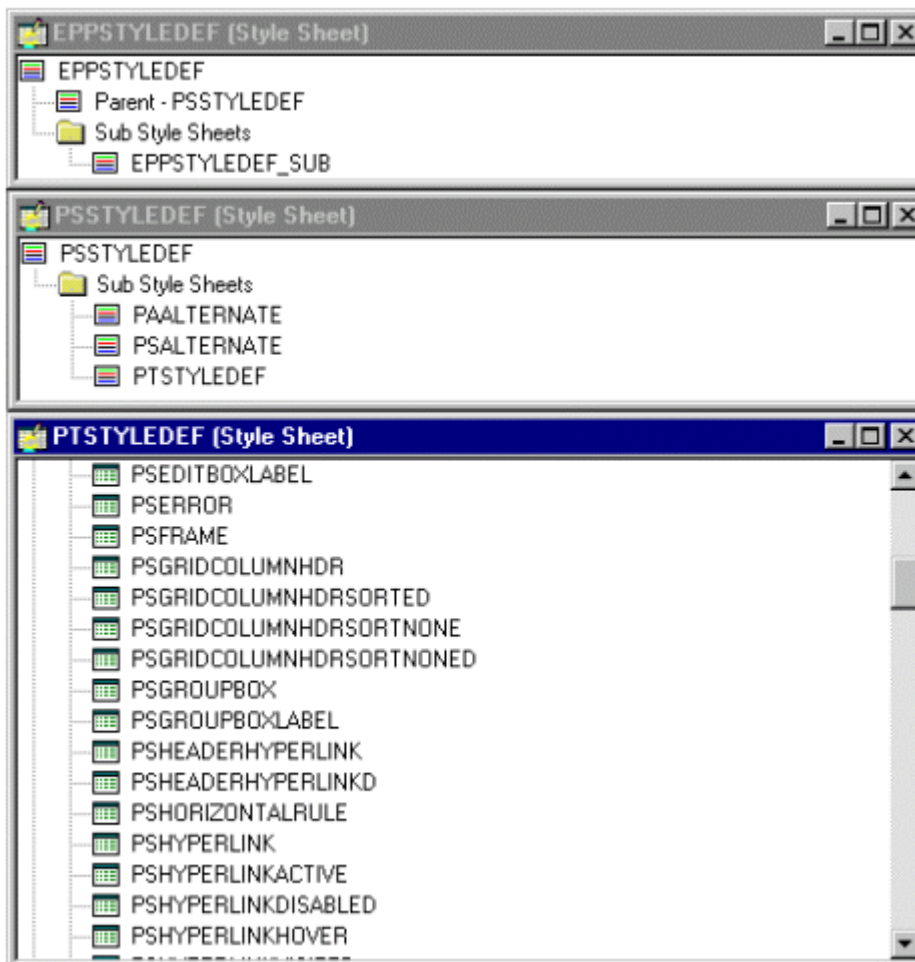
**Note.** In the same way that PeopleTools supports subrecords to enable shared definitions between records, you can define sub-stylesheets to share a common set of classes. A substyle sheet has all of the properties of a style sheet.

---

The definitions are actually stored in the database. They are accessed and modified using PeopleSoft Application Designer. The basic style sheet used by PeopleTools is called PSSTYLEDEF. The PeopleSoft Enterprise Portal uses a derivative version called EPPSTYLEDEF.

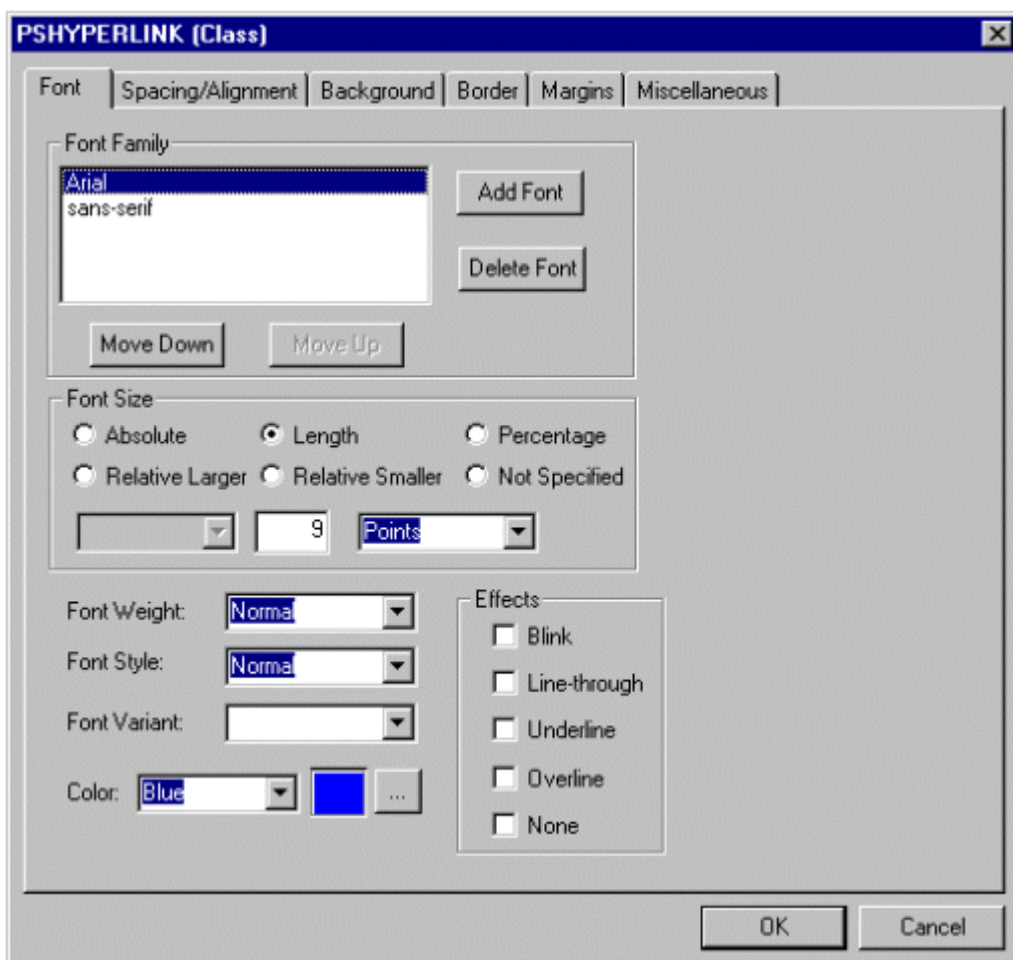
You can determine which style sheet is being used by navigating to PeopleTools, Utilities, Administration, PeopleTools Options. The value in the Style Sheet Name field designates the style sheet used by the database.

There are two ways in which you can see the style classes defined in a style sheet. If you have access to PeopleSoft Application Designer, you can open the designated style sheet and view its definitions in a graphical interface.



Style sheet definitions

Double-click on any style class to be able to view and modify its definition.



### Style class definition

If you don't have access to PeopleSoft Application Designer and want to see the CSS file, you can access the file cached for the CSS on the web server. For example, on WebLogic you'll find the file in this directory:

....\peoplesoft\applications\PORTAL\ps\cache.

The file is generally named EPPSTYLEDEF\_1.css, however it may have a slightly different name. For example, it might have a different numbered suffix.

Some of the most common styles used by pagelets include:

- PSTEXT  
General text label.
- PAADDITIONALINSTRUCTIONS  
Instructional text.
- PABOLDTEXT or PABOLDBLUETEXT  
Commonly used for group and category titles.

- PPSMALLTEXT  
Commonly used for footers and branding (credits).
- PSPAGE  
Commonly used for background colors.
- PSLEVEL1GRIDODDROW  
Shaded row, which is typically light gray or light blue.
- PSLEVEL1GRIDEVENROW  
Lighter-colored row, which is typically white.
- PSCHECKBOX
- PSDROPDOWNLIST
- PSEDITBOX
- PSGROUPBOX  
Commonly used to create groupings or dividers.
- PSHYPERLINK
- PSPUSHBUTTON
- PSRADIOBUTTON

Following are some examples of the ways in which these styles are used in HTML and translated into elements in the PeopleSoft Pure Internet Architecture interface:

This HTML generates the following interface:

```
<table border='0' cellpadding='0' cellspacing='0' cols='1'=>
  CLASS='PSGROUPBOX' style="border-style:none;" width='639'>
<tr><td class='PSGROUPBOXLABEL' align='LEFT'>Language Settings</td></tr>

...

<td colspan='2' valign='top' align='LEFT'>
<label for='PSOPTIONS_LANGUAGE_CD' class='PSDROPDOWNLABEL'=>
  >Language Code:</label>
</td>
<td valign='top' align='LEFT'>
<span class='PSDROPDOWNLIST' >English</span>
</td>

...

<input type='check box' name='PSOPTIONS_TRANS_CHG_LASTUPD'=>
  id='PSOPTIONS_TRANS_CHG_LASTUPD' tabindex='14' value="Y" />
<label for='PSOPTIONS_TRANS_CHG_LASTUPD' class='PSCHECKBOX'=>
  >Translations Change Last Update</label>
```

Language Settings		
<b>Language Code:</b>	English	<input type="checkbox"/> Translations Change Last Updat
<b>*Sort Order Option:</b>	Binary Sorting	

Language Settings on the PeopleTools Options page

This HTML generates the following interface:

```
<input type='text' name='PSOPTIONS_TEMP_TBLINSTANCES' =>
  id='PSOPTIONS_TEMP_TBLINSTANCES' tabindex='28' value="" =>
class='PSEDITBOX' style="width:23px; text-align:RIGHT;=>
  " maxlength='2' />
<label for='PSOPTIONS_TEMP_TBLINSTANCES' class='PSEDITBOXLABEL'=>
  >Temp Table Instances (Total):</label>
```

**Style Sheet Name:**  

Style Sheet Name field on the PeopleTools Options page

This HTML generates the following interface:

```
<tr align="left" class="PSLEVEL1GRIDODDROW">
<td>3</td>
<td>Enterprise Menu</td>
</tr>
<tr align="left" class="PSLEVEL1GRIDEVENROW">
<td>2</td>
<td>My Stock Chart</td>
</tr>
```



<u>Users</u>	<u>Portal Label</u>
3	Enterprise Menu
2	My Stock Chart
2	Company News
1	Popular Pagelets
1	PSFT YTD Stock Chart
1	RSS News
1	Weather Magnet
1	My DTheatre News
1	My Moreover News Search
1	My Moreover News Search Box

[Click here for more...](#)

powered by  
**PeopleSoft**

Popular Pagelets pagelet

### ***Hyperlinks and URLs***

When processing page-based templates, the portal servlet uses a process called proxying to help ensure that users always stay within the context of the portal and that familiar portal features such as the universal navigation header do not disappear when a user clicks a link.

When a user logs in to a PeopleSoft portal, they log in to a web server on which the portal servlet is running. The portal servlet processes all HTML that isn't in the simple URL format, converting all URL references to point to the portal web server rather than the original URL. The original URL is still necessary to retrieve the requested content; it is stored in the new URL in the URL query string parameter. The portal servlet proxies all links and all form actions in this manner.

For example, imagine that a user requests a page from an external website through a proxied link in the portal. The request arrives at the portal web server, invoking the portal servlet. The portal servlet then programmatically retrieves the page from the web server associated with the requested page. It proxies all the links on the retrieved response and sends the page (the contents of the HTTP response) back to the browser, formatted as the user would expect within the portal.

---

**Note.** If URLs are included in your HTML, you must use absolute URLs as opposed to relative URLs.

---

When a URL is invoked on a target page, as opposed to the homepage, the content associated with the URL is rendered within the target frame. The PeopleSoft Enterprise Portal and header and left-hand area will remain. Therefore, proxying isn't required to have the new content rendered in the context of the PeopleSoft Enterprise Portal.

## Performance

Homepages are cached on each user's browser. The browser does not access the web server after the homepage is initially retrieved. You can turn this feature on or off, and also adjust the time interval before the web server is accessed again to get a fresh homepage. In any case, if a user clicks the browser's Refresh button, the homepage is accessed from the web server again, overwriting the homepage cached on the browser.

The following PeopleCode function is used to trigger a refresh of the homepage:

```
FUNCLIB_PORTAL.TEMPLATE_FUNC.FieldFormula.ForceRefreshHomePage().
```

Additionally, the following configuration properties are associated with homepage caching. Any changes to these settings are applied to all users signing on to the web server.

- PortalCacheHomepageOnBrowser=<*True or False*>

If set to *True*, the homepage is cached on the browser. If set to *False*, the homepage is not cached on the browser.

- PortalHomepageStaleInterval=<*seconds until stale*>

A homepage cached on the browser is considered stale after the specified number of seconds. The next time a user accesses the homepage by clicking on a link, the web server is accessed and the homepage is refreshed on the browser.

Because different browser versions do not process HTML in exactly the same way, the browserprops.xml file on the web server on which the portal servlet is installed enables you to turn off homepage caching for selected browser versions.

This can be useful if you have one or two supported browsers and want to disable cache for non-standard browsers that could pose an administration problem. Follow the instructions in the file to disable caching for certain browser types.

As with homepages, navigation pages are cached on each user's browser. You can set options for navigation caching by using the Time page held in cache (METAXP) option.

Set the Time page held in cache option navigating to My Personalizations and clicking the Personalize Option button for the General Options personalization category. Note that this option is set in minutes, not seconds. A change to this option is picked up by the application server immediately. However, since the users' browsers already have cache control set by the previous value of the option, you have to delete the browser cache for the new Time page held in cache value to take effect.

PeopleSoft provides the option to prevent the system from caching PeopleSoft pages to the browser. You control browser caching using the EnableBrowserCache property in the configuration.properties file.

Being able to control browser caching is useful for situations where PeopleSoft applications are deployed to kiosk workstations where multiple users access the applications. Enabling you to prevent caching means that users can't click the Back button to view another individual's transaction or view any other sensitive data.

The side effect of turning off caching completely is degraded performance. For each new page, the system must access the database. However, PeopleTools offers a compromise related to browser caching in the form of the Time page held in cache (METAXP) option discussed earlier.

You can turn on browser caching for the navigation pages that remain relatively static. This option applies to PeopleSoft Pure Internet Architecture navigation pages, portal homepages, and navigation pages. Use it to take advantage of the performance gains of caching while limiting the amount of time that navigation pages—the menu pages—remain in cache.

The Time page held in cache option is set to 20 by default. To disable this option, enter 0 in the Override Value edit box. The minimum value in minutes for this option is 0 (disabled) and the maximum value is 525600, which is one year.

If the Time page held in cache option is set to 20, and if you assume that the time is 7 p.m. on February 24, then the header information in the HTML is dated as follows:

Sun 24 Feb 2004 07:20:00 PM

This header information indicates that in 20 minutes the system needs to check for a new page. This reduces the performance degradation when there is no caching at all.

By default the EnableBrowserCache property is set to *True*.

If the EnableBrowserCache property is set to *False*:

- The system never caches pages. When a user clicks the browser Back button, she receives a Data Missing message in Netscape and a Page Expired message in Internet Explorer.
- The setting overrides any date/time header settings.

The following table helps illustrate the way in which the EnableBrowserCache option works with the METAXP option.

<b><i>EnableBrowserCache</i></b>	<b><i>METAXP</i></b>	<b><i>Caching Behavior</i></b>
<i>True</i>	0	No caching due to the 0 value in METAXP.
<i>True</i>	> 1	Pages are cached with expiration values set in Greenwich Mean Time (GMT) based on the Time page held in cache value (METAXP).
<i>False</i>	0	No caching.
<i>False</i>	>1	No caching. The EnableBrowserCache option setting overrides the Time page held in cache value (METAXP).

### ***Multi-Language Support***

PeopleSoft currently supports the following languages:

- English.
- Danish.
- Dutch.

- French.
- French-Canadian.
- German.
- Greek.
- Italian.
- Japanese.
- Korean.
- Portuguese.
- Chinese.
- Spanish.
- Swedish.
- Thai.

The Pagelet Wizard can access the PeopleSoft run-time environment to determine details of a user's profile information, such as their language. For more information on how you can use the Pagelet Wizard feature to facilitate passing run-time parameters, such as language, see Pagelet Wizard documentation in this PeopleBook.

See [Chapter 11, "Using Pagelet Wizard," page 277](#).

In addition, Java classes delivered with PeopleTools enable you to call PeopleCode from your Java program and access contextual information from the run-time system. If needed, language can be retrieved through a PeopleCode function that is accessible from Java. Further information about this can be found in this document.

See [Chapter 12, "Building Pagelets," Developing Pagelets Using Java, page 428](#).

## Describing Pagelet Development Options

There are several ways to create a pagelet. Ultimately, the portal servlet assembles HTML, so the key is determining how to generate the HTML. Some methods leverage PeopleTools, while other options allow pagelet creation without PeopleTools.

One set of options is to develop pagelets with PeopleTools. This is the most straightforward approach if you are dealing solely with data from PeopleSoft applications. The two types of PeopleTools-based pagelets are:

- PeopleSoft Pure Internet Architecture pages.

PeopleTools dynamically generates the appropriate HTML to render the page data based on the definitions created within PeopleSoft Application Designer. This is the most straightforward approach when the data being rendered is in a PeopleSoft application database.

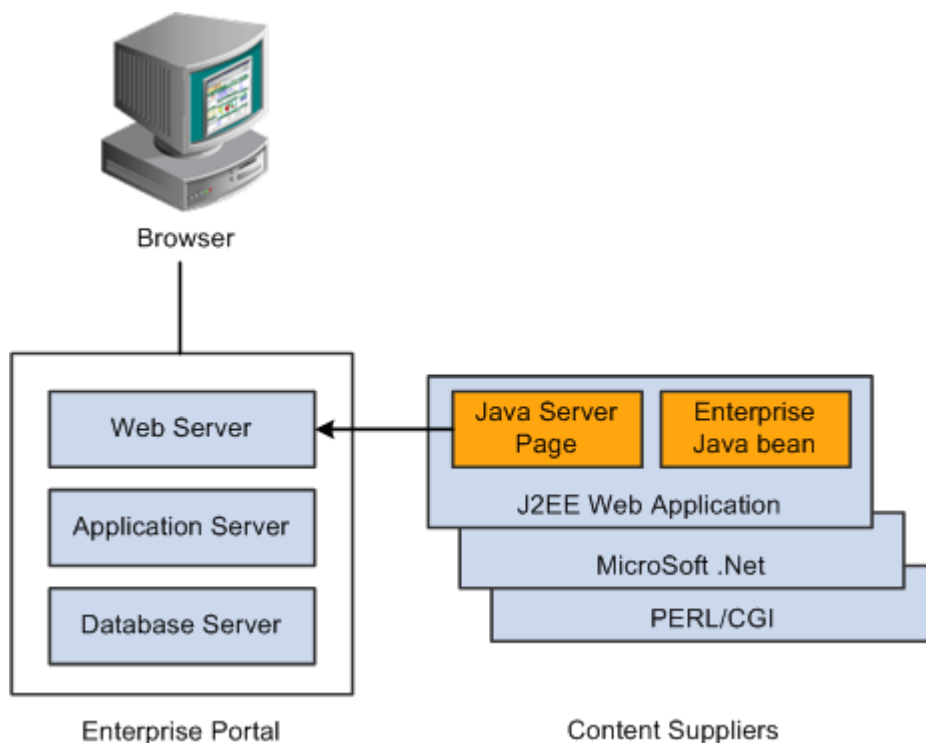
- iScripts.

You can write a PeopleCode function that renders HTML using %Request and %Response objects, which are similar to ASP or JSP programs. This allows more control over data retrieved and allows you to conditionally render HTML. This approach gives you maximum flexibility, but unlike the PeopleSoft Pure Internet Architecture page approach, you must code for things such as multi-language and browser support.

The focus of this document is a set of options that can be used to create a pagelet application with one of the many technologies that generate HTML. This may be the approach you take if you're integrating a non-PeopleSoft system with the PeopleSoft Enterprise Portal. For example, you're applying business logic to data from a non-PeopleSoft system, or you might be combining data from a PeopleSoft application with other systems.

The pagelet can be written in a variety of technologies, including:

- JSP.
- Servlets.
- ASP.
- HTML.
- JavaScript.
- Internet server API (ISAPI).
- CGI.
- Perl.
- Macromedia Cold Fusion.
- Tool Command Language (Tcl).



Developing in a PeopleSoft environment with non-PeopleTools technologies

There are two methods you can use to turn your own HTML into a pagelet:

- Pagelet Wizard.

The Pagelet Wizard walks you through a series of steps involved in creating and publishing a pagelet. Portal administrators and non-technical users are able to integrate and transform data from a wide variety of data sources, both internal and external to PeopleSoft applications.

External sources include web applications that can be referenced with a URL, HTML block (such as a form or table), and Java classes. You do not have to have PeopleSoft-specific application development tools or skills to be able to create pagelets.

- Registered URL.

You can also choose to register a URL, like a JSP or ASP, directly in the portal registry.

## Describing Size Considerations

As mentioned, the portal homepage is delivered to display two possible column widths: narrow and wide.

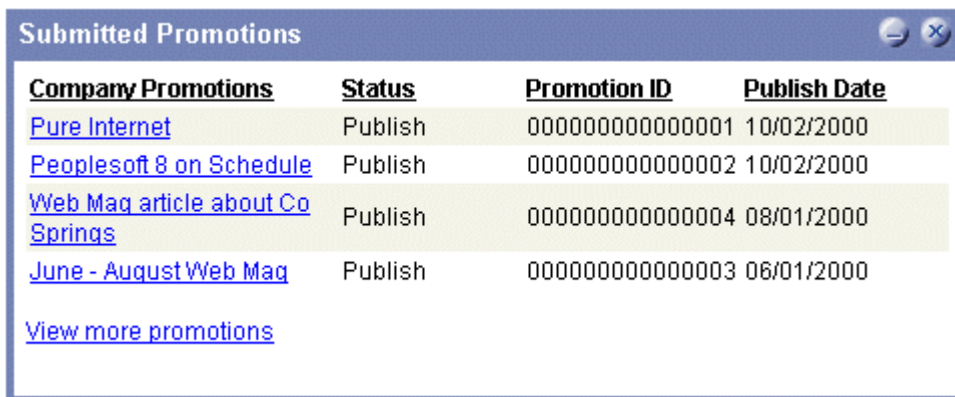
Narrow pagelets are designed to be 240 pixels wide. Subtracting the border and the internal margin leaves 218 pixels for the content. A narrow pagelet appears in a column that is one-third the width of the portal's homepage. Because the pagelet is narrow, you should provide a succinct list of values that users can quickly traverse. Select the minimum pieces of data that best encapsulate the pagelet's content. A narrow pagelet typically accommodates a single 30-character field. Generally, all pagelets should be developed with the narrow column size as the default.



<u>Company Promotions</u>	<u>Status</u>
<a href="#">Pure Internet</a>	Publish
<a href="#">Peoplesoft 8 on Schedule</a>	Publish
<a href="#">Web Mag article about Co Springs</a>	Publish
<a href="#">June - August Web Mag</a>	Publish
<a href="#">View more promotions</a>	

Example of a narrow pagelet

Wide pagelets are designed to span two-thirds of the width of the homepage. Although you can fit more data on the pagelet, the data must remain meaningful. Subtracting the border and the internal margin leaves 468 pixels for the content. A query string parameter is passed denoting the wide version. If the pagelet does not check nor use this parameter, the narrow pagelet will just be rendered in a wider area. Developing a wide version of a pagelet is optional.



<u>Company Promotions</u>	<u>Status</u>	<u>Promotion ID</u>	<u>Publish Date</u>
<a href="#">Pure Internet</a>	Publish	0000000000000001	10/02/2000
<a href="#">Peoplesoft 8 on Schedule</a>	Publish	0000000000000002	10/02/2000
<a href="#">Web Mag article about Co Springs</a>	Publish	0000000000000004	08/01/2000
<a href="#">June - August Web Mag</a>	Publish	0000000000000003	06/01/2000
<a href="#">View more promotions</a>			

Example of a wide pagelet

The homepage layout can vary based on end-user personalizations. An end user can decide to have a two- or three-column homepage layout. With three-column layout, all pagelets are rendered as narrow pagelets. With two-column layout, the first column allows spacing for a narrow pagelet and the second column allows spacing for wide pagelets.

A pagelet that has been designed to fit in a narrow column can be rendered in a wide column as well. However, there may be user interface issues when a pagelet that has been designed to render in a wide column only, is rendered in a narrow column. Therefore, a pagelet should always be able to operate in a narrow format. Optionally, a pagelet can be designed to take advantage of the additional space when it is rendered in a wide column.

The query string parameter `PORTALPARAM_COMPWIDTH` can contain the values *Wide* or *Narrow* and can be used to determine whether a pagelet is being rendered in a narrow or wide column.

Here's an example of some code that references the query string parameter `PORTALPARAM_COMPWIDTH`:

```

Component string &CompWidth;

&CompWidth = %Request.GetParameter("PORTALPARAM_COMPWIDTH");
If &CompWidth = "Wide" Then;
TransferPage(Page.EO_PE_CONTRK_PGLTW);
End-If;

```

### ***Pagelet Extensions***

Pagelet extensions are just regular web pages. They can be registered with a template that allows it to use the entire width of the browser with no left-hand frame for navigation, or with a template that includes a left-hand frame.

In the former case, there are no inherent sizing requirements. PeopleSoft pages are designed for browsers running with a resolution of least 800x600. Therefore, when a left-hand frame is not included, a pagelet extension should not be wider than 760 pixels, accounting for borders and so forth.

In the latter case, a pagelet extension should not be wider than 570 pixels.

## **Describing HTML Design Considerations**

Following are some HTML-related pagelet design considerations:

- The pagelet should encapsulate data to provide at-a-glance summary information.
- It should provide links to detailed application pages.
- Avoid large borders or a design that creates extraneous white space. Screen real estate on a homepage is valuable and data should be maximized.
- Avoid designs that make the pagelet wider than the prescribed size. This undesirable design forces the user to scroll horizontally, so design your HTML to be vertically oriented. For instance, radio buttons should be arranged vertically, not side-by-side. When creating a form, buttons should appear below a text box, not next to it.
- This is a business-oriented portal, so avoid extravagant or extraneous graphics. If you use graphics, they should be very small and unobtrusive. No graphic should be wider than 218 pixels, which would force the pagelet to be wider than a narrow pagelet.
- Avoid using any text or HTML tags that force the pagelet to a width greater than 218 pixels.
- In general, avoid explicit sizing. Let the browser render content in the space available, which should allow your text and graphics to fit appropriately. For example, place text in an HTML table to enable the table it to wrap any long strings.
- When appropriate, utilize the PeopleSoft style sheet to ensure that the look and feel of your pagelet is consistent with the rest of the portal content.
- If your pagelet does not utilize the entire width of the column, it's best to center its contents.
- If a hyperlink on the pagelet takes the user to another website and you want to display the content in a new window, rather than using *target=\_blank* in the hyperlink tag, use the following code:

```

<a href="javascript:void window.open
('http://www.company.com/cgi-bin?article3');" class="PSHYPERLINK">

```

- If a pagelet has personalization options, then it should have a default mode. Before a user personalizes the pagelet data, the pagelet should display a default set of data and a message conveying that default data can be personalized.

This section provides a listing of HTML-specific development guidelines for pagelets used with the PeopleSoft Enterprise Portal.

- Within PeopleTools, attributes from all content are merged into the single <BODY> tag. The order of attribute precedence is template, target content, and then template components. Events beginning with "on" are naively combined, so if one of them issues a "return," the following events will not run. All other attributes are not examined, rather they are used in order of precedence. For example, if both the template and a content component contain "topmargin," the template attribute will be applied.
- Avoid large borders or anything that creates extraneous white space. Screen real estate on a homepage is valuable and data should be maximized.
- Avoid implementing anything that will make the pagelet width wider than the prescribed size. Making the user scroll horizontally is undesirable. Make your HTML vertically oriented. For instance, radio buttons should be placed vertically, not side-by-side. When creating a form, any buttons should appear under a text box, not next to it.
- Avoid using any graphics, if possible. This is a business-oriented portal, so extravagant or extraneous bitmaps should not be used. If any are used, they should be very small and unobtrusive. No graphic should be wider than 218 pixels. A graphic larger than this size will force the width of the pagelet to exceed the size of a narrow pagelet.
- Avoid using any text or directives that force the pagelet to a width greater than 218 pixels.
- In general, it's best to avoid explicit sizing and allow the portal to set the width. Your text and graphics should then fit appropriately. For instance, placing text within an HTML table will allow it to wrap any long strings.
- Utilize the PeopleSoft stylesheet, when appropriate, to ensure the look and feel of your pagelet is consistent with the rest of the portal content.
- If your pagelet does not utilize the entire width of the column, it's best to center it.
- If a hyperlink takes the user to another Web site and you want it to display the content in a new window, rather than using "target=\_blank" within the hyperlink tag, use:
 

```
<a href="javascript:void window.open('http://www.company.com/cgi-bin?article3');" class="PSHYPERLINK">
```


## Describing Branding Considerations

As mentioned earlier, this is a business-oriented portal. Although content and name recognition are important, they are secondary to the primary goal of enabling end users to perform their work in a more efficient manner. Thus, any branding should be subtle and never detract from the pagelet content or the rest of the portal.

Thus, to operate within the PeopleSoft Enterprise Portal product, a pagelet developed outside of PeopleSoft can credit the source developer, but must follow these standards:

- Branding on a homepage pagelet should be done using text only and should be placed at the bottom of the pagelet. No organization logos can be placed on the pagelet without permission from PeopleSoft. You can use "Provided by XXX," where XXX is substituted with your company name. Also, XXX can be a hyperlink to an appropriate web page.
- Organization logos and further information about your organization, products, and services should be located on the pagelet personalization page or other pagelet extensions.
- Generally, graphics for organization logos should not be larger than 100 pixels wide x 40 pixels high.

Let's examine a sample HTML pagelet. If we wanted to create one that searched for keywords on another website, it might look like the following page.



Sample Pagelet

Enter keyword:

Search

General directions can be provided here.  
Remember that pagelets should be as easy and  
intuitive to use as possible.

Provided by: [Content Source Company](#)

Sample pagelet

The HTML used is shown below:

```

<table border=0 cellspacing=1 cellpadding=0>
<tr>
<td><!-- start of search form -->
<form method="POST" action=
"http://www.contentSource.com/cgi-bin/dofind.cgi">
<center><font class=
"PSEDITBOXLABEL">Enter keyword:</font> <input name="word" value=""><br>
<input type="submit" value="Search"></center>
</form>
<!-- end of search form --></td>
</tr>

<tr>
<td><!-- start of optional directions --><font class=
"PAADDITIONALINSTRUCTIONS">General directions can be provided here.
Remember that pagelets should be as easy and intuitive to use
as possible.<br>
<br>
</font>
<!-- end of optional directions --></td>
</tr>

<tr>
<td><!-- start of footer for branding -->
<center><font class="PSPSMALLTEXT">Provided by <a href=
"http://www.contentSource.com/">Content Source Company</a></font></center>
<!-- end of footer for branding --></td>
</tr>
</table>

```

Let's examine how this adheres to the standards mentioned above.

- It uses a table to help wrap words, rather than let long strings dictate the width of the pagelet.
- All text should use the PeopleSoft styles, such as PSEDITBOXLABEL, PAADDITIONALINSTRUCTIONS, and PSPSMALLTEXT.
- The branding is small and no graphics are used. They adhere to the branding standards mentioned earlier.

---

## Understanding Single Signon and Pagelets

This section provides overviews of:

- Single signon.
- The PeopleSoft authentication process.
- The PeopleSoft authentication API.

## Describing Single Signon

The examples shown thus far have utilized publicly available URLs. Even if the examples represent third-party applications, the discussion has concentrated on retrieving data and rendering a pagelet. We have not yet discussed the possible need to log in to a non-PeopleSoft system.

When there are two or more systems that need separate authentication, it is preferable to handle it in some automated fashion. It is inconvenient and annoying if a user needs to manually log in to several different systems each day. Users often expect a business portal to be similar to accessing a variety of internet websites. Once logged into the portal, there should rarely (if ever) be the need to log in to another system.

There are several means to accomplishing single signon between PeopleSoft and other systems. First, you need to determine the primary (or master) and secondary (or slave) authentication systems.

- PeopleSoft as master.

Once a PeopleSoft user has logged in, an authentication cookie is sent to the browser's memory. Other applications can choose to authenticate using this cookie. PeopleSoft provides an API that other applications can leverage. This is the option that is discussed in detail in this section.

- PeopleSoft as slave.

PeopleSoft's authentication process is flexible enough to allow for accessing another system.

See *Enterprise PeopleTools 8.50 PeopleBook: Security Administration*, "Employing Signon PeopleCode and User Exits."

- PeopleSoft and other applications leverage third-party authentication.

A variant of the previous option would be that all applications (including PeopleSoft) leverage, or trust, a third-party authentication system such as Netegrity, Oblix, or Securant.

If you are writing a pagelet for the PeopleSoft portal, there is no guarantee that all possible customers for the pagelet would have access to a third-party authentication system. Thus, this option is not discussed in this document.

## Describing the PeopleSoft Authentication Process

Before discussing how your pagelet could leverage PeopleSoft authentication, it is important to understand the process.

After the first application server/node authenticates a user, PeopleSoft delivers a web browser cookie containing an authentication token. PeopleSoft Pure Internet Architecture uses web browser cookies to store a unique access token for each user after they are initially authenticated. When the user connects to another PeopleSoft application server/node, the second application server uses the token in the browser cookie to re-authenticate the user behind the scenes so that they don't have to complete the signon process again. Your non-PeopleSoft application could do something similar.

Single signon is critical for PeopleSoft portal implementations because the portal integrates content from various data sources and application servers and presents them in a unified interface. When users sign on through the portal, they always take advantage of single signon. Users need to sign on once and be able to navigate freely without encountering numerous signon screens.

---

**Note.** The browser cookie is an in-memory cookie and is never written to disk. The cookie is also encrypted and digitally signed using a checksum to prevent tampering.

---

The following table presents the fields that appear in the PeopleSoft authentication token.

<i><b>Field</b></i>	<i><b>Description</b></i>
UserID	Contains the user ID of the user to which the server issued the token. When the browser submits this token for single signon, this is the user that the application server logs on to the system.
Language Code	Specifies the language code of the user. When the system uses this token for single signon, it sets the language code for the session based on this value.
Date and Time Issued	<p>Specifies the date and time the token was first issued. The system uses this field to enforce a time-out interval for the single signon token. Any application server that accepts tokens for signon has a "time-out minutes" parameter configured at the system level.</p> <p>A system administrator sets this parameter using the Single Signon page. The value is in Greenwich Mean Time (GMT), so the application server's time zone is irrelevant.</p>
Issuing System	<p>Specifies the name of the system that issued the token. When it creates the token, the application server retrieves this value from the database. Specifically, it retrieves the defined Local Node. Single signon is not related to PeopleSoft Integration Broker messaging, except for the fact that single signon functionality leverages the messaging concept of nodes and local nodes.</p> <p>You configure a node to only trust single signon tokens from specific nodes. Consequently, an application server needs an Issuing System value so that it can check against its list of trusted nodes to see if it trusts the issued token.</p>
Signature	<p>Contains a digital signature that enables the application server using a token for single signon to ensure that the token hasn't been tampered with after it was originally issued. The system issuing the token generates the signature by concatenating the contents of the token (all of the fields that appear in this table) with the message node password for the local node. The system then hashes the resulting string using the SHA1 hash algorithm.</p> <p>For example ("+" indicates concatenation).</p> <pre>signature = SHA1_Hash ( UserID + Lang + Date Time issued + Issuing System + Local Node Pswd )</pre> <p>There is only one way to derive the 160 bits of data that make up the signature, and this is by hashing exactly the same User ID, Language, Date Time, Issuing System, and node password.</p> <p><b>Note.</b> If you are using digital certificate authentication, the signature of the digital certificate occupies this space. The above description only applies to using password authentication.</p>

---

**Note.** Single signon does not depend on the use of a Lightweight Directory Access Protocol (LDAP) directory. You can store user credentials in an LDAP directory if desired, but it is not required.

---

The key security features of the cookie authentication token are:

- The cookie exists in memory; it is not written to disk.
- There is no password stored in the cookie.
- You can set the expiration of the cookie to be a matter of minutes or hours. This expiration option is a useful security feature.

## Describing the PeopleSoft Authentication API

PeopleSoft delivers a component interface named PRTL\_SS\_CI, which enables external applications to seamlessly integrate a single signon solution with the PeopleSoft portal applications. This component interface helps ensure that users who have already signed on to the portal don't have to sign on again for every system you reference in your portal.

Component interfaces are the focal points for externalizing access to existing PeopleSoft components. They provide real-time synchronous access to the PeopleSoft business rules and data associated with a component outside the PeopleSoft online system. Component interfaces can be viewed as "black boxes" that encapsulate PeopleSoft data and business processes, and hide the details of the structure and implementation of the underlying page and data.

To take advantage of the Single Signon API, you need to create a custom API, which includes building the dynamic link libraries, classes, and registry settings necessary to enable an external application to communicate with PeopleSoft. This can be done automatically through PeopleTools. More information about building dynamic link libraries, classes, and registry settings, as well as other details about PeopleSoft component interfaces can be found in the *PeopleTools 8.50 PeopleBook: PeopleSoft Component Interfaces*

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Component Interfaces*.

Only external applications, such as COM, C/C++, or Java programs require a component interface API. PeopleCode programs do not require a component interface API, and in fact, we do not recommend building a component interface API if the component interface is to be accessed from PeopleCode only.

The files of your custom API need to reside on the client machine; that is, the web server for ASP and the machine running the Java program for Java. The registry file may also need to be executed to update the registry with the new libraries.

### **The Signon Process with the API**

The PRTL\_SS\_CI component interface contains two user-defined methods:

- `Authenticate()`.

Your external authentication program distributes an authentication token that can be retrieved from a cookie in the browser. The `Authenticate` function determines if an authentication token is valid.

- `GetUserID()`.

If the token is valid, you use the `GetUserID` function to retrieve the user ID associated with the authentication token.

Before we describe the development requirements of your API, take a moment to examine the steps that occur internally when you use the API in conjunction with the delivered PRTL\_SS\_CI:

1. The user enters the user ID and password into the PeopleSoft portal signon page.
2. If the login to the portal application server is successful, the server generates a single signon token. The web server receives the single signon token from the application server and issues a cookie to the browser.
3. The user navigates in the portal and encounters a hyperlink to the external system. The user clicks the link.
4. The browser passes the PS\_TOKEN cookie to the external web server.
5. The external web server checks for the PS\_TOKEN cookie before displaying a signon page.
6. Once it is determined that the user is accessing the application through the PeopleSoft portal, you retrieve the authentication token and send it to the PRTL\_SS\_CI component interface to verify authentication. For instance, it calls `PRTL_SS_CI.Authenticate(Auth. token string)`.
7. After the system authenticates the token, it can then make calls to the `PRTL_SS_CI.Get_UserID()` function to return the appropriate user ID.

In general, cookies are not transferable across domains. The only domain that can access the cookie is the domain that created it. Therefore, the web server for the non-PeopleSoft system must be on the same domain as the PeopleSoft system so that the cookies are passed appropriately.

### ***External Application Support for Single Signon***

Developers of external applications need to alter the signon process to conform to the following requirements:

1. Check for the PS\_TOKEN cookie. If the cookie doesn't exist, continue with the normal signon process. Otherwise, bypass the signon screen.
2. Retrieve the authentication token from the PS\_TOKEN cookie.
3. Make a connection to PeopleSoft through the PRTL\_SS\_CI API.
4. Pass the authentication token to the `Authenticate()` function of the API.
5. If `Authenticate()` returns *True*, retrieve the user ID associated with the authentication token using the `Get_UserID()` function.

### ***Authentication API PeopleCode Example***

The following PeopleCode example walks through the process of validating your authentication token and retrieving the user ID. The following sample is designed to provide a general idea of the process involved and help you to incorporate the PRTL\_SS\_CI API into your signon process.

```

Local ApiObject &THISSESSION;
Local ApiObject &THISCI;
Local string &AUTHTKN;
/* Assigns the Authentication Token to a variable */
&AUTHTKN = %AuthenticationToken;
/* Open a session and make a connection */
&THISSESSION = GetSession();
If &THISSESSION.connect(1, "EXISTING", "", "", 0) <> True Then
WinMessage(MsgGet(30000, 1, "Session Connect Failed.));
Exit (1);
End-If;
/* Retrieves the component interface PRTL_SS_CI */
&THISCI = &THISSESSION.GetCompIntfc(CompIntfc.PRTL_SS_CI);
/* Checks to see if the component interface is NULL */
If &THISCI = Null Then
WinMessage("Component Interface PRTL_SS_CI not found.
Please ensure Component Interface Security access is granted
to this user.");
Exit (1);
End-If;
/* Key fields would usually be set before the Get() function is
called in order to map the component interface to a particular
set of data. This component interface is not mapped to data.
* Therefore, the component interface is retrieved and then the
user defined methods are retrieved */
&THISCI.get();
PRTL_AUTH = &THISCI.Authenticate(&AUTHTKN);
PRTL_USER_ID = &THISCI.Get_UserID();

```

---

**Note.** The component interface is not mapped to data because the key field for the data would be the authentication token. This token is dynamically assigned when the user signs on to the portal, and it is not stored as data anywhere in the system. Therefore, there are no key fields and the token is passed directly to the user-defined functions.

---

### ***Authentication API Java Example***

Here is an example of a similar operation written in Java. This is a file named SingleSignon.java.

```

package examples.migration.sso;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import psft.pt8.joa.*;
import PeopleSoft.Generated.CompIntfrc.*;

public class SingleSignon extends HttpServlet {
    public static ISession oSession;
    String psfttoken;
    public static void ErrorHandler() {
        //***** Display PeopleSoft Error Messages *****
        if (oSession.getErrorPending() || oSession.getWarningPending()) {
            IPSMessageCollection oPSMessageCollection;
            IPSMessage oPSMessage;

            oPSMessageCollection = oSession.getPSMessages();
            for (int i = 0; i < oPSMessageCollection.getCount(); i++) {
                oPSMessage = oPSMessageCollection.item(i);
                if (oPSMessage != null)
                    System.out.println(
                        "(" + oPSMessage.getMessageSetNumber() + ", "
                        + oPSMessage.getMessageSetNumber() + ") : " + oPSMessage.getText());
            }
            //***** Done processing messages in the collection;
            OK to delete *****
            oPSMessageCollection.deleteAll();
        }
    }

    public void doGet(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException {
        try {
            response.setContentType("text/html");
            PrintWriter out = response.getWriter();

            Cookie[] cookies = request.getCookies();
            if (cookies == null) {
                out.println("<TR><TH COLSPAN=2>No cookies");
            } else {
                Cookie cookie;
                for(int i=0; i<cookies.length; i++) {
                    cookie = cookies[i];

                    String pstoken = cookie.getName();
                    psfttoken = cookie.getValue();
                    if (pstoken.equals ("PS_TOKEN"))

                        out.println("<TR>\n" +
                            "    <TD>" + pstoken + "\n" +
                            "    <TD>" + psfttoken);
                }
            }
        }

        String strServerName, strServerPort, strAppServerPath;
        String strUserID, strPassword;
        strServerName = "jfinnon09";
        strServerPort = "9500";
        strUserID = "VP1";
        strPassword = "VP1";
    }
}

```

```

//Build Application Server Path
strAppServerPath = strServerName + ":" + strServerPort;

    //***** Create PeopleSoft Session Object *****
    //      ISession oSession;

oSession = API.createSession();

//***** Connect to the App Server *****
if (!oSession.connect(1, strAppServerPath, strUserID, strPassword,
null)) {
    out.println("\nUnable for Jason to Connect to
Application Server.");
    ErrorHandler();
    return;
}

    //***** Get Component Interface *****
IPrtlSsCi oPrtlSsCi;
String ciName;
    ciName = "PRTL_SS_CI";
oPrtlSsCi = (IPrtlSsCi) oSession.getCompIntfc(ciName);
if (oPrtlSsCi == null) {
    out.println("\nUnable to Get Component Interface " + ciName);
    ErrorHandler();
    return;
}

//***** Set the Component Interface Mode *****
oPrtlSsCi.setInteractiveMode(false);
oPrtlSsCi.setGetHistoryItems(true);
oPrtlSsCi.setEditHistoryItems(false);

//***** Set Component Interface Get/Create Keys *****

//***** Execute Get *****
if (!oPrtlSsCi.get()) {
    out.println("\nNo rows exist for the specified keys.
\nFailed to get the Component Interface.");
    ErrorHandler();
    return;
}

//***** BEGIN: Set/Get Component Interface Properties *****

//***** Set Level 0 Properties *****
//out.println("oPrtlSsCi.PrtlToken: " +
oPrtlSsCi.getPrtlToken());
//out.println("Checking token: " + psfttoken);

//oPrtlSsCi.setPrtlToken(<*>);
//String psfttoken;
//psfttoken = oPrtlSsCi.getPrtlToken();
//System.out.println("oPrtlSsCi.PrtlToken: "
+ psfttoken);

//***** END: Set Component Interface Properties *****

//***** Execute Standard and Custom Methods *****
//***** Execute Authenticate *****
boolean auth;

    auth = oPrtlSsCi.authenticate(psfttoken);
    //out.println("Auth: " + auth);

```

```

        //Execute Get_UserID
        String psftuser;
        psftuser = oPrtlSsCi.getUserid();
        //out.println("Psftuserid: " + psftuser);

        String title = "Welcome " + psftuser;

        //***** Get HTML *****
        out.println(ServletUtilities.headWithTitle(title));
        out.println("</BODY></HTML>");

        //***** Disconnect from the App Server *****
        oSession.disconnect();
        return;
    }

    catch (Exception e) {
        e.printStackTrace();
        System.out.println("An error occurred: ");
        ErrorHandler();
    }
}

/** Let the same servlet handle both GET and POST. */

public void doPost(HttpServletRequest request,
                   HttpServletResponse response)
    throws ServletException, IOException {
    doGet(request, response);
}
}

```

---

## Developing Pagelets Using PeopleSoft Pure Internet Architecture

This section discusses how develop pagelets based on PeopleSoft Pure Internet Architecture pages.

### Developing Pagelets Based on PeopleSoft Pure Internet Architecture Pages

To design a PIA page-based pagelet:

1. Go to *Start, Programs, PeopleTools 8.4x, Application Designer*.
2. Select *File, Open* to display the Open Definition dialog box.
3. Use the drop-down list box to select *Page* as the definition.
4. Enter selection criteria and open the desired page.
5. Select *File, Definition Properties* and then select the *Use* tab.
6. Select a page size of *240xVar portal home page comp* for narrow pagelets or *490xVar portal home page comp* for wide pagelets. Note that you can adjust the height of the pagelet now or later.

7. Design the new pagelet using the guidelines described above and similar design techniques used when designing a page.
8. Save the page.

---

**Note.** When the page is added to a component, the search record is specified. As noted in the above guidelines, you should avoid a search interaction so that the pagelet can render its data on the homepage without any prompting for keys. If keys are needed, then the pagelet should be coded to use some default key values and the personalization options should initially reflect these defaults.

---

9. Register the pagelet in the portal registry. This step also involves setting up security access to the component and therefore the PIA page.

See [Chapter 12, "Building Pagelets," Administering Pagelets in the Portal Registry, page 442.](#)

---

## Developing Pagelets Using iScripts

This section discusses how to develop pagelets based on iScripts.

### Developing Pagelets Based on iScripts

Use an iScript only for pagelets that reference non-PeopleSoft data or if a PIA page does not provide the necessary functionality.

The advantages of iScripts include the following:

- People may be familiar with the technique, since it's similar to creating Active Server Pages (ASP) or Java Server Pages (JSP). Basically, you are using a script to output HTML and JavaScript.
- There is some greater flexibility in the appearance since there is direct control of the HTML or JavaScript.
- There is some greater flexibility of sizing. PeopleSoft Internet Architecture pages tend to be fairly static in their object placement and sizing.

However, the disadvantages of iScripts include the following:

- Everything is manually coded. You must handle and account for support for multiple browsers and international considerations, such as language, date and currency formatting, and so forth. PeopleTools handles these considerations automatically on PIA pages.
- There may be related coding or performance issues.

This section provides an example of developing an iScript-based pagelet that uses content from an external site for display in the PeopleSoft portal. The pagelet is based on a public-use form that is provided at [www.onelook.com](http://www.onelook.com).

When developing a template pagelet based on an iScript, some of the functions provided by PeopleSoft Application Designer such as currency codes, language support, and multiple browser support are not automatically available in your iScript. Therefore, if they are desired, they will need to be developed.

---

**Note.** When using iScripts to create pagelets, use the styles (classes) that are defined in PeopleSoft Application Designer style sheets to specify the attributes of the objects referenced.

---

To create an iScript-based template pagelet:

---

**Note.** Use steps 1 and 2 when you are grabbing HTML from another Website, or if you basically want to create some relatively static HTML that is used by the iScript to render a pagelet. If you are creating a pagelet iScript that is going to render highly dynamic HTML, then skip to step 3 and create any necessary/complex PeopleCode to conditionally render the HTML/JavaScript.

---

1. Create the HTML code for the pagelet.

In many cases you can navigate to a Uniform Resource Locator (URL) that you want to turn into a pagelet and copy the HTML associated with that Web page. For form-based Web pages, copy all of the HTML code located between the Form tags of an existing HTML document.

2. Store the new HTML code in PeopleSoft Application Designer as an HTML definition.
3. Create an iScript that calls the new HTML definition in a Web library.

A Web library is a derived record that is created in PeopleTools for storing iScripts. The name of all Web library records begin with `WEBLIB_`.

---

**Note.** All iScripts should be coded in the FieldFormula PeopleCode event of a derived record called `WEBLIB_XX`, where `XX` is the product code. Only functions that start with *iScript\_*, such as *iScript\_iTracker*, can be invoked from a URL.

---

4. Navigate to the appropriate field name and edit the FieldFormula PeopleCode that is associated with the field.

This is where you create an iScript that calls the HTML definition that you created.

5. Create a new iScript and give it a meaningful name.

An easy way to create a new iScript is to copy and paste an existing iScript that is located in the Web library and then make the desired changes. All iScript names must begin with *iScript\_*.

6. Save the Web library.
7. Register the iScript as a pagelet in the portal registry. This step also involves setting up security access to the iScript.

See [Chapter 12, "Building Pagelets," Administering Pagelets in the Portal Registry, page 442.](#)

### **Modifying Security For the Web Library**

Security for the web library can be initially set up using the Registration Wizard. You can use the following steps to modify the security settings created by the wizard.

To modify security:

1. Select *PeopleTools, Security, Permissions & Roles, Permission Lists*.
2. Open the desired permission list, such as *ALLPAGES* or *PAPP9000*.

3. Go to the *Web Libraries* tab.
4. Select the *Edit* link next to the appropriate web library.
5. Modify the Weblib Permissions as desired by changing the Access Permissions drop-down option. The choices are Full Access and No Access. Click OK to return to the main Permission List page.
6. Click Save to save your changes to the permission list.

### **See Also**

*Enterprise PeopleTools 8.50 PeopleBook: Security Administration*, "Setting Up Permission Lists," Setting Web Library Permissions

*Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Application Designer Developer's Guide*, "Using the Registration Wizard," Understanding the Registration Wizard

*Enterprise PeopleTools 8.50 PeopleBook: PeopleCode API Reference*, "Internet Script Classes (iScript)," iScript Security

---

## **Developing Pagelets Using Java**

This section provides an overview of Java pagelet developing and Java pagelet development considerations and discusses how to:

- Write business logic in Java.
- Use Java to write business logic for a PeopleSoft or non-PeopleSoft system.
- Invoke PeopleSoft components from Java.

## **Understanding Developing Pagelets in Java**

Java is an extremely popular programming language today, especially for internet applications. PeopleTools provides support in several key areas to enable developers to create Java-based applications. Though the PeopleSoft development environment allows you to write PeopleSoft business logic in Java, the focus of the following sections is to provide information that is relevant to the developer who is integrating applications with the PeopleSoft Enterprise Portal. For example, development that is being done to integrate your non-PeopleSoft application with the PeopleSoft Enterprise Portal. Another example is a scenario in which you want to pull together data from your non-PeopleSoft system along with data from your PeopleSoft system for integration with the PeopleSoft Enterprise Portal.

There are two primary areas in which you can program in Java within the PeopleTools environment:

### 1. Writing business logic in Java.

Java can be used to write business logic for PeopleSoft or non-PeopleSoft applications that are to be rendered in the PeopleSoft Enterprise Portal. Java can be called from PeopleCode and the Java program can then reference PeopleSoft's run-time environment as needed.

In addition, Pagelet Wizard facilitates Java development in that it allows the Java programmer to concentrate on business logic, while Pagelet Wizard takes care of all the development aspects of rendering the application in the portal framework without any coding necessary.

### 2. Invoking PeopleSoft components from Java.

All PeopleSoft components can be invoked from Java programs through the component interface technology. This is useful if you want to create Java servlets, applets, JSP programs, or Enterprise JavaBeans that invoke PeopleSoft business logic.

Another consideration when developing applications with Java is that any application developed using Java that is URL-addressable and produces HTML can be easily integrated into the PeopleSoft Enterprise Portal through the PeopleSoft Enterprise Portal's registry functionality. This allows you to distribute your development and run-time environments making for better scalability for both.

For more information about leveraging the PeopleSoft Enterprise Portal's registry functionality to integrate your Java application with the PeopleSoft Enterprise Portal, see the following documentation.

See [Chapter 12, "Building Pagelets," Administering Pagelets in the Portal Registry, page 442.](#)

These areas of Java programming within the PeopleTools development environment are discussed further in the following sections.

## Describing Java Pagelet Development Considerations

This section discusses the following Java development considerations:

- Business rules.
- User interface.
- User personalization.
- Navigation.

### ***Business Rules***

Java can be used to apply business logic to data to create a result set. Pagelet Wizard can then be used to invoke the Java program. Parameters that are required by the Java program can be defined, managed, and passed to the Java program through Pagelet Wizard. These parameters may be derived from user personalization parameters, as well as administrator-defined parameters and system variables.

For example, the Java program may need the user's ID, the user's language, and the user's personalization selection regarding whether to include historical transaction information or just current information. These parameters can be defined in Pagelet Wizard in relation to a Java class. Pagelet Wizard can access the PeopleSoft run-time environment to determine the user's ID as well as other profile information, such as the user's language.

See [Chapter 11, "Using Pagelet Wizard," Understanding System Variables Supported as Data Source Parameters, page 325.](#)

Pagelet Wizard will also manage the user interface and storage of user personalization data. This is discussed in detail in the following "User Personalization" subtopic. When the Java program is invoked, Pagelet Wizard will handle passing all required parameters to the Java program.

In addition, Java classes delivered with PeopleTools enable you to call PeopleCode from your Java program and access contextual information from the run-time system. If needed, business data can be retrieved, as necessary, through the numerous PeopleCode functions that are accessible from Java.

The result set created by the Java program may be in an HTML format or in an XML format. Pagelet Wizard will accept the output from the Java program and will manage the presentation of the information within the portal. This is discussed in the following "User Interface" subtopic.

### ***User Interface***

Pagelet Wizard will manage all aspects of a pagelet's user interface. No user-interface-related programming for the pagelet is required in the Java program. Pagelet Wizard can accept an HTML- or XML-based result set from a Java program. The HTML from the Java program can be rendered directly. Alternatively, XML can be transformed with XSL to create an appropriate pagelet user interface.

In either case, Pagelet Wizard will manage the XSL and invoke the transformation at runtime. Extensible Stylesheet Language templates are provided with Pagelet Wizard for general use scenarios. Extensible Stylesheet Language development may be required, depending on the specific user interface required for a pagelet.

See [Chapter 11, "Using Pagelet Wizard," Defining Pagelet Wizard XSL Prototypes, page 300.](#)

### ***User Personalization***

Pagelet Wizard supports the persistence of user personalization selections for a pagelet application and handles all aspects of the user interface that allows a user to enter their preferences and selections. Personalization parameters are defined within Pagelet Wizard's administration pages. No programming is required, Java or otherwise.

User personalization selections can occur at two levels. From the homepage, users can select which pagelet applications they'd like to include on their homepage. After they've selected a pagelet application and have decided where they'd like the pagelet to display on their homepage, they can personalize a particular pagelet application.

Examples of user personalizations at the second level can include scenarios in which a user selects transactions from a particular region, or chooses to include transactions using a particular date as opposed to same-day transactions.

The user interface for all of these activities, as well as the storage of the personalization data for persistence, is managed by Pagelet Wizard without any additional programming, Java or otherwise.

Pagelet Wizard supports functionality that allows an end user to select a pagelet application to appear on their homepage. When a pagelet application is registered with Pagelet Wizard, a folder, which represents a portal application category, can be selected. When users personalize their homepage by selecting specific pagelet applications that they want to appear on their homepage, the selection will be presented within the category (folder) that is defined when registering the pagelet application through Pagelet Wizard.

See [Chapter 11, "Using Pagelet Wizard," Step 6: Specifying Pagelet Publication Options, page 365.](#)

Pagelet Wizard also supports functionality that allows an end user to select values that allow a specific pagelet application to be personalized. The functionality may dictate that some input parameters required by a Java program be entered by an end user, while other parameters can be set by an administrator. Pagelet Wizard manages parameter passing and the user interface. Pagelet Wizard introspects the Java source code for input parameters. The Pagelet Wizard administrator can then select which parameters can be accessed by an end user for personalization. The user interface to allow an end user to personalize a pagelet application and update the appropriate Java input parameters is automatically generated and managed by Pagelet Wizard.

## **Navigation**

When creating links in your pagelet application that allow a user to navigate to a related page for details and so forth, it may be important that the link be proxied by the portal to keep the user within the portal. For example, if a pagelet link is not proxied, then the pagelet can take a user to a site that is completely independent from the portal. In this scenario, the navigational links on the portal header, the homepage menu pagelet, and the left-hand menu pagelet will not be available to the user.

If the link is proxied by the portal, then the link can bring in content from a site that is independent of the portal. However, the content will be rendered within the portal. For example, the portal's header and left-hand menu will be available.

With the PeopleSoft Enterprise Portal, links that are included in a pagelet will automatically be altered to support proxying by the portal. No programming is required, Java or otherwise.

---

**Note.** Content that is referenced by the link must be HTML-based to be rendered appropriately in the portal.

---

## **Navigation on Extended Pages**

Invoking a link on an extended page will bring in content within the target frame. The PeopleSoft Enterprise Portal, header, and left-hand menu area will remain. Therefore, the new content will be rendered in the context of the PeopleSoft Enterprise Portal.

---

**Note.** Content that is referenced by the link must be HTML-based to be rendered appropriately in the portal.

---

The Return to Home link, which returns the user to the portal homepage, may be required from an extended page. The `GenerateHomepagePortalURL` function creates a URL string that represents an absolute reference to the portal homepage. Because you can access the PeopleSoft runtime environment from a Java program and you have access to PeopleCode built-in functions, you can invoke the `GenerateHomepagePortalURL` function from your Java program.

See [Chapter 12, "Building Pagelets," Developing Pagelets Using Java, page 428](#).

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleCode Language Reference*, "PeopleCode Built-in Functions."

## **Portal Integration**

The pagelet application needs to be integrated with the PeopleSoft Enterprise Portal so that the pagelet application is managed, secured, and presented through the PeopleSoft Enterprise Portal. Registration in the PeopleSoft Enterprise Portal makes the portal aware of a pagelet application. Information included in registration determines the name and security for your pagelet application. In addition, registration data determines the category in which the pagelet application resides. Users view pagelet applications within these categories when selecting items that they want to see on their homepage. No programming is required, Java or otherwise, to complete portal registration.

## Writing Business Logic in Java

There are many reasons why you would want to write business logic for your non-PeopleSoft (and even your PeopleSoft applications) in Java. Perhaps you have licensed a third-party set of Java classes to do some very specific processing (tax calculation, for example). Or perhaps you have developed some internal Enterprise JavaBeans for your specific business processes. Or perhaps you simply like to code in Java. Whatever the reason, you can easily integrate Java code with your applications through the Java PeopleCode functions.

### Invoking Java from PeopleCode

There are three primary Java PeopleCode functions that are used to invoke Java from PeopleCode:

- `GetJavaClass`
- `CreateJavaObject`
- `CreateJavaArray`

Java PeopleCode Function	Usage	Example
GetJavaClass	Finds a Java class you can manipulate in PeopleCode. This is used for those classes that have static members, where it isn't appropriate to instantiate an object of the class. You can only call static methods, that is, class methods, with the object created with this function.	<p>In Java, you access such static members of a class by using the class name:</p> <pre>result = java.class.name.SomeStaticMethod();</pre> <p>To do this in PeopleCode, do the following:</p> <pre>&amp;Result = GetJavaClass("java.class.name").SomeStaticMethod();</pre> <p>Following is a simple PeopleCode example that uses <code>GetJavaClass</code> to get a system class:</p> <pre>&amp;Sys = GetJavaClass("java.lang.System"); &amp;Sys.setProperty("java.security.policy",     "C:\java\policy"); WinMessage("The security property is: "   &amp;Sys.getProperty("java.security.policy")); &amp;Props = &amp;Sys.getProperties(); &amp;Props.put("java.security.policy",     "C:\java\policy"); &amp;Sys.setProperties(&amp;Props); WinMessage("The security property is: "   &amp;Sys.getProperty("java.security.policy"));</pre>

<b>Java PeopleCode Function</b>	<b>Usage</b>	<b>Example</b>
CreateJavaObject	<p>Creates a Java object that can be manipulated in your PeopleCode. You can use the CreateJavaObject function to create a Java array. If ClassName is the name of an array class (ending with [ ]), ConstructorParams are used to initialize the array.</p>	<p>In Java, you would do the following to initialize an array:</p> <pre>intArray = new int[]{1, 2, 3, 5, 8, 13};</pre> <p>Do the following to initialize a Java array from PeopleCode:</p> <pre>&amp;IntArray = CreateJavaObject("int[]", 1, 2, 3, 5, 8, 13);</pre> <p>If you want to initialize a Java array without knowing the number of parameters until runtime, use the CreateJavaArray function.</p> <p>Following is a simple PeopleCode program that creates a Java object from a sample program that generates a random password:</p> <pre>/* Example to return Random Passwords from a Java class */ Local JavaObject &amp;oGpw; /* Create an instance of the object */ &amp;oGpw = CreateJavaObject ("com.PeopleSoft.Random.Gpw_Demo"); &amp;Q = "1"; /* Call the method within the class */ &amp;NEW_VALUE = &amp;oGpw.getNewPassword(&amp;Q, PSRNDMPSWD.LENGTH); /* This is just returning one value for now */ PSRNDMPSWD.PSWD = &amp;NEW_VALUE;</pre>
CreateJavaArray	<p>Allows you to create a Java array without knowing the number or value of the elements.</p> <p>When you create an array in Java, you already know the number of elements in the array. If you don't know the number of elements in the array, but you want to use a Java array, use the CreateJavaArray function in PeopleCode. This will create a Java object that is a Java array, and you can pass in the number of elements that are to be in the array.</p>	<p>The following PeopleCode example passes a PeopleCode array of strings (&amp;Parms) to a Java method xyz of class Abc. This example assumes that when you are writing the code, you don't know how many parameters you will have.</p> <pre>Local JavaObject &amp;Abc, &amp;RefArray; Local array of String &amp;Parms; &amp;Parms = CreateArray(); /* Populate array how ever you want to populate it */ &amp;Abc = GetJavaObject("com.peoplesoft.def.Abc"); /* Create the java array object. */ &amp;JavaParms = CreateJavaArray("java.lang.String[]", &amp;Parms.Len); /* Populate the java array from the PeopleCode array. */ &amp;RefArray = GetJavaClass("java.lang.reflect. Array"); For &amp;I = 1 to &amp;Parms.Len &amp;RefArray.set(&amp;JavaParms, &amp;I - 1, &amp;Parms[&amp;I]); End-For; /* Call the method. */ &amp;Abc.xyz(&amp;JavaParms);</pre>

## Accessing the PeopleSoft Run-Time System From Java

After a Java class has been invoked, the PeopleSoft run-time system can then be accessed from Java. Java classes delivered with PeopleTools enable you to call PeopleCode from your Java program and access contextual information from the run-time system such as the current user's role and language preference. By importing the PeopleTools-delivered Java classes in your Java program, you can have access to PeopleCode objects and methods. There are hundreds of PeopleSoft system variables, constants, and built-in functions available for use with this approach. Discussions of the various methods follow.

Accessing the run-time system works only from a Java program that was initially called from PeopleCode. You must call PeopleCode facilities only from the same thread that was used for the call into Java. You cannot call any PeopleCode facility that would cause the server to return to the browser for an end-user action because the state of the Java computation cannot be saved and restored when the action is complete.

Java Class	Usage
SysVar	<p>Use to refer to PeopleSoft system variables, such as %Language or %Oprid.</p> <p>For example, %Session, becomes SysVar.Session().</p> <p>See <i>Enterprise PeopleTools 8.50 PeopleBook: PeopleCode Developer's Guide</i>, "Understanding the PeopleCode Language," System Variables.</p>
SysCon	<p>Use to refer to system constants, such as %SQLStatus_OK or %FilePath_Absolute.</p> <p>For example, %CharType_Matched becomes SysCon.CharType_Matched.</p> <p>See <i>Enterprise PeopleTools 8.50 PeopleBook: PeopleCode Developer's Guide</i>, "Understanding the PeopleCode Language," Constants.</p>
Name	<p>Allows you to use the PeopleSoft-reserved item references. This allows you to reference pages, components, records, fields, and so forth.</p> <p>For example, in PeopleCode you can refer to a record field using the following:</p> <pre>recname.fieldname</pre> <p>With the Name class, you can use a similar construct:</p> <pre>new PeopleSoft.PeopleCode.Name( "RECNAME" , "FIELDNAME" );</pre> <p>As another example, in PeopleCode you can refer to a PeopleSoft page using the following:</p> <pre>PAGE.pagename</pre> <p>In Java, it would be:</p> <pre>new PeopleSoft.PeopleCode.Name( "PAGE" , "PAGENAME" );</pre> <p>See <i>Enterprise PeopleTools 8.50 PeopleBook: PeopleCode Developer's Guide</i>, "Understanding the PeopleCode Language," Definition Name References.</p>
Func	<p>Use to refer to built-in PeopleCode functions, such as CreateRowset or GetFile.</p> <p>For example, SetLanguage(LANG_CD) becomes Func.SetLanguage(LANG_CD).</p> <p>The existing PeopleCode classes (Array, Rowset, and so forth) have properties and methods you can access from Java. PeopleCode classes have the same name, so Record becomes Record, SQL becomes SQL, and so forth. Methods are accessed by the method name. The name of a property is pre-pended with either "get" or "set," depending on whether you're reading or writing to the property. For example, to get the IsChanged property would be getIsChanged. To set the value for a field would be &amp;MyField.setValue.</p>

Java Class	Usage
Func (continued)	<p>Here is an example of a Java program that uses PeopleCode objects to access the database:</p> <pre> /* This code is used to test the Java/PeopleCode interface. */ import PeopleSoft.PeopleCode.*; public class Test { /* * Add up and return the length of all the * item labels on the UTILITIES menu, * found two different ways. */ public static int Test() { /* Get a Rowset to hold all the menu item records. */ Rowset rs = Func.CreateRowset(new Name("RECORD", "PSMENUITEM"), =&gt; new Object[]{}); String menuName = "UTILITIES"; int nRecs = rs.Fill(new Object[]{"WHERE FILL.MENUNAME = :1", =&gt; menuName}); int i; int nFillChars = 0; for (i = 1; i &lt;= rs.getActiveRowCount(); i++) { String itemLabel = (String)rs.GetRow(i) .GetRecord(new Name("RECORD", "PSMENUITEM")) .GetField(new Name("FIELD", "ITEMLABEL")) .getValue(); nFillChars += itemLabel.length(); } /* Do this a different way - use the SQL object to read each=&gt; menu item record. */ int nSQLChars = 0; Record menuRec = Func.CreateRecord(new Name("RECORD", =&gt; "PSMENUITEM")); SQL menuSQL = Func.CreateSQL("%SelectAll(:1) WHERE MENUNAME = :&gt; 2", new Object[]{menuRec, menuName}); while (menuSQL.Fetch(new Object[]{menuRec})) { String itemLabel = (String)menuRec .GetField(new Name("FIELD", "ITEMLABEL")) .getValue(); nSQLChars += itemLabel.length(); } return nFillChars + 100000 * nSQLChars; } } </pre>
Func (continued)	<p>This can be run from PeopleCode as follows:</p> <pre> Local JavaObject &amp;Test; Local number &amp;chars; &amp;Test = GetJavaClass("Test"); &amp;chars = &amp;Test.Test(); &amp;Test = Null; WinMessage("The character counts found are: "   &amp;chars, 0); </pre> <p>See <i>Enterprise PeopleTools 8.50 PeopleBook: PeopleCode API Reference</i>, "Quick Reference for PeopleCode Classes."</p>

## Using Java to Write Business Logic for a PeopleSoft or Non-PeopleSoft System

This section presents an example of pagelet application development where it is desired that all programming be done in Java. Business logic might be for a PeopleSoft system or a non-PeopleSoft system. In this development example, PeopleTools programming is not required. All programming is done with Java.

The PeopleTools infrastructure can be leveraged without any PeopleTools programming through the use of PeopleSoft Enterprise Portal features that support administration, integration and implementation for the portal. The ability to leverage the PeopleTools infrastructure can significantly reduce the amount of Java programming that is required to develop a pagelet application for use in the PeopleSoft Enterprise Portal.

Some of the information in this section is covered in other Pagelets Wizard documentation. The main difference in this section is that it presents the information as a development scenario as opposed to a feature description. Also, the ability to access the PeopleSoft run-time environment from Java is discussed in this section.

The development scenario below will address the following elements of portal application development, integration, and implementation:

- Business rules.

Apply business logic/rules to data in order to create a result set.

- User interface.

Create the appropriate presentation format for a set of business data.

- User personalization.

- User personalization data.

Allow pagelet applications to be selected by the end user in order to be placed on their home page. Also, allow a user to personalize a specific pagelet application. These selections should persist.

- User interface.

Create the appropriate presentation format for personalization data.

- Navigation for a pagelet.

Create links on your pagelet to allow a user to navigate to a related page for details and so forth.

- Navigation on extended pages.

An extended page is a page that a user can get to from a homepage pagelet. It might show details as well. An extended page may have links to allow a user to navigate to other related pages.

- Links.

Create links from your extended page to allow a user to navigate to a related page for further details and so forth.

- Return links.

Create links that allow the user to return to the portal's homepage.

- Portal integration and implementation.

Integrate the pagelet application with the PeopleSoft Enterprise Portal so that the pagelet application is managed, secured, and presented through the PeopleSoft Enterprise Portal.

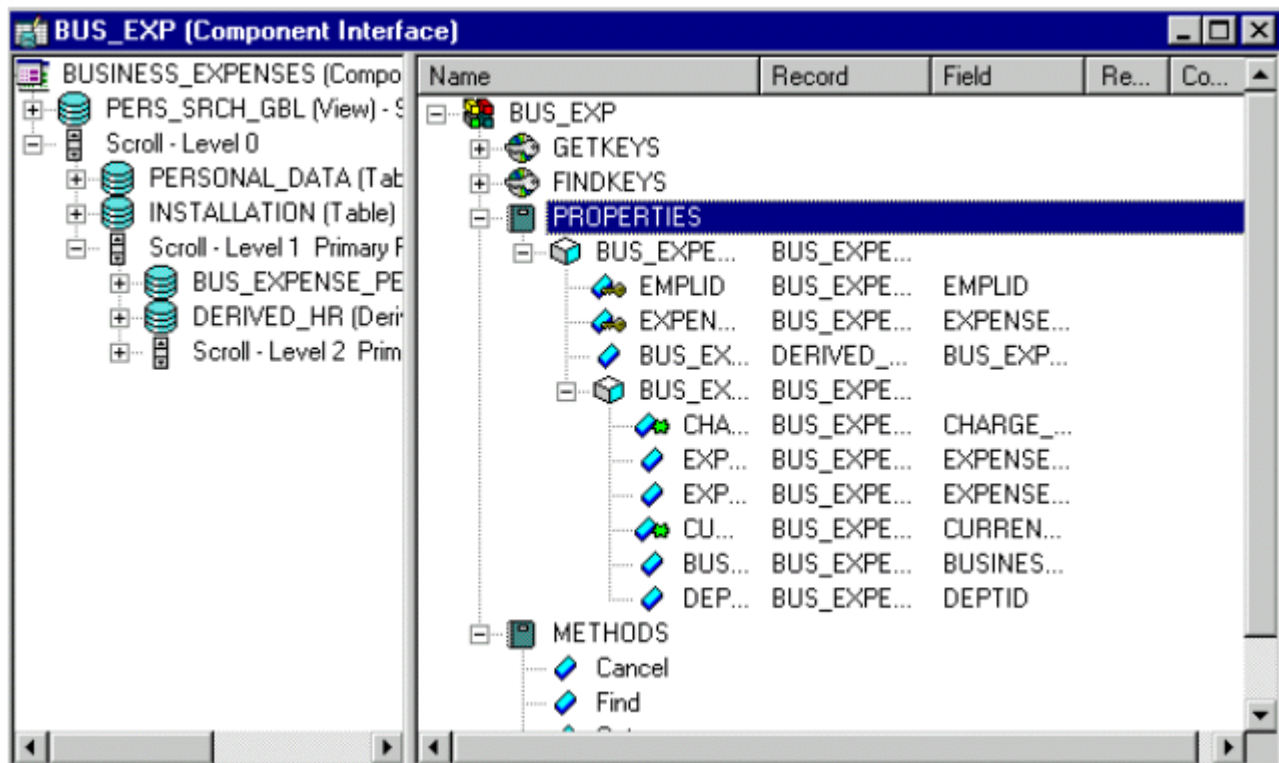
## Invoking PeopleSoft Components from Java

All PeopleSoft components can be invoked from Java programs through component interface technology. This is useful for those developers who want to create Java servlets, applets, JSP programs, or Enterprise JavaBeans that invoke PeopleSoft business logic. This section walks through an example of how to invoke a PeopleSoft component from Java. A Business Expense component is used as the example.

### Creating the Component Interface

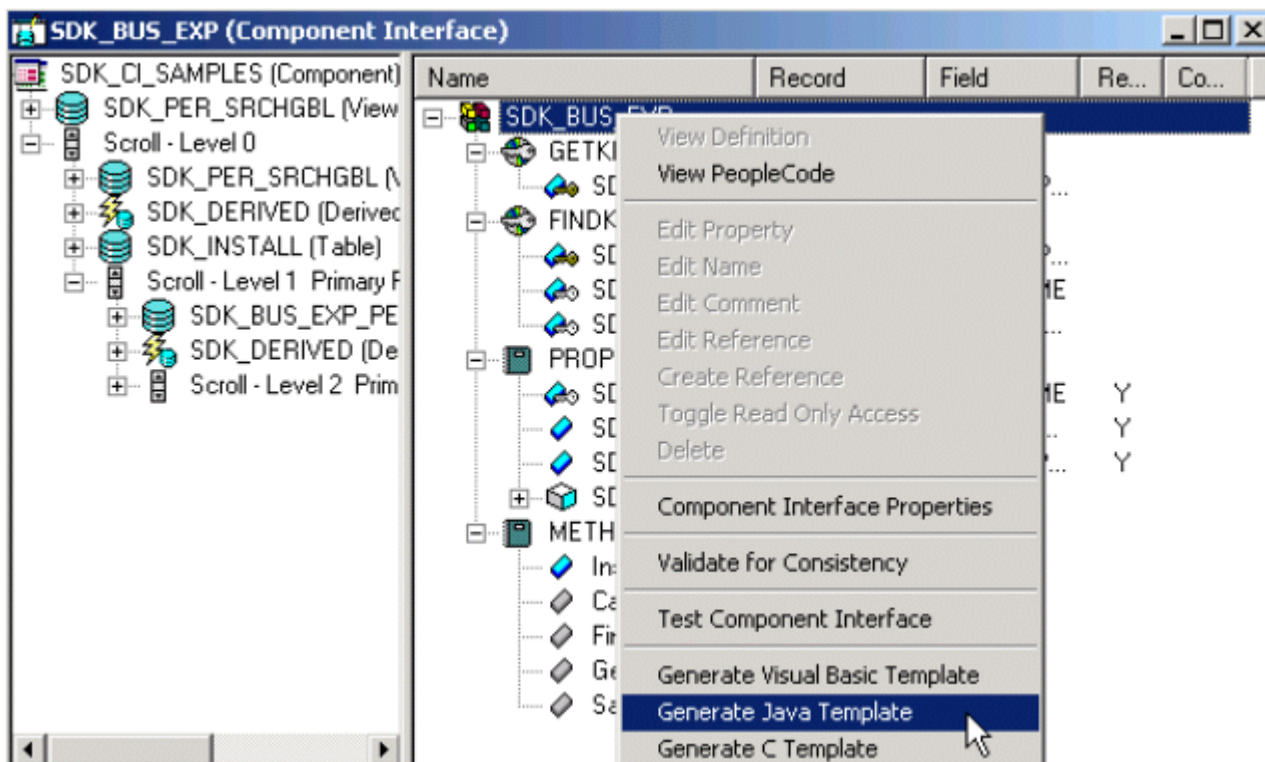
To expose a component to a third party, you must first create a component interface definition. This is done through the Component Interface Designer. Using drag-and-drop functionality, you can specify the properties and methods of the component that you wish to expose. Numerous component interface definitions are delivered out of the box.

The Business Expenses component definition is displayed in the left frame of the following example. The properties and methods that are exposed through this interface are displayed in the right frame.



BUS\_EXP component interface definition

Once the component interface definition is saved, you can then generate the Java classes for invoking this interface. This is also done using the Component Interface Designer:



SDK\_BUS\_EXP component interface

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleCode API Reference*, "Component Interface Classes."

### Invoking the Component Interface from Java

The following section walks through the steps involved in invoking the Business Expense component interface from Java.

1. Connect to the application server.

To access a component interface, you need to establish a PeopleSoft session. To create a session object, use the `Session.Connect()` method. The `Connect` method, which takes five parameters, actually logs in to a PeopleSoft session. The `Connect()` method connects a session object to a PeopleSoft application server. Note that there are various options available for using an existing connection and disconnecting and switching connections.

```
import PeopleSoft.ObjectAdapter.*;
import PeopleSoft.Generated.PeopleSoft.*;
import PeopleSoft.Generated.CompIntfc.*;
private ISession oSession;
private CAdapter oAdapter;
oAdapter = new CAdapter();
oSession = new CSession(oAdapter.getSession());
oSession.Connect(1, "//EHARRIS032000:9000", "PTDMO", "PTDMO", new byte[0]);
```

2. Get an instance of the component interface.

Use the `GetComponent()` method with a session object to get an instance of a previously created component interface.

```
busExpense = new CBusExp( oSession.GetComponent( "BUS_EXP" ) );
```

### 3. Find an existing record.

You can query a component interface to find relevant data instances based on primary and alternate search keys.

```
busExpense.setName( searchDialogStrings[ 0 ] );
busExpense.setLastNameSrch( searchDialogStrings[ 1 ] );
busExpense.setEmplid( searchDialogStrings[ 2 ] );
return( busExpense.Find() );
```

### 4. Get an instance of data.

GetKeys are the key values required to return a unique instance of existing data. GetKeys can be set using simple assignment to the properties of the component interface and then the Get() method can be invoked. This will populate the component interface with data based on the key values you set; this is what has been referred to here as a data instance.

```
busExpense.setEmplid( getKey );
boolean result = busExpense.Get();
```

### 5. Migrate through collections of data.

After getting a data instance, the next step will be to get access to the data in the component interface. PeopleSoft organizes component interface data within collections. Rows of data in a collection are called items.

The following code creates a connection to the application server, gets the component interface, and fetches the first item in a collection.

```
oAdapter = new CAdapter();
oSession = new CSession(oAdapter.getSession());
oSession.Connect(1,"//EHARRIS032000:9000","PTDMO","PTDMO",new byte[0]);
busExpense = new CBusExp( oSession.GetComponent( "BUS_EXP" ) );
busExpense.setEmplid( getKey );
boolean result = busExpense.Get();
busExpenseFirstScrollItemCollection = busExpense.getBusExpensePer();
busExpenseFirstScrollItem = firstScrollCollection.Item
( firstScrollIndex );
return( busExpenseFirstScrollItem.getBusExpenseDtl() );
```

## 6. Edit and access data in an item.

Editing and accessing component interface data in Java is rather straightforward. The following Java code accesses the various public members of the class.

```
long j = busExpenseSecondScrollCollection.getCount();
Object [][] data = new Object[ ((int)j + 1) ][ 7 ];
for( int i = 1; i < j + 1 ; i++ )
{
    busExpenseSecondScrollItem = busExpenseSecondScrollCollection.Item( i );
    data[(i - 1)][0] = busExpenseSecondScrollItem.getBusinessPurpose();
    data[(i - 1)][1] = busExpenseSecondScrollItem.getChargeDt();
    data[(i - 1)][2] = busExpenseSecondScrollItem.getCurrencyCd();
    data[(i - 1)][3] = busExpenseSecondScrollItem.getDeptid();
    data[(i - 1)][4] = busExpenseSecondScrollItem.getExpenseAmt();
    data[(i - 1)][5] = busExpenseSecondScrollItem.GetPropertyByName
        ( "ExpenseCd" );
    data[(i - 1)][6] = busExpenseSecondScrollItem.GetPropertyByName
        ( "CurrencyCd" );
}return( data );
```

In the following example, data is accessed using the `getNAME_OF_PRPERTY()` method of an item or by using the generic `getPropertyByName()` method. This code illustrates that way in which an entire collection of data can be captured and packaged into an object for transfer to a calling object.

```
busExpenseFirstScrollItem.setEmplid( emplid );
busExpenseFirstScrollItem.setExpensePeriodDt( expensePeriodDt );
return( busExpense.Save() );
```

Just as before, data is edited using item objects and the `setNameOfPropery()` method of those items. Also note that we needed to call the `Save()` method on the component interface to commit the changes.

## 7. Insert an item into a collection and delete an item from a collection.

Collection objects in Java contain the `InsertItem()` method in which the return value is the item that has just been inserted. After a new item is created, simply edit data in it and then remember to call the `Save()` method to commit the changes.

```
busExpenseSecondScrollItem = busExpenseSecondScrollCollection.
InsertItem( secondScrollIndex );
```

Similarly, there is a `DeleteItem()` method:

```
busExpenseSecondScrollCollection.DeleteItem( secondScrollIndex );
```

## 8. Disconnect from a session.

After a session is no longer needed, it should be disconnected from the application server. This is done by calling the `Disconnect()` method on the session object.

```
oSession.Disconnect();
```

---

# Developing Contextual Embeddable Pagelets

This section discusses how to develop contextual embeddable pagelets.

1. Create an embeddable pagelet using Pagelet Wizard.

See Chapter 11, "Using Pagelet Wizard," page 277.

2. Create a transaction page definition using Application Designer.

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Application Designer Developer's Guide*, "Creating Page Definitions."

3. Place an HTML Area on the page definition.

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Application Designer Developer's Guide*, "Using Page Controls," Inserting an HTML Area.

4. Write a PeopleCode function and map the pagelet parameters with any available values from the component buffer.

## PeopleCode Sample for Rendering the Context-Based Embeddable Pagelet

```

/* Import the Pagelet Wizard application package to create the embeddable
pagelet*/
import PTPPB_PAGELET:*;
import PTPPB_PAGELET:UTILITY:*;
Component object &Pagelet, &myDataSource;

/* Create the Pagelet Wizard application package and assign the pagelet
ID*/
&PWAPI = create PTPPB_PAGELET:PageletWizard();
&PageletID = "EMBEDDED_PAGELET";

/* Get the pagelet's pointer by passing the pagelet id*/
&Pagelet = &PWAPI.getPageletByID(&PageletID, False);
&myDataSource = &Pagelet.DataSource;

/* Set the pagelet parameters to default values*/
&Pagelet.PopulateInputsWithDefaults();

/* Read the pagelet parameters */
&DSPParamColl = &myDataSource.getParameterCollection();
&CollectionParamArray = &DSPParamColl.getCollectionAsArray();

/* To override the pagelet parameter default values, */
/* read the CollectionParamArray and set the parameter values */
/* from the component buffer based on the business requirement */
If &CollectionParamArray.Len > 0 Then
For &i = 1 To &CollectionParamArray.Len
If (&i = 1) Then
&DSPParameter = &CollectionParamArray [&i];
&CollectionParamArray [&i].value = PSOPRDEFN.OPRDEFNDESC;
End-If;
End-For;
End-If;

/* Get the Embeddable Pagelet HTML */
&PglHTML = &Pagelet.Execute();

/* Associate the Pagelet HTML with the HTML Area */
PSUSRPRFL_WRK.HTMLAREA = "<div class='PSTEXT'>" | &PglHTML | "</div>";

```

---

## Administering Pagelets in the Portal Registry

This section discusses how to:

- Register homepage and template pagelets.
- Modify pagelet attributes and security.
- Register URL-based pagelets.
- Register pagelet extensions.

---

**Note.** When registering pagelets, Pagelet Wizard automates the process of interacting with the portal registry. You supply some key information in the pagelet definition, such as pagelet name, title, folder, and so forth, and Pagelet Wizard passes this information on to the portal registry.

However, the portal registry encapsulates other metadata about URLs accessed by way of the portal. Even if you use Pagelet Wizard to define and initially register a pagelet, you may still need to access the portal registry to update an attribute or to register additional entries.

---

### See Also

Chapter 5, "Administering Portals," page 99

## Registering Homepage and Template Pagelets

Before you can access a pagelet through the portal, or associate a template pagelet with a target page, it must be registered in the portal registry. You can register pagelets using the Registration Wizard, or use the following procedure:

To register a new homepage pagelet or template pagelet:

1. Select *PeopleTools, Portal, Structure and Content*.
2. For template pagelets, select *Portal Objects, Template Pagelets*. Template pagelets must be registered in this folder.
3. For homepage pagelets, select any subfolder under *Portal Objects, Pagelets*. For example, you can register a homepage pagelet in the Portal Objects, Pagelets, Organizers folder.
4. Select the *Add Content Reference* link.
5. Use the drop-down list box to select *Pagelet* for the Usage Type.
6. Use the drop-down list box to select *Always use Local* for the Node Name.
7. Use the drop-down list box to select the URL Type:
  - Select *PeopleSoft Component* for PIA-based template pagelets.
  - Select *PeopleSoft Script* for iScript-based template pagelets.
8. Set the content reference parameters to point to the appropriate PIA component or iScript.
9. Set additional parameters, if needed.
10. Save the content reference.

## Modifying Pagelet Attributes and Security

Access the Content Ref Administration - General page (PeopleTools, Portal, Structure and Content. Click the Edit link for a content reference).

General

Security

Root > Portal Objects > Pagelets > Demo >

Content Ref Administration

Author: PS

Name: MY\_STOCKCHART

Parent Folder: Demo

Copy object

Select New Parent Folder

Label: My Stock Chart

Long Description: (254 Characters)

Product: ADMN

Sequence number:

Object Owner ID:

Usage Type: Pagelet

Creation Date: 08/21/2002

Storage Type: Remote by URL

Add Content Reference

URL Information

Node Name: Always use local

URL Type: Non-PeopleSoft URL

Portal URL: s/WEBLIB\_EOPPB.ISCRIPT1.FieldFormula.IScript\_PageletBuilder?EOPPB\_PAGELET\_ID=MY\_STOCKCHART

Pagelet Attributes

Default Column: Column 1

Help ID:

Content Ref Administration - General page (1 of 2)

Pagelet Attributes	
Default Column:	Column 1 <input type="button" value="v"/> Help ID: <input type="text"/>
<input type="checkbox"/> Hide minimize image <input type="checkbox"/> Hide refresh image	
Edit URL Information	
*Node Name	<input type="text" value="LOCAL_NODE"/> <input type="button" value="q"/>
URL Type:	PeopleSoft Component <input type="button" value="v"/>
Component Parameters	
*Menu Name:	<input type="text" value="EOPPB_MENU"/>
*Market:	<input type="text" value="GBL"/>
*Component:	<input type="text" value="EOPPB_USER_PREF"/>
Additional Parameters:	<input type="text" value="EOPPB_PAGELET_ID=MY_STOCKCHART"/> <input type="button" value="v"/>
Example: name1=value1&name2=value2	
<input type="checkbox"/> Hide from MSF navigation	
Content Reference Attributes	
Name:	<input type="text" value="PAGELET_ID"/> <input type="button" value="Delete"/>
Label:	<input type="text"/> <input checked="" type="checkbox"/> Translate
Attribute value:	<input type="text" value="MY_STOCKCHART"/> <input type="button" value="v"/>
<input type="button" value="Add"/>	

### Content Ref Administration - General page (2 of 2)

The portal registry stores every content reference available through the portal. The label and description defined here affect the appearance of the label and description in the navigation menu. Other attributes on this page will have an effect on the URL that will be used to reference the content associated with a definition.

See [Chapter 5, "Administering Portals," page 99.](#)

Access the Content Reference Security page. (PeopleTools, Portal, Structure and Content. Click the Edit link for a content reference. Click the Security tab.

GeneralSecurity


Root >

## Content Reference Security


Label: Change My Password

☐ Public  
☐ Author Access

The permissions for the component or script this content reference points to, control its permissions. To change these component or script permissions, click on the "View Definition" link for the appropriate permission list.

Security Authorizations				Customize   Find    First 1-2 of 2 Last
	Type	Name	Description	View Definition
1	Permission List	ALLPAGES	All pages and weblibs	<a href="#">View Definition</a>
2	Permission List	PTPT1000	PeopleSoft User	<a href="#">View Definition</a>

Inherited Security Authorizations				Customize   Find    First 1 of 1 Last
	Type	Name	Description	View Definition
				<a href="#">View Definition</a>

Security - Content Reference Security page

Permission lists can be used to secure the ability to navigate to the content item associated with this definition. One or more permission lists can be used.

See [Chapter 5, "Administering Portals," page 99](#).

## Registering URL-Based Pagelets

This is an example of the portal registration for a URL-based pagelet application that was not built with PeopleTools.

General **Security**

[Root](#) > [Portal Objects](#) > [Pagelets](#) > [Finance](#) >

### Content Ref Administration

**Name:** PAPP\_INLUMEN\_MKT\_EXT  
**\*Label:**   
**Long Description:** (254 Characters)  
**Product:**   
**Sequence number:**   
**Object Owner ID:**   
**Usage Type:**   
**Storage Type:**   
**Template Name:**

**Author:** VP1  
**Parent Folder:** Finance  
[Copy object](#) [Select New Parent Folder](#)

**\*Valid from date:**   
**Valid to date:**   
**Creation Date:** 08/25/2001

☐ No Template

[Add Content Reference](#) [Test Content Reference](#)

**URL Information**

**Node Name:**   
**URL Type:**   
**\*Portal URL:**

☐ Hide from portal navigation

General - Content Ref Administration page

General **Security**

[Root](#) > [Portal Objects](#) > [Pagelets](#) > [Finance](#) >

### Content Reference Security

**Label:** Inlumen Market Indices

**Access Type**

☒ Permission List ☐ Public

☐ Author Access

**Permissions** [Customize](#) | [Find](#) | [First](#) | [1 of 2](#) | [Last](#)

Permission List	Description	View Definition
PAPP1500	Internet Information Pagelets	<a href="#">View Definition</a> <a href="#">Delete</a>
PAPP9000	All Enterprise Portal objects	<a href="#">View Definition</a> <a href="#">Delete</a>

[Add](#)

**Inherited Permissions** [Customize](#) | [Find](#) | [First](#) | [1 of 1](#) | [Last](#)

Permission List	Description	View Definition
		<a href="#">View Definition</a>

Security - Content Reference Security page

## Registering Pagelet Extensions

This is an example of the registration for a pagelet extension that supports pagelet application personalization. In this example, the personalization component was built with PeopleTools. However, a non-PeopleSoft URL-based personalization page could have been referenced instead.

The screenshot displays the 'Content Ref Administration' interface. At the top, there are tabs for 'General' and 'Security', with 'Security' being the active tab. Below the tabs, a breadcrumb trail reads: 'Root > Enterprise Portal - Hidden > Pagelet Personalization >'. The main title is 'Content Ref Administration'. On the right, the 'Author' is listed as 'VP1'. The 'Parent Folder' is 'Pagelet Personalization', with buttons for 'Copy object' and 'Select New Parent Folder'. The form fields are as follows:

- Name:** PAPP\_EO\_PE\_RSS\_NEWS\_GBL
- \*Label:** Personalize RSS News
- Long Description:** (254 Characters)
- Product:** PAPP
- Sequence number:**
- Object Owner ID:** Enterprise Portal
- Usage Type:** Target
- Storage Type:** Remote by URL
- Template Name:** HOMEPAGE\_DESIGNER\_TEMPL
- \*Valid from date:** 01/01/1900
- Valid to date:** 12/31/2099
- Creation Date:** 08/25/2001
- Header, target only template:** ☐ No Template

Below the main form, there are links: 'Add Content Reference' and 'Test Content Reference'. A section titled 'URL Information' contains:

- \*Node Name:** Always use local
- URL Type:** PeopleSoft Component

Below 'URL Information' is the 'Component Parameters' section:

- \*Menu Name:** PORTAL\_COMPONENTS
- \*Market:** GBL
- \*Component:** EO\_PE\_RSS\_NEWS
- Additional Parameters:**

An example string is provided: 'Example: name1=value1&name2=value2'. At the bottom, there is a checkbox labeled 'Hide from portal navigation' which is checked.

### General - Content Ref Administration page

A personalization page must be identified in the registry as being related to a specific pagelet application.

Note that the personalization page (EO\_PE\_RSS\_NEWS) is referenced in the Component Parameters group box. This is where the personalization page that is related to the pagelet application is identified.

General Security

[Root](#) > [Enterprise Portal - Hidden](#) > [Pagelet Personalization](#) >

## Content Reference Security

**Label:** Personalize RSS News

**Access Type**

☐ Permission List ☒ Public

The permissions for the component or script this content reference points to, control its permissions. To change these component or script permissions, click on the "View Definition" link for the appropriate permission list.

Inherited Permissions		
<a href="#">Permission List</a>	<a href="#">Description</a>	<a href="#">View Definition</a>
		<a href="#">View Definition</a>

Customize | Find | First 1 of 1 Last

Security - Content Reference Security page

## Handling Special Situations

This section discusses how to:

- Determine pagelet running location.
- Use refresh tags in pagelets.

## Pagelet Running Location

You can determine if your code is running within the portal environment (that is being invoked using an https request coming from a portal servlet), as opposed to running in PIA outside of the portal environment. The PeopleCode %RunningInPortal system variable returns a Boolean value that lets you know whether or not you're in the portal. This variable works in both frame templates and HTML templates.

## Using Refresh Tags in Pagelets

The order of precedence of refresh tags in pagelets is as follows:

1. Template
2. Target content
3. Pagelet

Among pagelets, the first one in the HTML to include a meta refresh tag is applied. Subsequently found refresh tags are not included in the HTML.

**See Also**

[Chapter 9, "Using Portal Caching Features," page 199](#)

[Chapter 7, "Modifying the Portal Interface," Applying Changes, page 179](#)

## Chapter 13

# Working with Related Content Services

This chapter provides an overview of related content services and discusses how to:

- Define related content services.
- Assign and manage related content services.
- Use security handlers.

---

## Understanding Related Content Services

PeopleSoft Related Content Services is the framework within which subject matter experts or administrators can contextually link application pages with collaborative content.

The primary goal of related content services is to provide immediate access to relevant, contextual information for any activity without requiring any additional user effort. PeopleSoft Related Content Services Framework acts as an enterprise mashup, tying together all types of content such as Oracle Business Intelligence Enterprise Edition (OBIEE) analytics, queries, discussion threads, or any relevant non-PeopleSoft data into a single tool—PeopleSoft application pages.

Related content can be any collaborative, analytical, or informational content that is useful for performing the business process. A related content service is content, such as, a discussion forum or wiki page which is offered as a service to be consumed by other applications. Related content can be configured within the same database, among multiple PeopleSoft databases, and between a PeopleSoft application page and non-PeopleSoft data. The terms *related content* and *related content service* are synonymous.

When users access application pages that are configured with related data, the related content appears in the related content services frame of the bottom of the application page. You can assign multiple services to entire components, individual pages in a component, or a combination of the two. If an application page has multiple services, each service appears as a tab in the related content frame. In addition to the related content frame at the bottom of the page, a Related Information drop-down list box appears in the navigation bar at the top of pages configured with related content services.

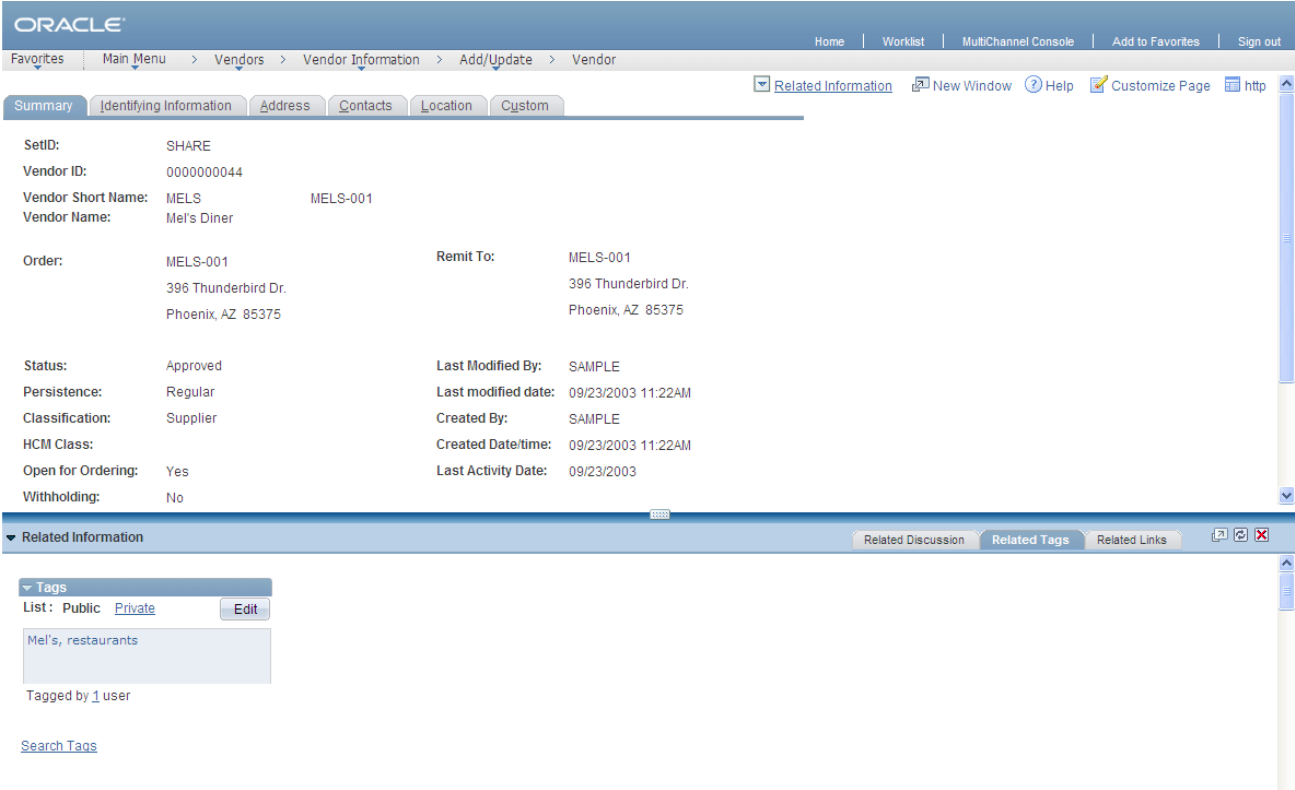
In the related content frame, you can:

- Slide the scroll bar, which is to the right of the frame, up and down to scroll through the related content.
- Resize the entire related content frame by clicking on the handle at the top middle border of the frame and dragging it up or down.
- Refresh the frame contents by clicking the refresh button in the header of the frame.
- Collapse the frame by clicking the collapse button to the left of the Related Information frame header.

- Close the frame by clicking the close button.

**Note.** To make the frame visible again, you must either return to the search page or access another page with associated related content.

You can see the Vendor - Summary page, the Related Information drop-down list box, and three related content services in the following example:



Related content services associated with the Vendor component

To use related content, developers and subject matter experts perform the following tasks:

- Develop the related content service.
- Develop the related content security.
- Define which content is related to which page or component.
- Map contextual fields.

Prerequisites

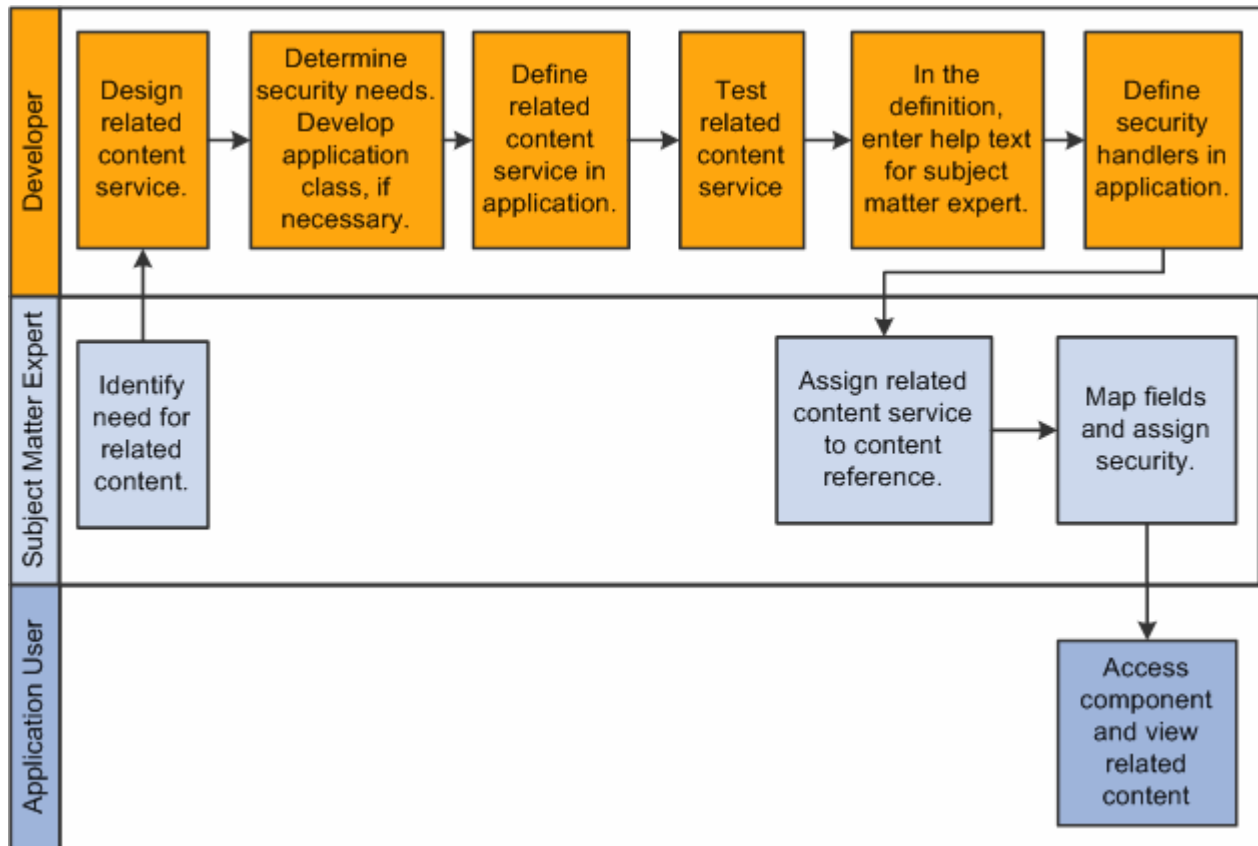
You must perform the following configuration tasks before implementing related content among applications:

Task	Reference
Configure single signon among all databases that are sharing related content.	See <i>Enterprise PeopleTools 8.50 PeopleBook: Security Administration</i> , "Implementing Single Signon."

Task	Reference
Enter an authentication domain in the web profile.	See <a href="#">Chapter 10, "Configuring the Portal Environment,"</a> <a href="#">Configuring General Portal Properties, page 220.</a>

### Related Content Business Process Flow

The following diagram illustrates the Related Content Framework business process along with the role involved in each step.



Business process flow for related content development and implementation

### Related Content Service Types

The following related content service types are accessible in the related content service frame:

- |                             |   |
|-----------------------------|---|
| <b>Non-PeopleSoft</b>       | This service type accesses any non-PeopleSoft pages, external content, or services in the related content services frame.   |
| <b>PeopleSoft Component</b> | This service type accesses any PeopleSoft components in the related content services frame. The component can be from the same PeopleSoft system or from a remote PeopleSoft system. Node name decides whether the service resides in the same PeopleSoft system or remote PeopleSoft system. |

<b>PeopleSoft Embeddable Pagelet</b>	This service type accesses Pagelet Wizard pagelets in the related content services frame. The pagelet can be of any pagelet data source but it should be published as an <i>Embeddable</i> pagelet.
<b>PeopleSoft OBIEE Pagelet</b>	This service type accesses analytics-based Oracle Business Intelligence Enterprise Edition (OBIEE) pagelets in the related content services frame. OBIEE reports are analytics reports that are consumed in the PeopleSoft system and pagelets are created and published as embeddable pagelets through the Pagelet Wizard based on an OBIEE data source.
<b>PeopleSoft Query</b>	This service type accesses PeopleSoft queries in the related content services frame. The query results can also be filtered based on the query definition and the parameters it requires.
<b>PeopleSoft Script</b>	This service type accesses iScript-based services in the related content services frame. Use this service type in situations in which an iScript has been used to develop the application functionality based on PeopleSoft data or non-PeopleSoft data, such as stock quotes or weather information.

---

## Understanding Delivered Related Content Services

PeopleTools provides the service definitions for three PeopleSoft Enterprise Portal related content services:

- Related Discussion service
- Related Links service
- Related Tags service

These related content services provide features of Oracle's PeopleSoft Enterprise Portal that can be added to PeopleSoft application pages through the PeopleTools Related Content Framework. While the service definitions are provided in PeopleTools, an active PeopleSoft Enterprise Portal database is required to provide the services and to store the related data.

Setup procedures for these services is covered in the PeopleSoft Enterprise Portal PeopleBooks.

See *PeopleSoft Enterprise Portal 9.1 PeopleBook: Using Portal Features*, "Setting Up Related Content Services."

## Related Discussion Service

Using the Related Discussion related content service, you can collaborate to share relevant information about a transaction or a transaction instance. Discussion forums in the Related Discussion service provide a platform that groups can use to discuss topics of interest in a manner similar to standalone forums in PeopleSoft Enterprise Portal. Discussion forums enable multiple relevant individuals to contribute to the review and resolution of a question. You can post discussion topics, such as issues, suggestions, or questions, and receive replies and feedback.

Details for using the Related Discussion service are provided in the PeopleSoft Enterprise Portal PeopleBooks.

See *PeopleSoft Enterprise Portal 9.1 PeopleBook: Using Portal Features*, "Working With the Related Discussion Service."

## Related Links Service

The Related Links related content service allows you to create links to relevant content for the current transaction instance or for all instances in a transaction. You can link to content residing in PeopleSoft Enterprise Portal—specifically, blogs, discussions, or collaborative workspaces—or external URLs. In addition, the Related Links service allows you to create a new workspace and to link that workspace to the transaction.

Details for using the Related Links service are provided in the PeopleSoft Enterprise Portal PeopleBooks.

See *PeopleSoft Enterprise Portal 9.1 PeopleBook: Using Portal Features*, "Working With the Related Links Service."

## Related Tags Service

The Related Tags related content service allows you to create user-specified tags for the current transaction instance. You can characterize and bookmark these transactional or business objects with terminology of your choosing, providing ease of access as well as collaborative classification of the transaction.

Details for using the Related Tags service are provided in the PeopleSoft Enterprise Portal PeopleBooks.

See *PeopleSoft Enterprise Portal 9.1 PeopleBook: Using Portal Features*, "Working With the Related Tags Service."

## Common Elements in This Chapter

This section describes the common elements on the Define Related Content Service page.

### *Related Content Service*

This table describes the common fields on the Define Related Content Service page. These fields are common to all service URL types:

<b>Service ID</b>	Enter the unique identifier for the related content service.
<b>Service Name</b>	Enter the name of the service. Enter up to 18 characters.
<b>Description</b>	Enter a description for the related content service.

<b>URL Type</b>	<p>Select from six service types:</p> <p><i>Non-PeopleSoft URL</i></p> <p><i>PeopleSoft Component URL</i></p> <p><i>PeopleSoft Embeddable Pagelet</i></p> <p><i>PeopleSoft OBIEE Pagelet</i></p> <p><i>PeopleSoft Query</i></p> <p><i>PeopleSoft Script</i></p> <p>See <a href="#">Chapter 13, "Working with Related Content Services," Understanding Service URL Types, page 459.</a></p>
<b>Write help text</b>	<p>Click to open a help text page where a developer can enter instructions that are specific to assigning this related content service. A subject matter expert can access the help text for assistance when making a related content assignment.</p> <p>See <a href="#">Chapter 13, "Working with Related Content Services," Assigning Related Content Services to Application Pages, page 470.</a></p>
<b>Copy Service Definition</b>	Click to clone a copy of the service definition and its parameters.

### URL Information

This table describes URL information fields. These fields are common to several or all service URL types:

<b>Node Name</b>	<p>Enter the node name of the PeopleSoft system where the service resides.</p> <p>If the service resides in the same PeopleSoft system, the node name will be either <i>LOCAL_NODE</i> or any default local node of the PeopleSoft system.</p>
<b>Populate parameters</b>	<p>Populates the service parameters in the Service URL parameter grid based on the URL type. This button appears for PeopleSoft Embeddable pagelet, PeopleSoft OBIEE pagelet, and PeopleSoft Query service types only.</p> <hr/> <p><b>Note.</b> For embeddable and OBIEE pagelets, only context sensitive parameters are populated. For Query pagelets, only prompt fields are populated.</p> <hr/> <p>See <a href="#">Chapter 11, "Using Pagelet Wizard," Step 3: Specifying Pagelet Data Source Parameters, page 324.</a></p>
<b>Post mapping definition data</b>	Select to have the mapping definition data, such as the parameter name and its mapping page field or key field name, and the parameter's actual mapped value posted with the service request. This option is cleared by default.
<b>Escape URL Parameters</b>	Select to have the parameters defined in the Service URL Parameter grid escaped when forming the service URL. The default is selected.
<b>Show Formed URL</b>	Click to display the entire formed URL with test parameter values in a display-only edit box beneath the Service URL Parameter grid.

<b>Formed URL</b>	This field appears only after you click the Show Formed URL link. The formed URL appears within a display-only edit box.
<b>Test Related Content Service</b>	Click to test the related content service in a new window. The Test Related Content Service page appears enabling you to enter a parameter value and test the functionality of the service.

### ***Service URL Parameter***

A service might need additional parameters to contextualize the data provided by the service. You define these parameters in the Service URL parameter grid.

This table describes service URL parameter fields. These fields are common to all service URL types:

<b>Parameter Name</b>	<p>Enter any additional service parameters for the service. For example, a service might require a search parameter.</p> <p>You can map the parameter to a contextual field on an application page when you assign the service to the page.</p> <p>See <a href="#">Chapter 13, "Working with Related Content Services," Assigning Related Content Services to Application Pages, page 470.</a></p>
<b>Required</b>	<p>Select this check box when the additional parameter is necessary for a service to return a valid output. This check box is cleared by default.</p> <p>For instance, when a service is launched, it might display default output if a null or blank value is passed to a parameter. If the service is designed with no default value and a blank value is passed to the parameter, an error or blank output could occur, depending on how the service is designed to handle this scenario. To avoid this situation, you can mark one or more additional service parameters as required so that the service will be invoked with a valid value for its additional service parameters.</p> <hr/> <p><b>Important!</b> When this required check box is selected, the subject matter expert or administrator must map a field or value to the parameter on the Related Content - Map Fields page or the system does not display the service.</p> <hr/>
<b>Description</b>	Enter a parameter description that is meaningful for the subject matter expert or administrator who configures the related content assignment and mapping.

### ***Display Options***

This table describes display options fields. These fields are common to all service URL types:

<b>Refresh</b>	Select to have the services frame display an additional button to refresh the frame.
----------------	--

**New Window**

Select to have the services frame display an additional button to open the services frame content in a new window.

When you choose to open the content in a new browser window or browser tab (based on the browser version and settings), the browser uses the current URL and the window has no contextual link to the parent application page. This means that as the context of the application page changes, the child window content does *not* change and you can continue to view and use the content in the child window. This is important in situations in which you are viewing content and might want to continue using it after completing work in the application page.

**Service Security Options**

This table describes service security options fields. These fields are common to all service URL types:

**Public Access**

Select to have no security applied.

This security option is intended for services such as generic help policy or various types of internet searches in which a keyword from a PeopleSoft application page field is accepted.

**Related Content Security**

Select to have the related content service, for example a discussion forum, use its own security, which it implements by using its own security model.

**Transaction Security**

Select to have the related content service, for example a discussion forum, use the security of the PeopleSoft transaction, which it implements real-time.

This security option indicates that application security such as basic PeopleTools security, row level security, or special security logic such as the HCM Direct Reports API is used to determine access to the related content. The application class that implements row level security provided by PeopleTools can be used with the Job Data component in HCM. For example, the HCM subject matter expert or related content administrator should pick the SEC\_ROWLEVEL application class for the Discussion Service, should she want the Discussion Service to implement the row level security of Job Data component.

For more information, consult the security documentation that is specific to your installed PeopleSoft applications.

See [Chapter 13, "Working with Related Content Services," Defining Security Handlers, page 477.](#)

**Implements hide from search**

Select to have the provider prevent the related content that is created by means of this service from being available when another user searches the content in the provider database. For example, if a discussion is created that contains sensitive information such as personal medical records, the subject matter expert can check this check box to block the content of the discussion from any search that is initiated from outside of all instances of this transaction.

Selecting this option ensures privacy of the related content and limits the access to the related content only in the related content model. Selecting this option essentially makes the related content private to the transaction data and the only way to access the related content is through the transaction instance. This option allows sensitive related content to be shown only in the related content frame in the context of the transaction and eliminates the overhead of security synchronization between the related content consumer and provider systems.

Using this option limits the ways in which information can be reused. To enable this option, the service provider should invoke the PTCS\_SEARCHENABLED web service to retrieve the service configuration settings in which the transactions are mapped to the service.

---

**Note.** To ensure that the security of the data is not breached, the Implements hide from search check box data does not appear in the HTTP request from the application pages.

---

---

## Defining Related Content Services

This section provides an overview of service URL types and discusses how to:

- Select the Non-PeopleSoft URL type.
- Select the PeopleSoft Component URL type.
- Select the PeopleSoft Embeddable Pagelet URL type.
- Select the PeopleSoft OBIEE Pagelet URL type.
- Select the PeopleSoft Query URL type.
- Select the PeopleSoft Script URL type.
- Test the related content service.
- Write help text for related content service definitions.
- Copy related content service definitions.

## Understanding Service URL Types

The Related Content Framework supports six service types. Each type is defined by its URL construction and each type requires different information to form the complete URL that requests the contextual data for the related content frame.

When you select a service URL type, various URL information parameters appear for you to configure. Some of the parameters are predefined for each service URL type. For example, when you select *PeopleSoft Embeddable Pagelet*, the Node and Pagelet Name fields appear as predefined parameters, because the framework must know in which node the pagelet resides and the name of the pagelet to construct the URL and retrieve the pagelet and its data.

In addition to predefined parameters, you can enter user-defined parameters which offers maximum flexibility in configuring the service to meet business requirements.

The information that the system uses to construct the six URL types differs for each type as described in the following sections.

## Pages Used to Define Related Content Services

<b>Page Name</b>	<b>Definition Name</b>	<b>Navigation</b>	<b>Usage</b>
Define Related Content Service	PTCSSERVICES	PeopleTools, Portal, Related Content Services, Define Related Content Service	Create the related content service to map to an application page.
Test Related Content Service	PTCS_TESTURL	On the Define Related Content Service page, click the Test Related Content Service link.	Test the functionality of the currently defined service.
Copy Related Content Service Definition	PTCSSERVICE_SAVEAS	On the Define Related Content Service page, click the Copy Service Definition button.	Clone a related content service definition.
Related Content Service Definition - Help information	PTCS_HELPTEXT_SEC	On the Define Related Content Service page, click the Write Help Text button.	Enter Help information regarding the use of the related content parameters.

## Creating the Non-PeopleSoft URL Type

Access the Define Related Content Service page (PeopleTools, Portal, Related Content Service, Define Related Content Service). Select the *Non-PeopleSoft URL* URL type.

URL Type: Non-PeopleSoft URL ? Write help text Copy Service Definition

**URL Information**

\*External URL:

Note: parameter names are case-sensitive. ☐ Post mapping definition data. ☒ Escape URL Parameters

Service URL Parameter			First	1 of 1	Last
	*Parameter Name	Required	*Description		
1	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input data-bbox="1393 422 1414 449" type="button" value="+"/>	<input data-bbox="1425 422 1446 449" type="button" value="-"/>

Formed URL: [Show Formed URL](#) [Test Related Content Service](#)

Define Related Content Service page - Non-PeopleSoft URL type

### URL Information

**External URL** Enter a non-PeopleSoft application URL.

### Service URL Parameters

Enter any parameters that the external service requires.

### Non-PeopleSoft URL Example

This is an example of a fully formed non-PeopleSoft URL:

*http://finance.google.com/finance?q=ORCL*

## Creating the PeopleSoft Component URL Type

Access the Define Related Content Service page (PeopleTools, Portal, Related Content Service, Define Related Content). Select the *PeopleSoft Component URL* URL type.

URL Type: PeopleSoft Component URL ? Write help text Copy Service Definition

**URL Information**

\*Node Name:

**Component Parameters**

\*Menu Name:  \*Market:  \*Component Name:

Note: parameter names are case-sensitive. ☐ Post mapping definition data. ☒ Escape URL Parameters

**Service URL Parameter**

	*Parameter Name	Required	*Description
1	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>

[Show Formed URL](#) [Test Related Content Service](#)

Define Related Content Service page - PeopleSoft Component URL type

### Component Parameters

**Menu Name** Enter the definition name of the PeopleSoft menu on which the component is located as configured in Application Designer.

**Market** Enter the market, such as *GBL*, of the component as configured in Application Designer.

**Component Name** Enter the component definition name as configured in Application Designer.

### Component URL Example

This is an example of a fully formed component URL:


*http://pta095.dsi-  
inet.peoplesoft.com/ps/ps/EMPLOYEE/PT\_LOCAL/c/MAINTAIN\_SECURITY.USERMAINT.GBL?OPRID=  
PTPORTAL&OPRDEFNDESC=Portal Administrator*

## Creating the PeopleSoft Embeddable Pagelet URL Type

Access the Define Related Content Service page (PeopleTools, Portal, Related Content Service, Define Related Content). Select the *PeopleSoft Embeddable Pagelet* URL type.

URL Type: **PeopleSoft Embeddable Pagelet** ? Write help text Copy Service Definition

**URL Information**

\*Node Name:  

\*Pagelet ID:

Note: parameter names are case-sensitive. ☐ Post mapping definition data. ☒ Escape URL Parameters

Service URL Parameter			First	1 of 1	Last
	*Parameter Name	Required	*Description		
1	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>		

[Show Formed URL](#) [Test Related Content Service](#)

Define Related Content Service page - PeopleSoft Embeddable Pagelet URL type

## URL Information

### Pagelet ID

Enter the unique identifier of the embeddable pagelet created in the Pagelet Wizard.

See [Chapter 11, "Using Pagelet Wizard," Step 2: Selecting a Pagelet Data Source, page 312.](#)

## Service URL Parameters

Click the Populate parameters button to populate the Service URL Parameter grid with parameters that you defined as *Context Sensitive* in step three of the pagelet construction in the Pagelet Wizard.

If the pagelet resides in the same system, as soon as you select the node, a lookup appears next to the pagelet ID field. When clicked, the lookup lists all of the available embeddable pagelets in the system.

The Populate parameters button populates the parameters only for the locally available pagelets. If a pagelet resides in a remote PeopleSoft system, you must manually enter the pagelet ID and pagelet parameters.

See [Chapter 11, "Using Pagelet Wizard," Step 3: Specifying Pagelet Data Source Parameters, page 324.](#)

## Embeddable Pagelet URL Example

This is an example of a fully formed embeddable pagelet URL:

```
http://pta095.dsi-
inet.peoplesoft.com/psc/ps/EMPLOYEE/PT_LOCAL/s/WEBLIB_PTEMBED.ISCRIPT1.FieldFormula.IScript
_EmbeddedPageletBuilder?PAGELETID=USER_ROLES&OPRID=PTDMO&.MAXROWS=10
```

## Creating the PeopleSoft OBIEE Pagelet URL Type

Access the Define Related Content Service page (PeopleTools, Portal, Related Content Service, Define Related Content). Select the *PeopleSoft OBIEE Pagelet URL* type.

URL Type: PeopleSoft OBIEE Pagelet ? Write help text Copy Service Definition

**URL Information**

\*Node Name:  🔍

\*Pagelet ID:

Note: parameter names are case-sensitive. ☐ Post mapping definition data. ☒ Escape URL Parameters

Service URL Parameter				First	1 of 1	Last
	*Parameter Name	Required	*Description			
1	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>			<span>+</span> <span>-</span>

[Show Formed URL](#) [Test Related Content Service](#)

Define Related Content Service page - PeopleSoft OBIEE Pagelet URL type

## URL Information

### Pagelet ID

Enter the unique identifier of the OBIEE pagelet created in Pagelet Wizard.

See [Chapter 11, "Using Pagelet Wizard," Step 1: Entering Pagelet Identifying Information, page 310.](#)

## Service URL Parameters

Click the Populate parameters button to populate the Service URL Parameter grid with parameters that you defined as *Context Sensitive* in step three of the pagelet construction in the Pagelet Wizard.

If the pagelet resides in the same system, as soon as you select the node, a lookup appears next to the pagelet ID field. When clicked, the lookup lists all of the available OBIEE pagelets in the system.

The Populate parameters button populates the parameters only for the locally available pagelets. If a pagelet resides in a remote PeopleSoft system, you must manually enter the pagelet ID and pagelet parameters.

See [Chapter 11, "Using Pagelet Wizard," Step 3: Specifying Pagelet Data Source Parameters, page 324.](#)

## OBIEE Pagelet URL Example

This is an example of a fully formed OBIEE pagelet URL:

```
http://pta095.dsi-
inet.peoplesoft.com/psc/ps/EMPLOYEE/PT_LOCAL/s/WEBLIB_PTEMBED.ISCRIPT1.FieldFormula.IScript
_EmbeddedPageletBuilder?PAGELETID=OBIEE_REPORT&1) Markets.Region=EASTERN REGION
```

## Creating the PeopleSoft Query URL Type

Access the Define Related Content Service page (PeopleTools, Portal, Related Content Service, Define Related Content). Select the *PeopleSoft Query* URL type.

URL Type: PeopleSoft Query ? Write help text Copy Service Definition

**URL Information**

\*Node Name:  🔍

\*PS Query:

Note: parameter names are case-sensitive. ☐ Post mapping definition data. ☒ Escape URL Parameters

Service URL Parameter			First	1 of 1	Last
	*Parameter Name	Required	*Description		
1	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>		

[Show Formed URL](#) [Test Related Content Service](#)

Define Related Content Service page - PeopleSoft Query URL type

### URL Information

**PS Query** Enter the query definition name as defined in Query Manager.

### Service URL Parameters

Click the Populate parameters button to populate the Service URL Parameter grid with fields that you defined as *prompts* when creating the query definition.

If the query resides in the same system, as soon as you select the node, a lookup appears next to the PS Query field. When clicked, the lookup lists all of the available queries in the system.

The Populate parameters button populates the parameters only for the locally available queries. If a query resides in a remote PeopleSoft system, you must manually enter the query name and query parameters.

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Query*, "Defining Selection Criteria," Defining Prompts.

### PeopleSoft Query URL Example

This is an example of a fully formed PeopleSoft Query URL:

```
http://pta095.dsi-
inet.peoplesoft.com/psc/ps_newwin/EMPLOYEE/PT_LOCAL/q/?ICAction=ICQryNameURL=PUBLIC.MESS
AGES_FOR_MSGSET&MESSAGE_SET_NBR=240
```

## Creating the PeopleSoft Script URL Type

Access the Define Related Content Service page (PeopleTools, Portal, Related Content Service, Define Related Content). Select the *PeopleSoft Script* URL type.

URL Type: PeopleSoft Script ? Write help text Copy Service Definition

**URL Information**

\*Node Name:

**iScript Parameters**

\*Record (Table) Name:  \*Field Name:

\*PeopleCode Event Name:  \*PeopleCode Function Name:

Note: parameter names are case-sensitive. ☐ Post mapping definition data. ☒ Escape URL Parameters

**Service URL Parameter** First 1 of 1 Last

	*Parameter Name	Required	*Description
1	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>

[Show Formed URL](#) [Test Related Content Service](#)

Define Related Content Service page - PeopleSoft Script URL type

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleCode API Reference*, "Internet Script Classes (iScript)," Web Libraries.

### ***iScript Parameters***

**Record (Table) Name** Enter the name of the WEBLIB\_ record where the iScript is stored.

**Field Name** Enter the name of the field where the iScript is stored.

**PeopleCode Event Name** Enter the name of the PeopleCode event under which the iScript is stored.  
The default event is FieldFormula.

**PeopleCode Function Name** Enter the name of the function that performs the logic of the service.

### ***Service URL Parameter***

Enter the names of all parameters that the iScript must have to render the service in the related content service frame.

### ***PeopleSoft Script URL Example***

This is an example of a fully formed PeopleSoft script URL:




```
http://pta095.dsi-
inet.peoplesoft.com/psc/ps/EMPLOYEE/PT_LOCAL/s/WEBLIB_STOCK.ISCRIPT1.FieldFormula.IScript_Sto
ckQuote?stockquote=ORCL
```

## **Testing the Related Content Service**

Access the Test Related Content Service page (on the Define Related Content Service page, click the Test Related Content Service link).

## Test Related Content Service

Enter valid values for parameters list below and click Test Related Content Service.

Find    First  1 of 1  Last	
Parameter Name	Parameter Value
1 q	<input type="text"/>

[Test Related Content Service](#)

Return

Test Related Content Service page

<b>Parameter Name</b>	Parameters that you enter on the Define Related Content Service page appear as a list in this area. This field is display-only.
<b>Parameter Value</b>	Enter a test value for the parameter.
<b>Test Related Content Service</b>	Click to test the related content service with the parameter values that you enter.
<b>Return</b>	Click to return to the Define Related Content Service page after testing the service.

### Testing Related Content Service Parameter Values

To test parameter values:

1. Enter the value in the Parameter Value field.
2. Click the Test Related Content Service link.

The service appears in a separate browser window with the parameter value.

3. Click the Return button to return to the Define Related Content Service page.

## Writing Help Text for Related Content Service Definitions

Access the Related Content Service Definition - Help information page (on the Define Related Content Service page, click the Write help text button).

### Related Content Service Definition - Help information

Service Name: GOOGLESRCH

Enter help information for the Related Content Service Definition "GOOGLESRCH" in the editor window below. This will be displayed as help information in the "Related Content Service - Map fields" page while assigning Related Content Service Definition to an application page.

Related Content Service Definition - Help information page

<b>Service Name</b>	The name of the related content service. This field is display-only.
<b>OK</b>	Click to save the text and return to the Related Content Service Definition page. <hr/> <b>Note.</b> You must save the definition to save the help text. <hr/>
<b>Cancel</b>	Click to discard any changes and return to the Related Content Service Definition page.
<b>Refresh</b>	Click to refresh the information and HTML in the editor and remain on the Help information page.

To write Help text for a related content service definition:

1. Enter help information for the related content service definition in the editor window.

This information appears when you click the Read help text link on the Related Content Service - Map fields page while mapping service parameters and assigning Related Content Services to application pages.

Help text is useful on the Map Fields page when the subject matter expert or administrator has to map complex parameters to make the service work as designed.

2. Click the OK button.

This returns the user to the Related Content Service Definition page.

3. Save the definition.

## Copying Related Content Service Definitions

Access the Copy Related Content Service Definition page (on the Related Content Service Definition page, click the Copy Service Definition button).

### Copy Related Content Service Definition

Service ID: RC\_NPSU  
Service Name: GOOGLESRCH

Save Related Content Service Definition As

\*Service ID:

\*Service Name:

Copy Related Content Service Definition page

To copy a related content service definition:

1. Enter a new value in the Service ID field.
2. Enter a new value in the Service Name field.
3. Click the Save button.

A message appears indicating a successful save.

4. Click the OK button to dismiss the message box.
5. Click the Go to Related Content Service Definition link that appears to access the new related content service definition or click the Cancel button to access the existing related content definition.

---

**Note.** Clicking the Cancel button does not cancel the save. The transaction is saved when you click the Save button.

---

---

## Assigning and Managing Related Content Services

This section discusses how to:

- Assign related content services to application pages.
- Map fields to service parameters.
- Manage related content configuration.

## Pages Used to Assign and Manage Related Content Services

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Manage Related Content Configuration	PTCS_SRVCFG_SRCH	PeopleTools, Portal, Related Content Service, Manage Related Content Service	Assign and manage related content services
Select a Content Reference	PTCS_CRFURL_SELECT	On the Manage Related Content Configuration page, click the Assign Related Content to Application Pages link.	Select content reference to assign related content to.
Assign Related Content	PTCS_SERVICECFG	On the Manage Related Content Configuration page, click the Assign Related Content to Application Pages link.	Assign related content service to component or page.
Related Content Service - Map Fields	PTCS_FIELDMAPS_SEC	On the Assign Related Content page, click the Map Fields link.	Configure mapping of service parameters to PeopleSoft page data.

## Assigning Related Content Services to Application Pages

Access the Manage Related Content Configuration page (PeopleTools, Portal, Related Content Service, Manage Related Content Service).

### Manage Related Content Configuration

*There are currently no application pages that have the Related Content Services assigned.*

[Assign Related Content to Application Pages](#)

[Create a New Related Content Service](#)

Manage Related Content Configuration page without any existing assignments

The Manage Related Content Configuration page is a central starting point for creating and assigning related content. The first time you access this page, you will see that no application pages have been assigned related content. After you assign related content services to application pages, a grid appears listing the related content assignments.

See [Chapter 13, "Working with Related Content Services," Managing Related Content Configuration, page 475.](#)

Click the [Assign Related Content to Application Pages](#) link to get started. This accesses the page where you select the content reference to which you are assigning the related content.

---

**Note.** The tree shows both PeopleSoft component URLs and PeopleSoft component-based generic URLs.

---

Select **Include hidden Crefs** to make hidden content references available for assignment. Selecting this check box redraws the tree including hidden crefs in the system. Clearing it redraws the tree with only the visible crefs in the system.

## Select a Content Reference

Click a content reference link to pick a content reference  
Click "Cancel" to go back to [Manage Related Content Service](#) page.

☐ **Include hidden Crefs**

Left | Right

---

 **Root**  
  
[Administer Procurement](#)  
  
[Control Inventory](#)  
  
[Define Business Rules](#)  
  
[Maintain Items](#)  
  
[Manage Production](#)  
  
[Manage Sales Activities](#)

Select a Content Reference or Content Reference Link page

---

**Important!** When you select a content reference, the services that you assign are unique to the menu *and* component combination. The same component, when it is placed on different menus, can take different services as components on different menus might have different business requirements.

---

When the [Select a Content Reference or Content Reference Link](#) page appears, navigate the tree hierarchy and select the content reference. When you click a content reference, the [Assign Related Content](#) page appears. An active link to the content reference appears beside the content reference name.

## Assign Related Content

Portal Name: EMPLOYEE

Content Reference: PT\_USERMAINT\_GBL

[User Profiles](#)

▼ Service Level

To configure different services for each page on a multi-tabbed component, select the "Services at Page level" radio button below and select the services for each page. Otherwise, leave the radio button at "Services at Component level" and select the services that will be common across all the pages in the selected component.

☐ Services at Component Level
 ☒ Services at Page level

Related Content to Display

Find

First 1 of 1 Last

	Page	Page Name	Service ID	Service Name	Default Service	Hide from Search	Sequence number	Map Fields		
1	USER_GENERA	General	QUERY_DOC	User Profile Query	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0	<a href="#">Map Fields</a>	<a href="#">+</a>	<a href="#">-</a>

☒ Open Services in expanded view

[Return to Manage Related Content Configuration page](#)

### Assign Related Content page

**Portal Name** This is the name of the current portal where the related content will be assigned. This field is display-only.

**Content Reference** This is the content reference to which you are assigning the related content. This is the name of the content reference that you selected on the Select a Content Reference page. This field is display-only.

**Open Services in expanded view** Select to have the related content frame open in the expanded view when you access the transaction page. This check box is enabled when you select a default service in the Related Content to Display grid. The default is unchecked.

### Service Level

**Services at Component Level** Select to view or assign related content to appear across all component pages.

**Services at Page level** Select to view or assign related content to appear on individual pages in a component.

---

**Note.** The same service cannot be assigned at both the component and page levels. If you decide to change the level at which the service appears, you must remove it from the existing level before assigning it at the new level.

---

### Related Content to Display

**Service ID** Select the service to assign to the content reference.

<b>Service Name</b>	The name of the service appears after you select a service ID. This field is display-only.
<b>Default Service</b>	Select to make a service the default service that appears when you access the transaction page. The default service is enabled for component level service only.  If you do not select a default service for the content reference, then the Open Services in Expanded View check box is disabled and the Related Content frame is closed when you initially access the transaction content reference. You may select only one related content service as the default service.
<b>Hide from Search</b>	Select to specify whether the current service related information for that transaction page is searchable in the services frame. This field applies to the search option that is available for Discussions service only. This also ensures that the discussion service provided by the Enterprise Portal disallows the content created in the discussion forum that is related to this transaction to be searched or accessed in Enterprise Portal product.
<b>Sequence number</b>	Enter a sequence number. This value determines the order in which services appear in the related content frame when you assign multiple related content services.
<b>Map fields</b>	Click to access the Related Content Service - Map Fields page where you assign parameters to parameter values.

## Mapping Fields to Service Parameters

Access the Related Content Service - Map Fields page, (on the Assign Related Content page, click the Map Fields link.

### Related Content Service - Map Fields

Service Name: **GOOGLESRCH** [Read help text](#)

Map Parameters							First	1 of 1	Last
	Service Parameter	Required	MapType	Map Data		Refresh Service On Change	Is Value Required?		
1	search parameter	<input type="checkbox"/>	Key Field	User ID		<input checked="" type="checkbox"/>	<input type="checkbox"/>		

▼ Service Security Options

☐ Public Access  
☐ Related Content Security  
☒ Transaction Security

Security Application Class:

Related Content Service - Map Fields page

<b>Service Name</b>	The name of the service as defined on the Related Content Service Definition page.
<b>Read help text</b>	Click to display the help text as defined on the Related Content Service Definition page.  See <a href="#">Chapter 13, "Working with Related Content Services," Writing Help Text for Related Content Service Definitions, page 467.</a>
<b>Map Parameters</b>	
<b>Service Parameter</b>	Displays the name of the service parameter as defined on the Related Content Service Definition page in the URL information scroll area.
<b>Required</b>	Displays whether the service parameter is defined as <i>Required</i> on the Related Content Service Definition page in the Service URL Parameter grid.  See <a href="#">Chapter 13, "Working with Related Content Services," Service URL Parameter, page 457.</a>
<b>Map Type</b>	Select from three mapping types to map the page field data to:  <i>Fixed Value:</i> use when you want to map the service parameter to a constant.  <i>Key Field:</i> use when you want to map the service parameter to one of the level 0 search keys of the component assigned to the content service.  <i>Page Field:</i> use when you want to map the service parameter to any level 0 page field.  <i>System Variable:</i> use when you want to map the service parameter to one of the available system variables.
<b>Map Data</b>	Depending on the map type, enter a constant, select a key field, a page field, or select a system variable.  When you select either the Key Field or System Variable map type, a Map Data prompt appears for you to select from valid values. <hr/> <b>Note.</b> You can map parameters to level zero fields only; only level zero fields appear in the prompt. <hr/>
<b>Refresh Service On Change</b>	Select to have the service refresh when the key values on the application page changes. The default value is cleared.
<b>Is Value Required?</b>	Some service might accept valid null values being passed to its parameters. If the Refresh Service On Change and the Is Value Required? check boxes are selected for a parameter, then when the mapped field value of that parameter on the application page changes to a null value, the system does not refresh the services frame.

---

**Note.** To reduce mapping effort the Feed Publishing Framework defines a set of reserved words and service parameters.

---

See [Appendix H, "Related Content Reserved Words and Service Parameters," page 575.](#)

**Service Security Options**

The security options selected in the related content service definition, specify which security options appear.

<b>Public Access</b>	Select to make the related content visible to anyone who is authorized to access the related content parent page.
<b>Related Content Security</b>	Select to have the service invoke the security that is native to the content, such as security views in a PeopleSoft Financial application.
<b>Transaction Security</b>	Select to have the service apply a security handler. When you select this option, the Security Application Class field appears.
<b>Security Application Class</b>	<div>This field appears when you select the Transaction Security option.</div> <div>Enter the application class name in which the security logic resides.</div>

**Managing Related Content Configuration**

Access the Manage Related Content Configuration page (PeopleTools, Portal, Related Content Service, Manage Related Content Service).

**Manage Related Content Configuration**

Application pages, that have the Related Content services assigned, are shown below:

Services assigned to Application Pages			
	Content Reference	Edit Configuration	Remove Configuration
1	User Profiles	Edit Configuration	Remove Configuration

- [Assign Related Content to Application Pages](#)
- [Create a New Related Content Service](#)

Manage Related Content Configuration page

**Services assigned to Application Pages**

This table describes elements used to assign and manage related content services.

<b>Content Reference</b>	Displays active content reference links to application pages that have existing related content assignments.  Click the link to display the application page and its related content.
<b>Edit Configuration</b>	Click to review or change the existing related content assignment.
<b>Remove Configuration</b>	Click to delete the existing related content assignment from the content reference.
<b>Assign Related Content to Application Pages</b>	Click to access the Select a Content Reference or Content Reference Link pages for selecting a content reference for related content assignment.
<b>Create a New Related Content Service</b>	Click to access the Define Related Content Service page.

---

## Creating Security Handlers

This section provides an overview of related content security handlers and describes how to define security handlers.

### Page Used to Create Related Content Security Handlers

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Related Content - Security Handlers	PTCS_HANDLER	PeopleTools, Portal, Related Content Service, Security Handlers	Define security handlers for related content security.

## Understanding Security Handlers

Security handlers are PeopleCode application packages that define the way security is enforced when accessing related content outside of the local node. You associate a content reference in the remote system to security logic that you write into an application class.

### **Default Security Handler**

Developers must create new security handlers and application classes or use the default security handler, PeopleToolsBasicSecurity (PTCS\_SECURITY application class). The default security handler uses these three methods:

`onAccessRequest(&MSG As Message) Returns Message;` - determines whether the current user has access to the remote related content.

`onSearchRequest(&MSG As Message) Returns Message;` - determines whether the current user has access to search the information in the related content discussion.

`onHelpInfoRequest(&MSG As Message) Returns Message;` - determines whether the related content has associated Help information.

The default security handler includes no data-level security. If you want data or row-level security, you must write the logic into the application class of the security handler.

Defining Security Handlers

Access the Related Content - Security Handlers page (PeopleTools, Portal, Related Content Service, Security Handlers).

Related Content - Security Handlers

Customize   Find   View All    First 1 of 1 Last								
	Handler Name	Default Security	*Cref Name	*Root Package ID	*Qualified Package/Class Path	*Class ID	Method	Description
1	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Related Content - Security Handlers page

- Handler Name

Enter a unique name for the handler.
- Default Security

Select this check box to use the default delivered security.
- Cref Name

Enter the name of the content reference to which you wish to apply the security logic.
- Root Package ID

Enter the name of the application package that contains the security logic.
- Qualified Package/Class Path

Enter the class path for the security logic.
- Class ID

Enter the class name for the security logic.
- Method

Enter the name of the method that invokes the security logic.
- Description

Enter a short description of the handler.



## Chapter 14

# Configuring OBIEE Report Pagelets

This chapter provides an overview of the OBIEE within the PeopleSoft framework, and discusses how to:

- Install and configure the OBIEE plug in.
- Configure OBIEE Server Setup.
- Load OBIEE report definitions.
- View OBIEE report details.
- Use Pagelet Wizard to create OBIEE report-based pagelets.

---

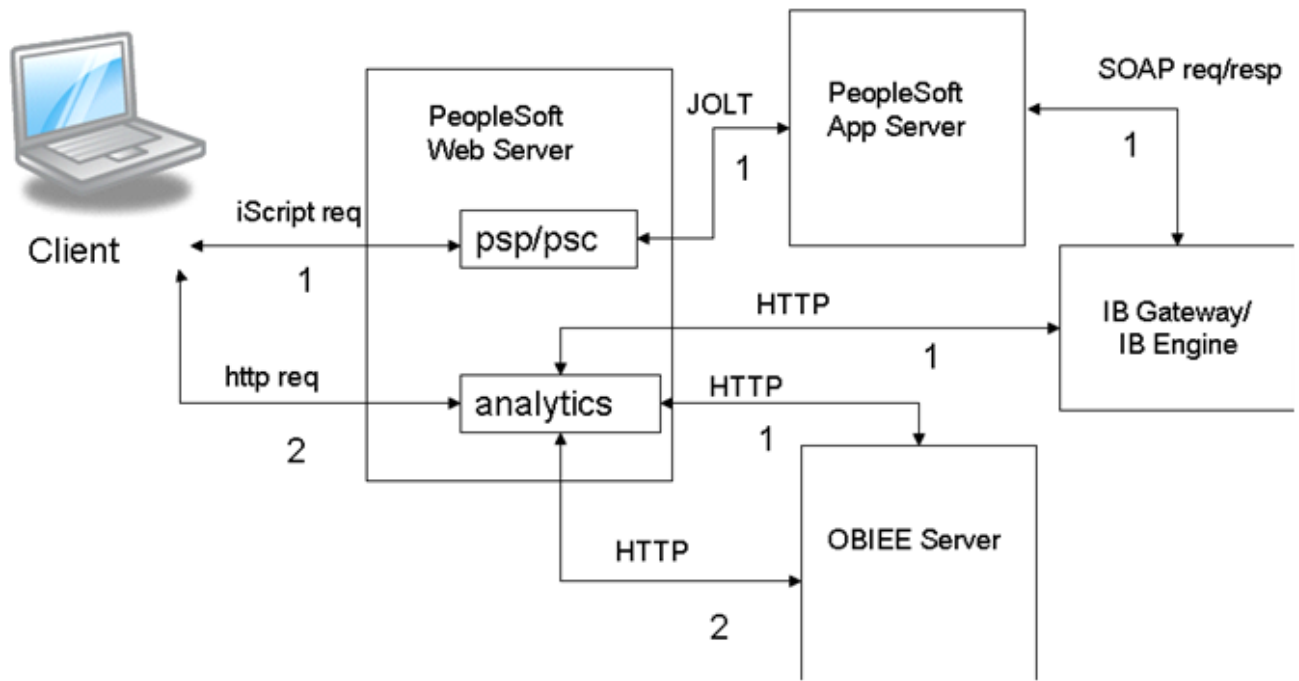
## Understanding OBIEE Integration Within the PeopleSoft Framework

The OBIEE analytics servlet enables the application developers to integrate with OBIEE analytics and display OBIEE reports as embedded pagelets on application pages, portal homepages, and as template pagelets, which are part of the Context Manager framework. This functionality defines the OBIEE reports as iScript-based pagelets in PeopleSoft applications. The iScript essentially acts as a client for the report request and sends SOAP requests to the OBIEE server.

PeopleSoft applications store the credentials of an OBIEE administrator—a user with adequate privileges—in the PeopleSoft database and use these credentials to login to the OBIEE server, impersonate a real OBIEE user, and establish a session. The OBIEE server returns a session ID for this user session which is sent as a query parameter with each SOAP/HTTP request being made to the OBIEE server. The OBIEE server responds to all valid requests. If the session ID is expired or is invalid, a login screen appears to the user. The OBIEE server provides a session ID and then appends a session ID in each HTTP request that is part of the HTML markup provided by the OBIEE server. If the user ID value in the PeopleSoft application is not found on the OBIEE server, an authorization error message appears to the user in the PeopleSoft application. Similarly, if the user exists in OBIEE but does not have authorization to the requested report, an authorization error message appears to the user.

If authentication passes, then the OBIEE server provides the HTML markup to be displayed in its entirety in the PeopleSoft application. The OBIEE report can appear on the application or portal homepage because it is wrapped in an iFrame construct to ensure that no cross domain issues occur when subsequent HTTP requests are made to the OBIEE server from the browser.

This diagram describes the architecture and the flow of information among the PeopleSoft web server, the PeopleSoft application server, the OBIEE server and the Integration Broker gateway.



OBIEE and PeopleSoft Architecture

---

**Note.** OBIEE reports are designed in the OBIEE system by the report developers. The OBIEE server generates the complete HTML markup for the report. The PeopleSoft system does not format the report; rather the PeopleSoft pagelets only display the report as designed by the developer.

---

## Installing and Configuring the OBIEE Plug In

This section provides a brief overview and discusses how to:

- Configure the web.xml file.
- Installing the OBIEE plug in.

## Understanding the OBIEE Plug In

The Analytics servlet is an Oracle Business Intelligence (BI) Presentation server plug-in which is part of Oracle BI web component. It is provided as a Web Archive File (.war) by OBIEE. This servlet communicates to the Oracle BI Presentation server using the proprietary Siebel Analytics Web (SAW) protocol. The analytics servlet acts as a client which forwards the request made from PeopleSoft applications to the Oracle BI Presentation server which listens for incoming requests. This servlet can be deployed in any J2EE Application.

---

**Note.** The analytics servlet is not part of the PIA installation process and therefore must be configured and then deployed on the web server separately from the PIA installation.

---

## Configuring the web.xml File

The web.xml file is included as part of the analytics.war file. The web server uses the web.xml file to communicate between the analytics servlet and the OBIEE server. You must configure two parameters on the web.xml file: the OBIEE server host name and the SAW server port number.

To configure the web.xml file:

1. Access and unarchive the analytics.war file.

---

**Note.** An archive file is comparable to a zip file. Use decompression software to unarchive the file.

---

2. Open the web.xml file.
3. Set the OBIEE server host name to the machine name where the OBIEE analytics server is running.
4. Set the SAW server port number to the correct value.
5. Save the web.xml file.
6. Recreate the analytics.war file and deploy it in by using the web server administrative console.

## Installing the OBIEE Plug In

To install the OBIEE plug in:

1. Copy the analytic servlet .war file to a folder in the PIA installation folder.
2. Sign on to the web server.
3. Use the web server console to deploy the analytics servlet.

---

**Note.** Deployment varies by web server product. Consult the documentation for the web server platform that you implement.

---

4. Test to determine the success of the analytics servlet deployment.

---

## Configuring OBIEE Server Setup

This section discusses:

Access the OBIEE Server Setup page (PeopleTools, Portal, OBIEE Setup, Server Setup).

## Server Setup

\*Analytics URL:

(Example: http://<machine>:<port>/<Analytic Server Name>)

\*OBIEE Server:

☒ Active Server

Description:

\*Admin User ID:

Password:

Confirm Password:

Server Setup page

Use the OBIEE Server Setup page to define the location and administrative information necessary to access the OBIEE reports.

Analytics URL	Enter the URL of the OBIEE server, for example http://ptsft12.us.oracle.com:8739/AnalyticServer1
OBIEE Server	This value defaults from the OBIEE Server name that you define when you add a server.
Active Server	Toggle this check box to make this server available or to take it offline in the PeopleSoft application.
Description	Enter a description of the server definition. This field works well for documentation purposes and when using multiple analytic servers.
Admin User ID	Enter the user ID of an OBIEE server administrator. Enter a value to ensure that you do not have to sign on in the OBIEE pagelet. <div><b>Note.</b> This user should already exist in the OBIEE server.</div>
Password	Enter the OBIEE administrator's password.
Confirm Password	Reenter the OBIEE administrator's password for confirmation.

## Loading OBIEE Report Definitions

Access the Load Report page (PeopleTools, Portal, OBIEE Setup, Load Report Definition).

## Load Report

OBIEE Server: OBIEE Server1

Report Path:

**Go**

[Root](#) > [USERS](#) > PTDMO

Folders			<a href="#">View All</a>	First	1-5 of 9
<a href="#">Name</a>	<a href="#">Owner</a>	<a href="#">Last Modified Date</a>			
<a href="#">briefingbook</a>	PTDMO	12/11/07 08:39:14Z			
<a href="#">delivers</a>	PTDMO	12/11/07 08:39:14Z			
<a href="#">filters</a>	PTDMO	12/11/07 08:39:14Z			
<a href="#">ibots</a>	PTDMO	12/11/07 08:39:14Z			
<a href="#">portal</a>	PTDMO	12/11/07 08:39:14Z			

Reports						<a href="#">View All</a>	First	1-5 of 11
<a href="#">Select</a>	<a href="#">Loaded in PeopleSoft</a>	<a href="#">Name</a>	<a href="#">Owner</a>	<a href="#">Last Modified Date</a>	<a href="#">Last Loaded Date</a>			
<input type="checkbox"/>	<input type="checkbox"/>	1	PTDMO	01/24/2008 06:16:16Z	--			
<input type="checkbox"/>	<input checked="" type="checkbox"/>	MARKET_REGION	PTDMO	01/28/2008 08:26:06Z	01/25/08 2:23AM			
<input type="checkbox"/>	<input type="checkbox"/>	obiee	PTDMO	12/12/2007 05:17:33Z	--			
<input type="checkbox"/>	<input type="checkbox"/>	PromptRpt	PTDMO	01/16/2008 09:08:07Z	--			
<input type="checkbox"/>	<input type="checkbox"/>	Rpt1	PTDMO	01/16/2008 09:13:36Z	--			

**Load Report**

Report List for Deletion					<a href="#">View All</a>	First	1 of 1
<a href="#">Select</a>	<a href="#">Report Exist In OBIEE</a>	<a href="#">Name</a>	<a href="#">Report Path</a>	<a href="#">Last Loaded Date</a>			
<input type="checkbox"/>	<input checked="" type="checkbox"/>	MARKET_REGION	/USERS/PTDMO	01/25/08 2:23AM			

**Delete Report**

### Load Report page

**Report Path** Enter the directory from which to retrieve OBIEE reports. The format should be */USERS/<PS\_UserID>/*, for example */USERS/PTDMO..*

**Go** Click this button to retrieve the OBIEE reports defined in the report path directory

### Folders

The PeopleSoft application retrieves this information from the OBIEE system.

**Name** This display-only field displays the names of folders within the root directory.

**Owner** This display-only field displays the userID of the owner as indicated in the OBIEE server.

**Last Modified Date** This display-only field displays the date the OBIEE report folder was last modified.

## Reports

<b>Select</b>	Check the check box to select the OBIEE report to load into the PeopleSoft application.
<b>Loaded in PeopleSoft</b>	This display-only field indicates whether the report has been loaded into the PeopleSoft application.
<b>Name</b>	This display-only field indicates the name of the OBIEE report as entered in the OBIEE system.
<b>Owner</b>	This display-only field displays the userID of the owner as indicated in the OBIEE server.
<b>Last Modified Date</b>	This display-only field displays the date the OBIEE report was last modified. This information is helpful in determining whether the version of the report loaded into the PeopleSoft application is the most current.
<b>Last Loaded Date</b>	This display-only field displays the date the OBIEE report was last imported into the PeopleSoft application.

## Report List for Deletion

<b>Select</b>	Check the check box to select the OBIEE report to delete from the PeopleSoft application.
<b>Report Exist In OBIEE</b>	This display-only field indicates whether the report has been deleted from the OBIEE server.  Pagelets based on reports that you remove from the OBIEE server will <i>not</i> produce results in the pagelet.
<b>Name</b>	This display-only field indicates the name of the OBIEE report as entered in the OBIEE system.

## Loading OBIEE Reports into PeopleSoft Applications

To load OBIEE reports into PeopleSoft applications:

1. Check the Select check box for the reports that you want to load into the PeopleSoft application.

---

**Note.** You may select multiple check boxes.

---

2. Click the Load Report button.

---

**Note.** Loading OBIEE reports into the PeopleSoft application does *not* remove the report from the OBIEE server.

---

## Deleting OBIEE Reports from PeopleSoft Applications

To delete OBIEE reports from PeopleSoft applications:

1. Check the Select check box for the reports that you want to deleted from the PeopleSoft application.

---

**Note.** You may select multiple check boxes.

---

2. Click the Delete Report button.

---

**Note.** Deleting OBIEE reports from the PeopleSoft application does *not* delete the report from the OBIEE system.

---

---

## Viewing OBIEE Report Details

Use the Report Details pages to view information about OBIEE reports

Access the Report Detail page (PeopleTools, Portal, OBIEE Setup, View Report Detail).

OBIEE Server:

OBIEE Server1

Report Name:

MARKET\_REGION

Report Path:

/USERS/PTDMO

Version:

2

🔍

\*Most Current Version

Description:

⬆

⬇

⬆

⬇

Last Modified Date:

01/25/08 10:23AM

Last Loaded Date:

01/25/08 2:23AM

Save

Report Detail page

<b>Version</b>	Click the prompt to select the version of the OBIEE report to view.
	<b>Note.</b> Each time you load the same report from the OBIEE server into the PeopleSoft application—regardless of whether the report has changed since last being loaded— the report version number increases by 1.

Access the Report Expression page (Click the Report Expression tab).

OBIEE Server: OBIEE Server1  
 Report Name: MARKET\_REGION  
 Report Path: /USERS/PTDMO  
 Version: 2 \*Most Current Version

**Filter**

Expression:

Report Views		View All	First	1-3 of 3
View Name				
1	compoundView!1			
2	titleView!1			
3	tableView!1			

## Report Expression page

**Filter Expression**

This display-only field indicates whether the OBIEE report has embedded filters. The *is prompted* filter indicates that the preceding field—in this case Markets.Region—is context sensitive to the target page on the PeopleSoft application.

**View Name**

This display-only field indicates the types of views created by the OBIEE report builder as stored on the OBIEE server.

See OBIEE documentation for explanations of the available types of views.

Access the Report References page (Click the Report References tab).

OBIEE Server: OBIEE Server1  
 Report Name: MARKET\_REGION  
 Report Path: /USERS/PTDMO

PeopleSoft Pagelets		View All	First	1 of 1
Pagelet ID	Pagelet Description			
MARKET_AND_REGION				

User Developed Pagelets		View All	First	1 of 1
Pagelet ID	Pagelet Description			
<input type="text"/>	<input type="text"/>			

## Report References page

This table describes the fields on the Report References page.

<b>PeopleSoft Pagelet ID</b>	This value defaults from the <i>PTPPB_PAGELET_ID</i> content reference Attribute value that you enter on the pagelet Content Ref Administration page.
<b>User Developed Pagelet ID</b>	Enter the pagelet ID of any user developed pagelets that you wish to track. This field is informational only and has no validation behind it.

Access the Xml Definition page (Click the Xml Definition tab).

OBIEE Server:

OBIEE Server1

Report Name:

MARKET\_REGION

Report Path:

/USERS/PTDMO

Version:

2    \*Most Current Version

Report XML Definition:

<?xml version="1.0"?>  
<saw:report xmlVersion="200705140"  
xmlns:saw="com.siebel.analytics.web/report/v1"  
xmlns:sawx="com.siebel.analytics.web/expression/v1"  
xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">  
  <saw:criteria subjectArea="Paint">  
    <saw:columns/>  
    <saw:filter subjectArea="Paint">  
      <sawx:expr op="prompted" xsi:type="sawx:special">

Xml Definition page

<b>Report XML Definition</b>	This field displays the XML tags and embedded data for the version of the report as it is currently loaded. Place the cursor inside the field and press Ctrl + A to select the text in its entirety.
------------------------------	--

---

## Creating OBIEE Report Pagelets

This section discusses how to:

- Use Pagelet Wizard to create OBIEE report pagelets.
- Use Context Manager to configure context sensitive OBIEE report pagelets.

### Using Pagelet Wizard to Create OBIEE Report Pagelets

This section discusses how to create OBIEE report pagelets.

1. Load the OBIEE report into the PeopleSoft application.

2. Select PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard.
3. Specify the pagelet information in step one.  
See [Chapter 11, "Using Pagelet Wizard," Step 1: Entering Pagelet Identifying Information, page 310.](#)
4. Specify *OBIEE* as the Data Type in step two. In addition, select the Report ID and Report View ID from which the pagelet will read data.

---

**Note.** If you have not loaded the OBIEE report into the PeopleSoft application, this step fails.

---

See [Chapter 11, "Using Pagelet Wizard," Selecting an OBIEE Report Data Source, page 318.](#)

5. Specify the data source parameters.  
See [Chapter 11, "Using Pagelet Wizard," Specifying OBIEE Data Source Parameters, page 333.](#)
6. Select *Passthru* as the display format.
7. Specify display options, such as additional text and a search capability.  
See [Chapter 11, "Using Pagelet Wizard," Step 5: Specifying Pagelet Display Options, page 342.](#)
8. Specify the publishing options.  
See [Chapter 11, "Using Pagelet Wizard," Specifying Publication Options, page 367.](#)

## **Part 5**

# **Working with Portlet Technologies**

### **Chapter 15**

**Understanding Web Services For Remote Portlets (WSRP)**

### **Chapter 16**

**Configuring WS-Security For WSRP Consumption and Production**

### **Chapter 17**

**Using WSRP to Consume and Produce Remote Portlets**

### **Chapter 18**

**Working with JSR 168 Portlets**



## Chapter 15

# Understanding Web Services For Remote Portlets (WSRP)

This chapter provides an overview of Web Services for Remote Portlets (WSRP), a terminology list, and a scenario of how to implement portlet producers and consumers.

---

## Understanding WSRP

WSRP is a web services protocol, which is used to bring together content and interactive web applications from remote sources. WSRP incorporates standards such as XML, SOAP, and WSDL to serve as a foundation, while allowing for the implementation of evolving standards.

A WSDL document is an XML file that contains information about the interface, semantics and other details of a call to a web service. When someone wants to use a service, they request a WSDL document to find out the location of the service, the associated function calls and how to access them. They then use this information in the WSDL to form a Simple Object Access Protocol (SOAP) request and send it via HTTP to the external system endpoint.

PeopleSoft provides support for:

- consuming portlets using WSRP technology.
- producing WSRP portlets through Pagelet Wizard.
- producing WSRP portlets for PeopleSoft pages and iScripts.
- implementing WS-Security for single signon with third party portals that support WSRP and WS-Security.

The terms *Producer* and *Consumer* are used to describe parties implementing the WSRP protocol.

*Producer:* A web service that offers one or more portlets and implements various WSRP interfaces. A producer may offer just one portlet, or may provide a container for deploying and managing several portlets.

*Consumer:* A web service client that invokes producer-offered WSRP web services and provides an environment for users to interact with portlets offered by one or more producers.

Most PeopleSoft pages, portlets, and iScripts are WSRP-compliant and available for consumption in WSRP-compliant portals; this refers to existing definitions and newly created definitions. While these PeopleSoft items, by default, are available for WSRP consumption, an administrator controls which items are actually produced, or exposed, for WSRP consumption.

This functionality enables you to incorporate PeopleSoft application pages, pagelets, and iScripts into a portal of your choice. When users in the non-PeopleSoft portal interact with the WSRP portlet, depending on how the portlet is configured, users can interact directly within the portlet or they can be transferred to the PeopleSoft portal where the portlet originated.

To enable components for WSRP production, you select the WSRP Compliant option on the Internet tab of the Component Properties dialog box.

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Application Designer Developer's Guide*, "Creating Component Definitions," Setting Internet Properties.

## **WSRP Interfaces and Operations**

A producer must implement the following WSRP interfaces:

*Service Description Interface:* Provides metadata of itself and the list of portlets it offers. The Consumer invokes the `getServiceDescription` operation of this interface to obtain the metadata.

*Markup Interface:* Generates markup and processes interaction requests. The Consumer invokes the `getMarkup` operation of this interface to obtain the portlets markup. They then invoke the `performBlockingInteraction` operation to process users interactions to the Producer.

WSRP also specifies the following optional interfaces:

*Registration Interface:* Registration interface provides an in-band mechanism for a Consumer to register with a Producer. It lets the Producer customize its behavior for each Consumer based on registration information. WSRP also allows out-of-band registrations and no registration.

*Portlet Management Interface:* Allows Consumers to clone or destroy portlets, as well as customize portlets by changing any associated properties.

---

**Note.** The registration interface and portlet management interfaces are not used by the PeopleSoft producer. Consumers are therefore not required to register with the PeopleSoft producer.

---

## **WS-Security**

By implementing the WS-Security standard, PeopleSoft provides the ability to leverage emerging XML security technologies to address web services security requirements. WS-Security provides:

- A way for applications to construct secure SOAP message exchanges.
- A general-purpose mechanism for associating security tokens with SOAP messages.
- XML Message integrity and confidentiality.

By providing WS-Security capabilities, you can leverage the standard set of SOAP extensions, that you use when building secure web services, to implement message content integrity and confidentiality. WS-Security provides a way to insert and convey security tokens in SOAP messages. The ability to leverage WS-Security standards provides for better interoperability and improved usability, enabling the implementation of robust security within a WSRP-capable environment. The solutions being provided through the PeopleSoft WS-Security implementation include:

- Single Signon solution between WSRP consumer and producer.

The web services consumer passes the appropriate identification to a producer as part of the SOAP message, so that producer can verify the identity in order to execute requested web services on behalf of the user without requiring a user to log in. Single signon between web services consumer and producer feature is currently supported in PeopleSoft WSRP Portal, PeopleSoft Integration Broker, and BPEL product.

- SOAP message integrity. Ensuring that messages have not been tampered with
- SOAP message confidentiality. Guaranteeing that messages are protected against eavesdroppers.

WS-Security UsernameToken Profile defines a standard way to associate user ID and password information in the SOAP messaging for web services interoperability.

The Security Assertion Markup Language (SAML) Token defines assertions, protocols, bindings, and profiles.

The PeopleSoft portal solution provides support for WS-Security for single-signon with third party applications—limited to user authentication using user name/password or user authentication using user name and digital signature through the use of Web Services Security: Username Token Profile and SAML Token.

---

**Note.** PeopleSoft provides multiple levels of security for WSRP. These levels, or options, are discussed in the following chapter. PeopleSoft recommends that you determine the level that is appropriate for your needs before implementing WS-Security. Using SSL connections to secure transmissions may be sufficient.

---

See [Chapter 10, "Configuring the Portal Environment," Improving Same-Server Performance Under SSL, page 264](#) and [Chapter 16, "Configuring WS-Security For WSRP Consumption and Production," page 497](#).

---

## Terminology

<b>ASF</b>	Apache Software Foundation.
<b>Java Portlet</b>	A java software module that conforms to the Portlet API.
<b>JCP</b>	<i>Java Community Process</i> : Established for the development of java technology.
<b>JSR</b>	<i>Java Specification Request</i> : Each submission to the JCP gets assigned a unique JSR number.
<b>JSR 168</b>	The JCP specification that describes the Portlet API.
<b>SAML</b>	<p><i>Security Assertion Markup Language</i>: An XML standard for exchanging authentication and authorization data between entities.</p> <p>SAML provides a standard security token—a SAML assertion—that can be used with standard web services security frameworks.</p> <p>See <a href="http://www.oasis-open.org/glossary/index.php">http://www.oasis-open.org/glossary/index.php</a></p>

<b>SOAP</b>	<i>Simple Object Access Protocol</i> : An XML-based messaging protocol framework for building and exchanging distributed, structured information in a decentralized and distributed environment.
<b>WSDL</b>	<i>Web Service Description Language</i> : A XML language for describing web services; it defines the core language that can be used to describe web services based on what the services offer.
<b>WSRP</b>	<i>Web Services for Remote Portlets</i> : A web services protocol for bringing together content and interactive web applications from remote sources.
<b>WSS</b>	<i>Web Services Security Language</i> : Supports security mechanisms, each using implementation and language-neutral XML formats. They include the use of: <ul style="list-style-type: none"><li>• XML signature to provide SOAP message integrity.</li><li>• Use of XML encryption to provide SOAP message confidentiality.</li><li>• Attaching and referencing security tokens and associating signatures with security tokens.</li></ul>
<b>WSRP4J</b>	The ASF WSRP reference implementation project.
<b>XML</b>	<i>Extensible Markup Language</i> : Describes data and focuses on what data is. XML is designed to structure, store, and send information.

---

## Implementing WSRP Protocol Scenario

This scenario discusses interactions between two companies; PeopleSoft, Inc. (a WSRP producer), and Kane Consulting (a WSRP Consumer).

In this example, Kane Consulting is an online company, providing personalized financial services to clients by subscription. PeopleSoft, Inc. would like to host a number of financial applications, including a web based inventory planning application. Kane Consulting would like to offer this application to its clients via its web pages.

Without WSRP, in order to offer the inventory planning application to clients, PeopleSoft, Inc. and Kane Consulting must agree on the following procedure:

1. PeopleSoft, Inc. makes the metadata of the inventory planning application available to Kane Consulting. Kane Consulting uses this metadata to create a page that clients can use to manage their plans.
2. A client visits Kane Consulting's web site and clicks on a link to the inventory planning application.
3. Kane Consulting then transmits a request to PeopleSoft, Inc. to obtain the initial view of the application. PeopleSoft Inc. responds by returning HTML markup that represents the first page of the application.
4. Kane Consulting processes the returned markup and prepares it for conversion. If the returned markup has links, Kane Consulting transforms the markup such that when they are activated they return to the Kane Consulting web site.

5. Kane Consulting converts the markup into a web page, writes it into the response of the browser's connection, and transmits the page to the client's browser.
6. The client reviews the page, and finds a form to submit a new vendor ID. The client then fills in the ID number and other details and submits the entry.
7. Kane Consulting receives the request containing the new data. Upon determining that the request is for the inventory planning application, Kane Consulting transmits another request to PeopleSoft, Inc. to process the client transaction.
8. PeopleSoft, Inc. processes the transaction, adds the vendor ID to the clients plan and returns a new state for the plan.
9. Kane Consulting then sends a request to get the changed markup based on the current state of the plan. PeopleSoft Inc. generates the markup and returns.
10. Kane Consulting then repeats steps 4 and 5.
11. The client receives a new page containing the updated plan.

Instead of developing a protocol to achieve the above preceding procedure, PeopleSoft Inc. and Kane Consulting can use WSRP as the protocol. PeopleSoft Inc. is a WSRP Producer offering portlets, and Kane Consulting is a WSRP Consumer consuming portlets and aggregating portlets for clients to access aggregated portlet pages. The inventory planning application is a portlet offered by the WSRP Producer.

To implement the preceding procedure PeopleSoft Inc. and Kane Consulting uses WSRP to define various interactions, with PeopleSoft Inc. implementing the following required WSRP interfaces and operations.

*Service Description Interface:* Provides metadata of itself and the list of portlets it offers. Kane Consulting invokes the `getServiceDescription` operation of this interface to obtain this metadata. (Step 1 of the procedure).

*Markup Interface:* To generate markup and to process requests, PeopleSoft Inc. implements the markup interface specified by WSRP. Kane Consulting, invokes the `getMarkup` operation to obtain the portlet's markup (steps 3 and 9), and invokes `performBlockingInteraction` to generate the client's interactions to PeopleSoft Inc. (Step 7).

By implementing the preceding interfaces, and agreeing to conform to WSRP, both PeopleSoft Inc. and Kane Consulting can use a standard process to offer and consume portlets. In addition PeopleSoft Inc. can offer the same portlets to company X as long as X adheres to WSRP, and Kane Consulting can consume additional portlets offered by company Y provided Y also implements WSRP interfaces.

---

## WSRP and Server Cluster Configuration Considerations

Support for web server clustering with session replication is supported. However, the open source products Enterprise PeopleSoft uses for WSRP functionality, Apache Pluto and WSRP4J, do not support serialization. If you intend to implement web server clustering with session replication *and* WSRP functionality, you must set up two separate web server environments. The web server environment servicing WSRP functionality cannot have session replication enabled.



## Chapter 16

# Configuring WS-Security For WSRP Consumption and Production

This chapter provides overviews of determining security requirements and WS-Security, and discusses how to:

- Set up WSRP security options.
- Set up SAML.
- Configure WS-Security For WSRP Producers.
- Define Nodes with WS-Security.

---

**Important!** The procedures for configuring WSRP consumption and production must be performed in the order set out in this chapter.

---

---

## Determining Security Requirements

Before implementing WSRP, it is important that you first determine the level of security you require. If you are consuming remote portlets, the level you select *must* be the same as the producer. If you are producing portlets, consider whether you need to secure only the transmission channel, or the transmission channel and the messages. You should also consider how security will affect the performance of your servers.

Including the *No Security* option, there are six types of token security options for WS-Security (set in the server-config.wsdd file):

- *None*. No security token required.
- *Username Token*. The Username Token is in clear text format.

---

**Important!** If you require a secure channel between the producer and consumer, the .HTTPS protocol *must* be used.

---

- *Authentication Token as Username Token with full security*. Username Token is encrypted and messages are digitally signed.

---

**Note.** For this option, either the HTTP or HTTPS protocol can be used to communicate between the producer and consumer.

---

- *Username token with no password and digital signature*.

- *Authentication Token as SAML Token.* The SAML token messages are digitally signed.

---

**Note.** Security Assertion Markup Language (SAML) Token messages are digitally signed by default.

---

- *Authentication Token as SAML Token with full security.* The SAML token is encrypted and messages are digitally signed.

---

**Important!** The security option used by the producer *must* be an exact match to the security option used by the consumer.

---

---

## Understanding the PeopleSoft WS-Security for WSRP

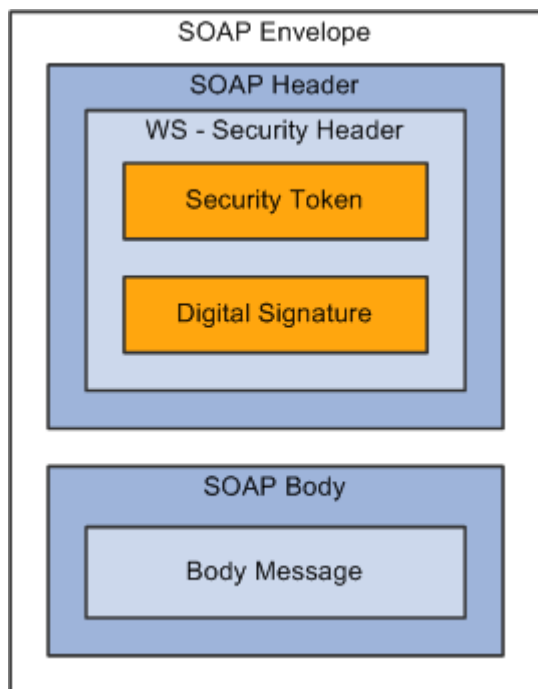
WSRP involves passing of SOAP messages between the WSRP consumer and producer. To safely utilize WSRP, PeopleSoft provides message-level security between the consumer and the producer by incorporating WS-Security.

WS-Security is an extension to the concept of the SOAP envelope header that enables applications to construct secure SOAP message exchanges. It also provides a means for associating security tokens with messages.

WS-Security provides three main mechanisms:

- *Integration solution between the WSRP consumer and producer:* The consumer passes their identification to the producer as part of the SOAP message so that the producer can verify the identity and produce WSRP content without requiring the user to log in.
- *Message integrity:* Ensures that messages have not been tampered with.
- *Message confidentiality:* Guarantees that messages are protected.

This figure shows the SOAP envelope, SOAP header and SOAP body and how WS-Security embeds the security token in the SOAP messages:



WS-Security SOAP Message Structure

When the PeopleSoft portal is a WSRP consumer, the user ID and password of the user signed into the portal is placed into a Username or SAML Token in the SOAP header, and the WSRP portlet consumes it. It is presented to each portlet during the initial markup request. The security handler performs WS-Security token generation, digital signature generation, and token encryption, before the PeopleSoft system sends the SOAP request message and the WS-Security SOAP header.

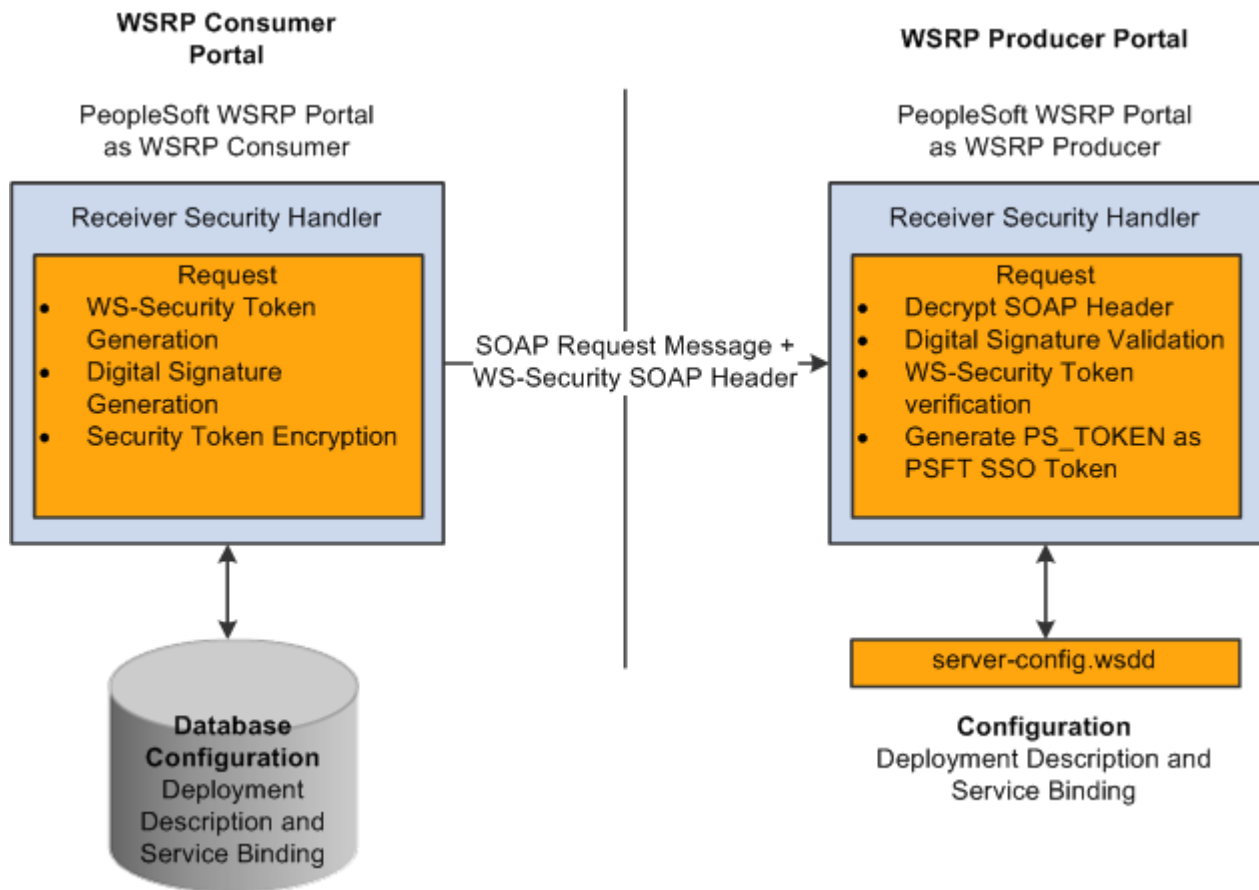
---

**Important!** The PeopleSoft portal authentication information (both the user ID and password), must be the same as the producer's.

---

When PeopleSoft is the producing system, the consuming system sends PeopleSoft the SOAP request message and the WS-Security SOAP header. The portal accepts the Username Token in the WS-Security header from the remote portal, assuming that the ID and password are acceptable to PeopleSoft. The receiver security handler decrypts the SOAP header, validates the digital signature, verifies the WS-Security token, and generates a PS\_TOKEN cookie, the PeopleSoft authentication token.

This diagram shows PeopleSoft as both WSRP consumer and WSRP producer.



PeopleSoft Portal as WSRP Consumer and Producer

The WS-Security support provided by PeopleSoft applications includes

- Username Token Profile.
- Security Assertion Markup Language (SAML) Token Profile.

---

**Note.** PeopleSoft applications support SAML 1.1.

---

### ***UserNameToken Security***

With Username Token support, a consumer can supply a UsernameToken as a means of identifying the requestor by username, and optionally using a password to authenticate that identity to the web service provider.

This is an XML example of a UserName Token with password:

---

**Note.** The password is in cleartext.

---

```
<soapenv:Header>
  <wsse:Security soapenv:mustUnderstand="1" xmlns:wsse="http://docs.oasis-open.org/
/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
    <wsse:UsernameToken>
      <wsse:Username>QEDMO</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
username-token-profile-1.0#PasswordText">QEDMO</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</soapenv:Header>
```

This is an XML example of a UserName Token with Digital Signature:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
<soapenv:Header>
<wsse:Security soapenv:mustUnderstand="1" xmlns:wsse="http://docs.oasis-open.org/
wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"><ds:Signature xmlns:ds="
http://www.w3.org/2000/09/xmldsig#">
<ds:SignedInfo>
<ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
<ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
<ds:Reference URI="#id-18871350">
<ds:Transforms>
<ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
</ds:Transforms>
<ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
<ds:DigestValue>Do+wViC4mSHBWYmRMWKArdF8xmU=</ds:DigestValue>
</ds:Reference>
<ds:Reference URI="#id-11549664">
<ds:Transforms>
<ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
</ds:Transforms>
<ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
<ds:DigestValue>4NVgNpjXl6B4+Wrmw+7bnGfeJSS=</ds:DigestValue>
</ds:Reference>
</ds:SignedInfo>
<ds:SignatureValue>
QFXQuStu8sLAvsDiGNaZHUVWBIES9GpgY6fGb9KLP209hzNi+cPp3NahsFlmVxcXzCjrui2Vc3vq
wLtYA9kaFsm7EMY8TdJC9cmiepr07pR2iXXMmDYyp0ldOQkz+3SeBg9F2qPOBAg/jTrWgppys2rS
ES9pV6OIKRk0try8BcU=
</ds:SignatureValue>
<ds:KeyInfo Id="KeyId-1049443">
<wsse:SecurityTokenReference wsu:Id="STRId-14721926" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"><ds:X509Data>
<ds:X509IssuerSerial>
<ds:X509IssuerName>CN=PeopleTools TEST root CA,DC=peoplesoft,DC=com,OU=PeopleTools=
Development,O=PeopleSoft Inc,L=Pleasanton,ST=CA,C=US</ds:X509IssuerName>
<ds:X509SerialNumber>566474176348341487536757</ds:X509SerialNumber>
</ds:X509IssuerSerial>
</ds:X509Data></wsse:SecurityTokenReference>
</ds:KeyInfo>
</ds:Signature>
<wsse:UsernameToken wsu:Id="id-11549664" xmlns:wsu="http://docs.oasis-open.org/wss-
/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"><wsse:Username>QEDMO</wsse:
Username><wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">QEDMO</wsse:Password></wsse:Username-
Token></wsse:Security></soapenv:Header>    <soapenv:Body wsu:Id="id-18871350" xmlns:
=>
=>
=>
=>
=>
=>
=>
wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd">    <ns1:testMethod xmlns:ns1="http://axis/service/security/test9/Log-
TestService9" />    </soapenv:Body></soapenv:Envelope>

```

## Security Assertion Markup Language (SAML) Token Security

A SAML Token makes statements about a principal. All SAML tokens include the following common information:

- Issuer ID
- Issuance timestamp
- NotBefore or NotOnOfAfter conditions
- Subject
- Name
- Subject confirmation
- Conditions under which the assertion is valid

This is an XML example of a SAML token:

```
<Assertion AssertionID="d9aeaa4c1126df5ee0c6df64fdf961b1" IssueInstant="2008-05-14T18:18:47.246Z" Issuer=".peoplesoft.com" MajorVersion="1" MinorVersion="1"
  xmlns="urn:oasis:names:tc:SAML:1.0:assertion" xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
=>
=>
=>
=>
=>
=>
=>
1.0:assertion" xmlns:samlp="urn:oasis:names:tc:SAML:1.0:protocol">
  <Conditions NotBefore="2008-05-14T18:18:47.184Z" NotOnOrAfter="2008-05-14T18:28:47.184Z"/>
  <AuthenticationStatement AuthenticationInstant="2008-05-14T18:18:47.215Z"
  AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:password">
    <Subject>
      <NameIdentifier NameQualifier=".peoplesoft.com">QEDMO</NameIdentifier>
      <SubjectConfirmation>
        <ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:sender-vouches</
/ConfirmationMethod>
      </SubjectConfirmation>
    </Subject>
  </AuthenticationStatement>
</Assertion>
```

This is an XML example of a SAML token with Digital Signature

```

- <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
- <soapenv:Header>
- <wsse:Security soapenv:mustUnderstand="1" xmlns:wsse="http://docs.oasis-open.org/
/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
- <Assertion AssertionID="ede14876b3389b653824f0456e07676d" IssueInstant="2008-05-=>
13T22:54:17.417Z" Issuer=".peoplesoft.com" MajorVersion="1" MinorVersion="1"=>
xmlns="urn:oasis:names:tc:SAML:1.0:assertion" xmlns:saml="urn:oasis:names:tc:SAML:=>
=>
=>
=>
=>
1.0:assertion" xmlns:samlp="urn:oasis:names:tc:SAML:1.0:protocol">
  <Conditions NotBefore="2008-05-13T22:54:17.386Z" NotOnOrAfter="2008-05-13T23:04:=>
17.386Z" />
- <AuthenticationStatement AuthenticationInstant="2008-05-13T22:54:17.401Z"=>
AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:password">
- <Subject>
  <NameIdentifier NameQualifier=".peoplesoft.com">QEDMO</NameIdentifier>
- <SubjectConfirmation>
  <ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:sender-vouches</Confirmation=>
Method>
  </SubjectConfirmation>
  </Subject>
  </AuthenticationStatement>
  </Assertion>
- <wsse:SecurityTokenReference wsu:Id="STRSAMLId-11733267" xmlns:wsu="http:=>
//docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
  <wsse:Reference URI="#ede14876b3389b653824f0456e07676d" ValueType="http:=>
//docs.oasis-open.org/wss/2004/XX/oasis-2004XX-wss-saml-token-profile-=>
1.0#SAMLAssertion-1.1" />
  </wsse:SecurityTokenReference>
  <wsse:BinarySecurityToken EncodingType="http://docs.oasis-open.org/wss/2004/01=>
/oasis-200401-wss-soap-message-security-1.0#Base64Binary" ValueType="http:=>
//docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3"=>
wsu:Id="CertId-28365241" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-=>
200401-wss-wssecurity-utility-1.0.xsd">MIIEIzCCBACgAwIBAgIKQrSKy=>
QAAAAAL+DANBgkqhkiG9w0BAQUFADCBvJELMAkGALUEBhMCVVMxCzAJBgNVBAGTAKNBMRMwEQYDVQQHEwp=>
QbGVhc2FudG9uMRcwFQYDVQQKEw5QZW9wbGVUb2Z0IEluYzEgMB4GALUECzMXUGVvcGx1VG9vbHMg=>
RGV2ZWxvcGllbnQxEzARBGoJkiaJk/IsZAEZFgNjb20xGjYAYBGoJkiaJk/IsZAEZFgppwZW9wb=>
GVzb2Z0MSEwHwYDVQQDEzhQZW9wbGVUb29scyBURVNUIHJvb3QgQ0EwHhcNMDCwODE2MTc1MDIzWhc=>
NMDgwODE2MTgwMDIzWjBxMQswCQYDVQQGEwJVUzETMBEGALUECBMKQ2FsaWZvcn5pYTEtMBEGALUEBx=>
MKUGxlYXNhbnRvbJEPMA0GALUEChMGT3JhY2x1MRQwEgYDVQQLEwtQZW9wbGVUb29sczERMA8GALUEAw=>
IcWVfaWJ0Z3QwgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMrvvgIBLkotkSm+tnbxjyNQ1=>
Ti3h3p8h44WhIj1Ikdvhtwk890doxdF1HkDaX0Zz7/9IcW4wc5l3z/C/r0kchcq95ToRxKZ5UC50Sx2wb=>
T9/GL6GVUP/EBtGqcaZSeE4G9OHivMeXxe7opb3InyyAxU/dbuLj/bMpOvABgbmdhAgMBAAGjggHm=>
MIIB4jAdBgNVHQ4EFgQUZ1KpPrdYQY+mEdNk/YWKe4Iqn+wwHwYDVR0jBBGwFoAU/jeXdgwdjCBw=>
II0g3c+t5n06BswwgawGALUdHwSBpDCBoTCBnqCBm6CBmIZMaHR0cDovL3B0bnRhc2EYLnBlb3BsZXNvZn=>
QuY29tL0NlcnRfbnJvbGwvUGVvcGx1VG9vbHMlMjBURVNUIHJvb3QgQ0EwHhcNMDCwODE2MTc1MDIzWhc=>
L1xccHRudGFzMTIucGVvcGxlc29mdC5jb21cQ2VydEVucm9sbFwQZW9wbGVUb29scyBURVNUIHJvb3Qg=>
Q0EuY3J3SjMIHwBggrBgEFBQcBAQSB4zCB4DBwBggrBgEFBQcwAoZkaHR0cDovL3B0bnRhc2EYLnBlb3Bs=>
ZXNvZnQuY29tL0NlcnRfbnJvbGwvUGVvcGxlc29mdC5jb21fUGVvcGx1VG9vbHMlMjBURVNUIHJvb3Qg=>
BURVNUIHJvb3QgQ0EwHhcNMDCwODE2MTc1MDIzWhcL1xccHRudGFzMTIucGVvcGxlc29mdC5jb21cQ2VydEVucm9sbFwQZW9wbGVUb29scy=>
Gxlc29mdC5jb21cQ2VydEVucm9sbFwQZW9wbGVUb29scy=>

```

```
BURVNUIHJvb3QgQ0EuY3J0MA0GCSqGSib3DQEBBQUAA4GBABaZwo6xhkZFRbESi3ICewrBTKFjtDbmJv
Vq=>
BaB0pConBMRGDJ0bQf9Rwo6/Ucm/BoUEEP/dzBkLYM0NEuEqoQvF4ZHRD73qwNV9CCUHU3nlwfn=>
L5K54qrda4V2CoBvgpHEU7EVdt47YV2E8HUAUfyDaXZ0prRMB6I2KtKaaYBNI</wsse:BinarySecurity=
Token>
- <ds:Signature Id="Signature-22949069" xmlns:ds="http://www.w3.org/2000/09=>
/xmlldsig#">
- <ds:SignedInfo>
  <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-cl4n#" =>
/>
  <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmlldsig#rsa-shal" />
- <ds:Reference URI="#STRSAMLId-11733267">
- <ds:Transforms>
- <ds:Transform Algorithm="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss->
=>
=>
=>
=>
=>
=>
=>
soap-message-security-1.0#STR-Transform">
- <wsse:TransformationParameters>
  <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-cl4n#" =>
/>
  </wsse:TransformationParameters>
</ds:Transform>
</ds:Transforms>
<ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmlldsig#sha1" />
<ds:DigestValue>TK2a7xf+ldF9MkiIXYut8g5RG+A=</ds:DigestValue>
</ds:Reference>
- <ds:Reference URI="#id-78219">
- <ds:Transforms>
  <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-cl4n#" />
</ds:Transforms>
  <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmlldsig#sha1" />
  <ds:DigestValue>wTVh0phi6NrTKDWnyXBx/WNCw68=</ds:DigestValue>
</ds:Reference>
</ds:SignedInfo>
  <ds:SignatureValue>LGWmXxak++duS8IfY+/4BphfPJW+Ka6a8DxUfYmUGUZ57d1HOmTs0fLR0Roqd=>
IOKJtSD33qRyU6p 7lufZXsRoiJD5iEUJR+El7KgBEmpFPV5hDx2a+dnHN8Zd9A1DRh7qzrlwKcRpd=>
BDdoS2mJnqjcz mlLOU6aPQqKAjgRtZtg=</ds:SignatureValue>
- <ds:KeyInfo Id="KeyId-15595312">
- <wsse:SecurityTokenReference wsu:Id="STRId-24840600" xmlns:wsu="http:>
//docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
  <wsse:Reference URI="#CertId-28365241" ValueType="http://docs.oasis-open.org/wss=>
/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3" />
  </wsse:SecurityTokenReference>
</ds:KeyInfo>
</ds:Signature>
</wsse:Security>
</soapenv:Header>
- <soapenv:Body wsu:Id="id-78219" xmlns:wsu="http://docs.oasis-open.org/wss/2004=>
/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
  <nsl:testMethod xmlns:nsl="http://axis/service/security/test9/LogTestService9" =>
/>
</soapenv:Body>
</soapenv:Envelope>
```

Note these points about PeopleSoft SAML assertions

- The PeopleSoft SAML token is concerned with the authentication statement *only*.
- The PeopleSoft SAML token supports SAML with digital signature and encryption. SAML tokens without digital signatures are not supported.
- The PeopleSoft SAML profile of WSS: SOAP Message Security requires that systems support *sender-voucher* methods of subject confirmation.
- The SAML Assertion validity or condition by default is set to 10 minutes. However, you can override the default time by adding `org.apache.ws.security.saml.AssertValidMins=15` in the `wssSAML.properties` file which is located in the `\\WEB-INF\classes\wssSAML.properties` directory.

---

## Setting Up WSRP Security Options

This section discusses how to:

- Use HTTPS protocol to communicate between producer and consumer.
- Install certificates for encryption and signature.

## Using HTTPS Protocol to Communicate Between Producer and Consumer

If you choose the security option "Authentication token as Username token in clear text format", it is recommended to use HTTPS to protect the username and password. To use HTTPS protocol both the producer and consumer must first configure and enable SSL for HTTPS on their web servers.

Before the consumer imports the producer `\pspc\wsdl\wsrp4j_service.wsdl`, the producer must modify the file by setting it to HTTPS mode and changing the port number from the HTTP port number to the HTTPS port.

---

**Note.** The system automatically inserts the hostname and port number using the values entered during installation.

---

Example:

```

<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="urn:oasis:names:tc:wsrp:v1:wsdl"
  xmlns:bind="urn:oasis:names:tc:wsrp:v1:bind"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">

  <import namespace="urn:oasis:names:tc:wsrp:v1:bind"
    location="wsrp_v1_bindings.wsdl"/>

  <wsdl:service name="WSRPService">
    <wsdl:port binding="bind:WSRP_v1_Markup_Binding_SOAP"=>
name="WSRPBaseService">
      <soap:address location=>
"https://<Producer Hostname>:<SSL port>/pspc/wsrp4j/WSRPBaseService"/>
    </wsdl:port>
    <wsdl:port binding="bind:WSRP_v1_ServiceDescription_Binding_SOAP"=>
name="WSRPServiceDescriptionService">
      <soap:address location="https://<Producer Hostname>:<SSL port>/pspc/wsrp4j/WSRPServiceDescriptionService"/>
    </wsdl:port>
    </wsdl:service>
  </wsdl:definitions>

```

---

**Note.** Where <Producer Hostname> is the hostname of the Producer web server, <SSL port> is the SSL port number of the Producer web server.

---

If you are using an SSL accelerator in your PeopleSoft system, and you want to use the HTTPS protocol for WSRP to communicate with PeopleSoft Pure Internet Architecture, you need to use the following properties file:

<PIA\_HOME>\websrv\peoplesoft\applications\peoplesoft\pspc\WEB-INF\config\services\PIAService.properties.

---

**Note.** This is similar to using Web Profile's Default Addressing fields.

---

See [Chapter 10, "Configuring the Portal Environment," Configuring Web Profiles, page 216.](#)

In this properties file, specify the default protocol, host, and port to override the incoming request protocol, host and port. The settings WSRP checks in the PIAService.properties file are:

<b>usePIAConfig = true</b>	Flag to check if settings in the PIAService.properties file should be used or not.  If you set usePIAConfig to be true, the incoming request is overridden by values set in the PIAService.properties file. Otherwise, the system uses the incoming request values for protocol (scheme), host, and port.
<b>portalHost=&lt;your webserver host&gt;</b>	The default host to use instead.
<b>portalPort=&lt;your port&gt;</b>	The default port to use instead.
<b>SSLToPIA=true</b>	If you set SSLToPIA to be true, the system will use HTTPS, otherwise it will use the HTTP scheme.

**See Also**

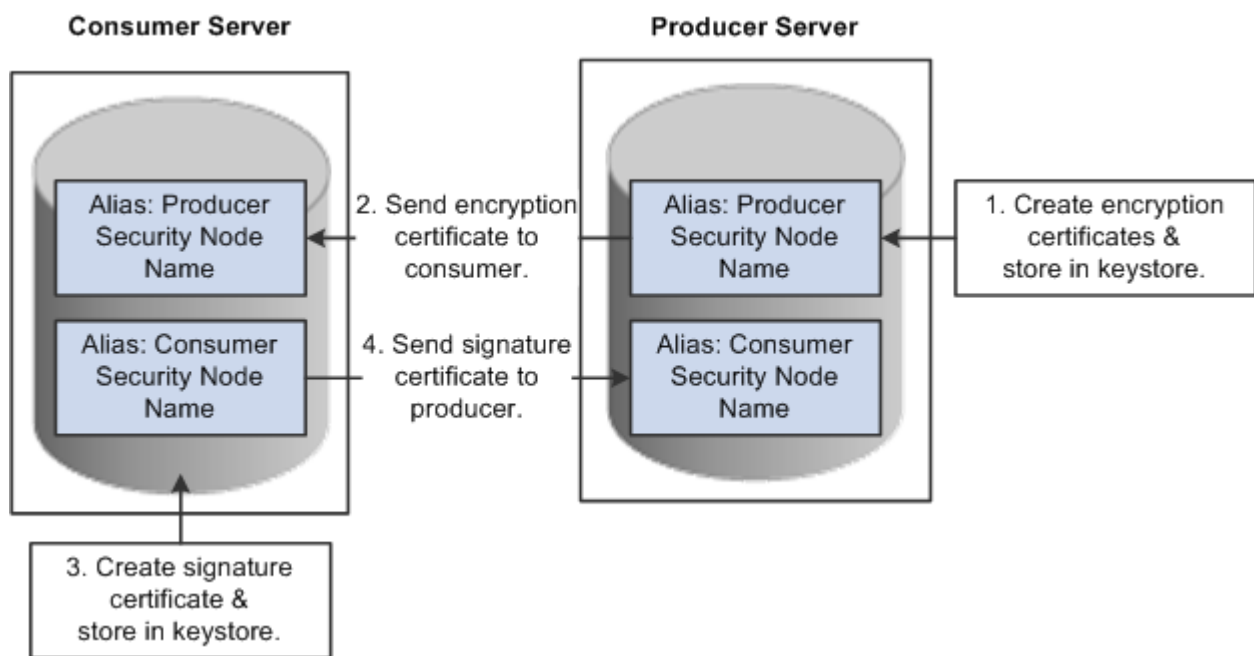
*Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*, "Working with Oracle WebLogic," Implementing WebLogic SSL Keys and Certificates

*Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*, "Working with IBM WebSphere," Setting Up SSL For WebSphere

**Installing Certificates For Encryption and Signature**

When you select the *Authentication token as Username token with full security* security option, both the producer and the consumer must install the certificates that are used to enable the encryption and signature.

This diagram illustrates the process flow for installing certificates and enabling encryption and signature.



Process for installing certificates and enabling encryption and signature

First use the following reference for instructions on how to create and request your encryption and signature certificates.

See *Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*, "Working with Oracle WebLogic," Implementing WebLogic SSL Keys and Certificates and *Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*, "Working with IBM WebSphere," Setting Up SSL For WebSphere.

**Enabling Encryption***Producer:*

The encryption key and certificate are created on the producer and stored in the web server predefined keystore.

---

**Note.** Make sure when you create the entry to hold your encryption keys and certificate in your keystore, the entry's alias name must be the security node name of your Producer server.

---

Send the encryption certificate to the consumer. This can be done using an out-of-band method, such as file copy or email.

*Consumer:*

Install the encryption certificate on the Consumer server by running `pskeymanager.cmd -import`.

---

**Note.** Make sure when you import the certificate, the entry's alias name must be the security node name of your Producer server.

---

### ***Enabling Message Signature***

*Consumer:*

Create the signature key and certificate on the consumer and store in the web server predefined keystore.

---

**Note.** Make sure when you create the entry to hold your signature keys and certificate in your keystore, the entry's alias name must be the security node name of your Consumer server.

---

Send the signature certificate to the producer. This can be done using an out-of-band method, such as file copy or email.

*Producer:*

Install the signature certificate on the Producer server by running `pskeymanager.cmd -import`.

---

**Note.** Make sure when you import the certificate, the entry's alias name must be the security node name of your Producer server.

---

### ***Encrypting Keystore Password Using PSCipher***

A copy of the `wss.properties` file, which contains PKI java keystore information, such as keystore file location and password for the keystore can be found on both the Consumer and Producer servers.

The location of the `wss.properties` file on the Consumer web server is:

...\PORTAL\WEB-INF\classes

The location of the `wss.properties` file on the Producer web server is:

....\pspc\WEB-INF\classes

The following is the sample `wss.properties` file:

```
org.apache.ws.security.crypto.provider=org.apache.ws.security.components.
crypto.Merlin
org.apache.ws.security.crypto.merlin.keystore.type=jks
org.apache.ws.security.crypto.merlin.keystore.password=UWZzB57U6SE=
org.apache.ws.security.crypto.merlin.file=keystore/pskey
```

To define the location of the keystore file, the Portal Administrator needs to define the proper keystore file to replace above `org.apache.ws.security.crypto.merlin.file` property.

To encrypt the keystore password using Java program PSCipher:

1. Run PSCipher.bat (.sh for UNIX) to encrypt the password.

```
% PSCipher.bat <password>
```

For example, PSCipher interop provides output as UWZzB57U6SE=

2. Write down the output of PSCipher and then update the encrypted password in the wss.properties file for `org.apache.ws.security.crypto.merlin.keystore.password`

See *Enterprise PeopleTools 8.50 PeopleBook: Security Administration*, "Encrypting Text With PSCipher."

---

## Setting Up SAML

This section discusses how to:

- Create the SAML administrator.
- Import digital certificates.
- Configure the SAML inbound setup
- Run the RedeployWSRP.cmd executable.

---

**Note.** You must perform all of the tasks in the order presented to correctly implement the use of the SAML token.

---

## Creating the SAML Administrator

The SAML administrator must have access to the SAML pages. You grant access to the SAML pages through the *PTPT1000* permission list.

To create the SAML administrator:

1. Access the User Profile page (PeopleTools, Security, User Profiles, User Profiles).
2. Add a new user or select an existing user who will be the SAML administrator.
3. Access the Roles page and insert a role that contains the *PTPT1000* permission list.
4. Save the user profile.

### See Also

*Enterprise PeopleTools 8.50 PeopleBook: Security Administration*, "Administering User Profiles," Creating a New User Profile

## Importing Digital Certificates

To implement SAML, you must import the digital certificate of the sender and store it in the keystore of participating PeopleSoft applications.

See *Enterprise PeopleTools 8.50 PeopleBook: Security Administration*, "Working with SSL and Digital Certificates," Configuring Digital Certificates.

## Configuring the SAML Inbound Setup

The SAML Inbound Setup page creates an InBound webservice in the producer site that maps the one PeopleSoft user ID to one SAML assertion subject and links the subject with the sender's digital certificate (public key). The SAML administrator sets up a webservice for each external user who accesses the PeopleSoft system who is using the SAML security option. This information should be configured by the SAML administrator—someone who understands the external requirements and how these requirements map to the component permissions necessary for the user to accomplish the business task.

Access the Security Assertion MarkUp Language [SAML] Inbound Setup page (PeopleTools, Security, SAML Administration Setup, SAML Inbound Setup).

### Security Assertion MarkUp Language [SAML] Inbound Setup.

*Certificate Alias	<input type="text" value="qe_local"/>	<input type="button" value="Find"/>	*Active Flag	<input type="text" value="A"/>	<input type="button" value="Find"/>
*Issuer	<input type="text" value=".PEOPLESOFT.COM"/>				
*SubjectName	<input type="text" value="QEDMO"/>				
*QualifierName	<input type="text" value=".PEOPLESOFT.COM"/>				
*Mapping Peoplesoft UserID	<input type="text" value="QESAML"/>				
Long Description	<div></div>				

Security Assertion MarkUp Language [SAML] Inbound Setup page

<b>Certificate Alias</b>	Enter the sender's public key, which you imported in the previous step (Importing Digital Certificates).
	<b>Note.</b> This key must be base 64 encoded.
<b>Issuer</b>	Enter the domain name of the issuing entity.
<b>SubjectName</b>	Enter a user ID or email address.

<b>QualifierName</b>	Enter the domain name of the issuing entity.
<b>Mapping PeopleSoft UserID</b>	Enter the UserID to map to the SubjectName. This field sets the PeopleSoft internal permissions for the external user and prevents cross-site vulnerability.
	<b>Note.</b> This userID does not have to be the user ID of the sender, but must be a valid PeopleSoft user in the PSOPRDEFN table..
	<b>Note.</b> This field is internal to the PeopleSoft application and is hidden from all consumer sites and third party systems.

## Running the RedeployWSRP.cmd Executable

To run the RedeployWSRP.cmd executable:

- Navigate to the producer web server folder, for example *PSHOME/Webserver/bin*.
- Double-click the file to launch the program.
- Select *Option 5: Redeploy WSRPBaseService with SAMLToken Security Option*.

## Configuring WS-Security For PeopleSoft as a WSRP Producer

The *...peoplesoft\pspc\WEB-INF\* directory on the producer web server contains a *server-config.wsdd* file, which contains the WSRPBaseService definition plus the following variations. Each one of the variations is associated with a security option:

<b>Security Option</b>	<b>Filename</b>
1. None	server-config.wsdd.none
2. UsernameToken in ClearText	server-config.wsdd.username-token
3. Authentication Token as Username Token with full security	server-config.wsdd.fullsecurity
4. Username Token with no password and digital signature	server-config.wsdd.untnp
5. SAML Token either with digital signature only or with digital signature and encryption.	server-config.wsdd.saml-token

PeopleSoft applications provide two options for modifying the security constraint on the WSRPBaseService. Which option you should use depends on whether your web server is currently running. If your web server is running, you can modify WSRPBaseService without rebooting the server.

**Note.** The *PIA\_HOME\websevr\%DOMAIN\_NAME%\* directory contains the redeployWSRP batch file.

### ***Modifying WSRPBaseService Without Rebooting***

To modify WSRPBaseService without having to reboot the web server:

1. Using the Command Prompt, change to the directory containing the redeployWSRP file:

```
cd %PIA_HOME%\webserv\%DOMAIN_NAME%\
```

2. Enter the command: redeployWS <option #>

For example:

```
redeployWSRP 2
```

This will redeploy WSRPBaseService using UsernameToken in clear text security option. The web service is undeployed and redeployed dynamically without having to reboot the web server.

### ***Modifying WSRPBaseService and Rebooting***

To modify WSRPBaseService if the web server is not up and running:

1. Using the Command Prompt, change to the ...\\peoplesoft\\pspc\\WEB-INF directory:

For example:

```
cd %PIA_HOME%\webserv\%DOMAIN_NAME%\applications\peoplesoft\pspc\WEB-INF\
```

2. Copy <desired server-config.wsdd.option> to replace server-config.wsdd file.
3. Reboot the WSRP Producer Portal web server to allow the newly installed server-config.wsdd file to deploy web services for the producer.

---

## **Defining Nodes With WS-Security**

This section discusses how to set up nodes for WS-Security.

### **Pages Used to Define Nodes With WS-Security**

<b><i>Page Name</i></b>	<b><i>Definition Name</i></b>	<b><i>Navigation</i></b>	<b><i>Usage</i></b>
Portal	IB_NODEURI	PeopleTools, Portal, Node Definitions, Portal	Enter required information for the WS-Security node.

Page Name	Definition Name	Navigation	Usage
WS Security	IB_NODESECURITY	Click the WS Security tab from the Portal page.	To enable secure access based on WS-Security for remote producers.  <b>Note.</b> Secure access is optional. You must be an administrator to access this page.

## Configuring WS-Security For WSRP Consumers

Access the WS Security page (PeopleTools, Portal, Node Definitions).

Node Name ASYNCR\_MDN

Authentication Token

\*Authentication Token Type None

☐ Encrypted

☐ Digitally Signed

WS Security page

Select Authentication Token Type options based on the options specified in the server-config.wsdd file.

Security Option	Authentication Token Options
None	Select <i>None</i> as the Authentication Token Type, and clear all check boxes.
Authentication Token as Username Token	Select <i>Username Token</i> as the Authentication Token Type, and clear all check boxes.
Authentication Token as Username Token with full security	Select <i>Username Token</i> as the Authentication Token Type, and select both the Encrypted and Digitally Signed check boxes.  <b>Note.</b> With this setting, the Username Token is encrypted and messages are digitally signed.

<b>Security Option</b>	<b>Authentication Token Options</b>
Authentication Token as Username Token with no password and digital signature	Select <i>Username Token, no password</i> as the Authentication Token Type, and select the Digitally Signed check box.
Authentication Token as SAML Token with digital signature	Select <i>SAML Token</i> as the Authentication Token Type . <b>Note.</b> By default, SAML Tokens are digitally signed.
Authentication Token as SAML Token with digital signature and encryption.	Select <i>SAML Token</i> and select the Encrypted check box. <b>Note.</b> With this setting, the SAML Token is encrypted and by default messages are digitally signed.

**See Also**

Chapter 17, "Using WSRP to Consume and Produce Remote Portlets," Importing Producer Information, page 520

Chapter 16, "Configuring WS-Security For WSRP Consumption and Production," Determining Security Requirements, page 497

Chapter 10, "Configuring the Portal Environment," Defining Portal Nodes, page 254



## Chapter 17

# Using WSRP to Consume and Produce Remote Portlets

This chapter discusses how to:

- Set up the local gateway URL. (Consumer)
- Use the WSRP Consumption Wizard. (Consumer)
- Display remote portlets on a homepage. (Consumer)
- Produce remote portlets. (Producer)

---

**Note.** Use the following link to the wsrp-primer for more information about WSRP.

---

### See Also

Chapter 6, "Administering Portal Homepages and Pagelets," page 153

<http://www.oasis-open.org/committees/download.php/10539/wsrp-primer-1.0.html>

---

## Setting Up the Local Gateway URL (Consumer)

This section discusses how to set up the local gateway URL.

The local gateway is used for administrative tasks , such as registering producers.

### Page Used to Set Up the Local Gateway URL

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Gateways	IB_GATEWAY	PeopleTools, Integration Broker, Configuration, Gateways Select <i>Local</i> Gateway.	Set up a web server to use for administration tasks, such as registering producers.

## Setting Up the Local Gateway URL

Access the Gateways page (PeopleTools, Integration Broker, Configuration, Gateways).

### Gateways

Gateway ID: LOCAL

☒ Local Gateway



☐ Load Balancer

URL:

[Ping Gateway](#)

[Gateway Setup Properties](#)

[Load Gateway Connectors](#)

Connectors				Customize   Find      			First	1-9 of 9	Last
	*Connector ID	Description	*Connector Class Name						
1	SMTPTARGET		SMTPTargetConnector	<a href="#">Properties</a>	+	-			
2	FILEOUTPUT		SimpleFileTargetConnector	<a href="#">Properties</a>	+	-			
3	PSFTTARGET		PeopleSoftTargetConnector	<a href="#">Properties</a>	+	-			
4	JMSTARGET		JMSTargetConnector	<a href="#">Properties</a>	+	-			
5	HTTPTARGET		HttpTargetConnector	<a href="#">Properties</a>	+	-			
6	GETMAILTARGET		GetMailTargetConnector	<a href="#">Properties</a>	+	-			
7	FTPTARGET		FTPTargetConnector	<a href="#">Properties</a>	+	-			
8	AS2TARGET		AS2TargetConnector	<a href="#">Properties</a>	+	-			
9	PSFT81TARGET		ApplicationMessagingTargetConnector	<a href="#">Properties</a>	+	-			

Gateways page

To set up a web server for WSRP administration tasks:

1. Change the URL to: *http://<web server name:port>/PSIGW/PeopleSoftListeningConnector*

---

**Note.** If you omit the port number, the system presumes a value of 9000.

---

2. Click Save.
3. Click Load Gateway Connectors.

The connectors will appear in the Connectors grid.

4. Click Save.

---

## Using the WSRP Consumption Wizard (Consumer)

This section provides an overview of the WSRP Consumption Wizard and discusses how to:

- Import producer information.
- Register producers.

- Create content references.


## Understanding the WSRP Consumption Wizard Data

The Consumption Wizard is a tool used to easily import producer information, register producers, and create content references (CRefs) for the producer offered portlets. Once a producer has been registered, the offered portlets become available for use by any of your portals. Although the automatically created CRefs are only available to users of the current portal, an administrator can log into a different portal and manually create CRefs for any of the producer offered portlets.

### See Also

[Chapter 5, "Administering Portals," Administering Content References, page 109](#)

## Common Elements Used in This Section

	<p>Click a number to go to the corresponding page within portal wizard.</p> <p>1: Import Producer Information.</p> <p>2: Registration.</p> <p>3: Producer Offered Portlets.</p> <p>The different background signifies:</p> <ul style="list-style-type: none"><li>• <i>Pale Blue</i>: The page has been updated.</li><li>• <i>Orange</i>: Current page.</li><li>• <i>Dark Blue</i>: The page has not been updated.</li></ul>
<b>Previous and Next</b>	Click to navigate page by page.
<b>Producer Title and Description</b>	<p>Enter meaningful titles and descriptions.</p> <hr/> <p><b>Note.</b> The producer title is also used as the default folder label for the content references created based on this producers offered portlets.</p> <hr/>
<b>Creation Date/Time and Last Update Date/Time</b>	Indicates the date and time that the remote portlet information was first entered in the PeopleSoft system and most recently updated.

## Pages Used to Navigate WSRP Consumption Wizard

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Define Producers	PT_WSRP_PRO_LIST	PeopleTools, Portal, WSRP Consumption	Add a new producer, edit an existing producers information, or delete a producer.
Import Producer Information	PT_WSRP_IMPORT	Click Edit or Add a Import Producer Information	Enter the producers WSDL URL to import required information.
Registration	PT_WSRP_REGISTER	Click Save on the Import Producer Information page.	Input producer registration information.
Producer Offered Portlets	PT_WSRP_PORT_LIST	Click Next on the Registration page.	View and edit portlet information. Create PeopleSoft portlets based on these portlets.
Producer Details	PT_WSRP_PRODETAIL	Click the Producer Details link on the Producer Offered Portlets page.	View the producers actual Service DescriptionResponse.
Portlet Content References	PT_WSRP_PORT_CREFS	Click the View button for the desired portlet on the Producer Offered Portlets page.	View the Content Reference of the newly created portlet.
Portlet Details	PT_WSRP_PORTDETAIL	Click the Details button for the desired portlet on the Producer Offered Portlets page.	View the XML that was returned from the producer for this portlet.

## Importing Producer Information

Access the Import Producer Information page (PeopleTools, Portal, WSRP Consumption. Click the Add a Producer or Edit buttons).

## WSRP Consumption

## Step 1 of 3



Next &gt;

## Import Producer Information

Enter new or modify existing producer information.

<b>Producer Name:</b>	XYZ_CORP
<b>Creation Date/Time:</b>	11/01/2004 11:45:39AM
<b>Last Update Date/Time:</b>	
<b>*Producer Title:</b>	<input type="text" value="XYZ Corp"/>
<b>Description:</b>	<input type="text" value="XYZ Corporation"/>
<b>Security Node:</b>	<input type="text"/>

**Web Service Endpoint URL**

<b>*URL:</b>	<input type="text" value="http://sunwsrp.dyndns.org/portal/wsrp/wsd/default"/>	<input type="button" value="Import"/>
<b>Status:</b>	Import Complete	

Import Producer Information page

- Producer Name** Enter a meaningful name for this producer.  
The value entered here is also used as part of the default folder name for content references created based on this producers offered portlets.
- Security Node (Optional)** Select the node definition that was created for WS-Security.  
See Chapter 16, "Configuring WS-Security For WSRP Consumption and Production," page 497.

**Web Service Endpoint URL**

- URL** Enter the web service WSDL URL for this producer.  

---

**Note.** The producer provides this information.

---
- Import** Click to import the required producer information.
- Status** If the WSDL URL is correct the status will automatically update from *No Data Imported* to *Import Complete*. If the URL is incorrect, or the producer is unavailable, you will receive a warning message with information on how to make corrections.

**Save**

Click to save the producer information and jump to the Registration page.

## Registering Producers

Access the Registration page (Click the Next button).

### WSRP Consumption

Step 2 of 3



&lt; Previous

Next &gt;

## Registration

Get the producer registration information.

**Producer Title:** XYZ Corp

**Description:** XYZ Corporation

**Registration Type:** In-Band

**Creation Date/Time** 11/01/2004 11:45:40AM

**Last Update Date/Time** 11/01/2004 11:47:37AM

#### ▼ Status

Registration is complete.

Registration is required.

**Handle:** 4WMUDGQAAAAANQASGFQAAAA

#### Consumer Properties

**Consumer Name:** http://c1jcolli031902.peoplesoft.com/psc/ps/3c6fd4e4-285f-11d9-9b1c-ab327c31d4f2**Consumer Agent:** PeopleSoft Portal V.8.46-110.2

#### ▼ Registration Properties

**\*Email address:** erica\_kane@peoplesoft.com**\*Company Name:** PeopleSoft, Inc.**\*Location:** Pleasanton, CA 94588**Save**[Return to Producer List](#)**Register****Deregister**

Registration page

**Registration Type**

Choose a registration type:

*In Band:* In this process, the consumer sends a request to register with the producer, along with any properties required by the producer.

*Out of Band:* In this process the producer and consumer specify manual processes such as email or phone conversations to establish registration.

*None:* Registration is not required. No further information is required. Continue to the next page.

**Status****Handle**

*Out of band:* Enter the handle value conveyed to you by the producer.

*In Band:* Read-only. The value will display automatically when you click Register.

**Validate**

Click to check the validity of the handle.

---

**Note.** This button appears only when the registration type is *Out of band*.

---

**Handle Status**

Indicates the validity of the handle.

---

**Note.** This field appears only when the registration type is *Out of band*.

---



---

**Note.** Messages regarding registration status are also displayed within this group box.

---

**Consumer Properties****Consumer Name and Consumer Agent**

Displays the consumer information, (URL and software version), which is required by and sent to the producer.

**Registration Properties**

This group box appears only when the registration type is *In-Band*.

---

**Note.** The fields contained in this group box may vary according to each producers requirements.

---

**Register**

Click to register the producer.

If registration is successful, the registration message in the Status group box is updated and the Deregister button appears.

---

**Important!** When you deregister a producer, the handle becomes invalid and all user portlet personalizations are removed. You will need to obtain a new handle before attempting a reregistration.

---

## Creating Content References

Access the Producer Offered Portlets page (Click the Next button)..

**WSRP Consumption**

**Step 3 of 3**

1 2 3

< Previous

### Producer Offered Portlets

View and edit portlet information for this producer.

**Producer Title:** XYZ Corp  
**Description:** XYZ Corporation  
**Creation Date/Time** 11/01/2004 11:45:40AM  
**Last Update Date/Time** 11/01/2004 11:47:37AM





Update Portlet List

[Producer Details](#)

Select All

Deselect All

Create Pagelets for All Selected Portlets

Producer Offered Portlets			Customize   Find   		First  1-13 of 13  Last	
Actions		Description				
Portlet Handle	Portlet Title	Selected	View	Details	Delete	
BookmarkPortlet	Sun: BookmarkPortlet	<input checked="" type="checkbox"/>	<a href="#">View</a>	<a href="#">Details</a>	<a href="#">Delete</a>	
GooglePortlet	Sun: Google Search	<input checked="" type="checkbox"/>	<a href="#">View</a>	<a href="#">Details</a>	<a href="#">Delete</a>	
HelloWorldPortlet	Sun: HelloWorldPortlet	<input checked="" type="checkbox"/>	<a href="#">View</a>	<a href="#">Details</a>	<a href="#">Delete</a>	
InputStreamPortlet	Sun: Test Input Stream	<input checked="" type="checkbox"/>	<a href="#">View</a>	<a href="#">Details</a>	<a href="#">Delete</a>	
JSPPortlet	Sun: ShowTime Portlet	<input checked="" type="checkbox"/>	<a href="#">View</a>	<a href="#">Details</a>	<a href="#">Delete</a>	
NamedDispatcherGetAttributePortlet	Sun: Test NamedDispatcher	<input checked="" type="checkbox"/>	<a href="#">View</a>	<a href="#">Details</a>	<a href="#">Delete</a>	
NavStatePortlet	Sun: Nav State	<input checked="" type="checkbox"/>	<a href="#">View</a>	<a href="#">Details</a>	<a href="#">Delete</a>	
NotepadPortlet	Sun: Notepad Portlet	<input checked="" type="checkbox"/>	<a href="#">View</a>	<a href="#">Details</a>	<a href="#">Delete</a>	
PrefPortlet	Sun: PrefPortlet	<input checked="" type="checkbox"/>	<a href="#">View</a>	<a href="#">Details</a>	<a href="#">Delete</a>	
RequestDispatcherGetAttributePortlet	Sun: Test RequestDispatcher	<input checked="" type="checkbox"/>	<a href="#">View</a>	<a href="#">Details</a>	<a href="#">Delete</a>	
RssPortlet	Sun: RSS	<input checked="" type="checkbox"/>	<a href="#">View</a>	<a href="#">Details</a>	<a href="#">Delete</a>	
TestPortlet	Sun: TestPortlet	<input checked="" type="checkbox"/>	<a href="#">View</a>	<a href="#">Details</a>	<a href="#">Delete</a>	
WeatherPortlet	Sun: WeatherPortlet	<input checked="" type="checkbox"/>	<a href="#">View</a>	<a href="#">Details</a>	<a href="#">Delete</a>	

Producer Offered Portlets page

<b>Update Portlet List</b>	Click to refresh the list of offered portlets.
	<b>Important!</b> When previously listed portlets are no longer offered by a producer, the Delete button associated with those portlets becomes active.
<b>Producer Details</b>	Click to go to the Producer Details page and view the producers actual ServiceDescriptionResponse.
<b>Create Pagelets for All Selected Portlets</b>	<p>Click to create content references for the selected portlets.</p> <p>These content references can be located in a folder created automatically using the values entered for this producer name and title for its name.</p> <p>This will make the portlets available when end users are personalizing their homepages.</p> <p><b>Note.</b> You can also create the content references manually in any portal using the Structure and Content page.</p> <p>See <a href="#">Chapter 5, "Administering Portals," Administering Content References, page 109.</a></p>

### ***Producer Offered Portlets***

<b>Selected</b>	<p>Select the portlets you want to consume, then click the Create Pagelets for All Selected Portlets button to create content references.</p> <p>You can also use the Select All and Deselect All buttons.</p>
<b>View</b>	<p>Click to go to the Portlet Content References page to view the newly created content reference in a new browser window. The page displays the portal where the new portlet resides, the name and label of the portlet, and a link to the content reference.</p> <p><b>Note.</b> This button is disabled until a content reference is created.</p>
<b>Details</b>	Click to go to the Portlet Details page to view the XML returned from the producer for this portlet.
<b>Delete</b>	<p>Click the Update Portlet List to refresh the list of portlets that are being offered by this producer. The Delete button will become active for portlets that are no longer being offered.</p> <p>Click the Delete button to remove the portlet and associated content reference.</p>

## **Displaying Remote Portlets on a Homepage (Consumer)**

This section discusses how to:

- Review content references.
- Display remote portlets on a homepage.

## Pages Used to Display Remote Portlets

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Content Ref Administration	PORTAL_CREF_ADM	<ul style="list-style-type: none"> <li>• PeopleTools, Portal, Structure and Content, Portal Objects, Pagelets  Click the desired folder name, then click Edit for the desired portlet.</li> <li>• PeopleTools, Portal, WSRP Consumption, Define Producers  Click Edit for the desired producer, then navigate to the Producer Offered Portlets page.  Click View on the Producer Offered Portlets page for the desired pagelet, then click the navigation link on the Portal Content References page.</li> </ul>	Review content reference data or manually create a content reference for an offered portlet.
Tab Content	PORTAL_TAB_CON_ADM	PeopleTools, Portal, Structure and Content, Portal Objects, Homepage, Tabs  Click the Edit button associated with the My Page content reference.  Click Tab Content.	Select the portlets that can be used for this homepage tab definition.
Personalize Content	PORTALHPCOMP	Click the Personalize Content link on the Homepage.	Choose the remote portlets that you want to appear on your homepage.

## Reviewing Content References

Access the Content Ref Administration page for a remote portlet (PeopleTools, Portal, Structure and Content, Portal Objects, Pagelets). Click the desired folder name, then click Edit for the desired portlet).

General

Security

[Root](#) > [Portal Objects](#) > [Pagelets](#) > [XYZ Corporation](#) >

## Content Ref Administration

**Name:** WSRP\_PORTLET\_5

**Author:** QEDMO

**\*Label:** My Calendar

**Parent Folder:** XYZ Corporation

[Copy object](#)

[Select New Parent Folder](#)

**Long Description:**  
(254 Characters)

**Product:**

**Sequence number:**

**Object Owner ID**

**Usage Type:** Pagelet

**Storage Type:** Remote by URL

**Creation Date:** 12/07/2004

[Add Content Reference](#)

### URL Information

**URL Type:** Remote Portlet

**\*Producer:** XYZ\_CORP

**\*Portlet:** PORTLET\_5

[Producer Details](#)

### Pagelet Attributes

**Default Column:** Column 1

**Help ID:**

☐ Hide minimize image

☐ Hide refresh image

#### Edit URL Information

**Node Name**

**URL Type:**

### Content Reference Attributes

**Name:**

**Label:**

☒ Translate

[Attribute Information](#)

**Attribute value:**

[Delete](#)

[Add](#)

### Content Ref Administration page

The field values that are automatically entered are based on the values entered for the producer via the portal wizard. These include:

- Name** Defaults to WSRP\_<portlet handle>.  
This field is read-only and cannot be changed.
- Label** Enter a meaningful label description for this portlet.  
Defaults to the remote portlets short title.

<b>Parent Folder</b>	Defaults to the producers title. Click the Select New Parent button to move the content reference to a different folder.
<b>Usage Type</b>	Defaults to <i>Pagelet</i> .
<b>Storage Type</b>	Defaults to <i>Remote by URL</i> . This field is read-only and cannot be changed.

### ***URL Information***

<b>URL Type</b>	Defaults to <i>Remote Portlet</i> .
<b>Producer</b>	Defaults to the producers name.
<b>Portlet</b>	Defaults to the portlets handle.
<b>Producer Details</b>	Click to go to the consumption wizards' Import Producer Information page for this producer.

---

**Note.** When you create content references manually for offered portlets update fields by using the same default values explained in this section.

---

See [Chapter 5, "Administering Portals," Defining Content References, page 112.](#)

## **Displaying Remote Portlets on a Homepage**

Access the Tab Content page (PeopleTools, Portal, Structure and Content, Portal Objects, Homepage, Tabs. Click the Edit button associated with the My Page content reference. Click the Tab Content tab).

General Security **Tab Content** Tab Layout

[Root](#) > [Portal Objects](#) > [Homepage](#) > [Tabs](#) >

## Tab Content

**Label:** My Page

- \* Select the pagelets that can be used for this homepage tab definition.
- \* Set the pagelet behavior with the drop down list next to the selected pagelet.
- \* Select the "Include all?" checkbox to display all pagelets from the portal registry for this category. This setting is used for the "Personalize Content" page.

The screenshot shows the 'Tab Content' configuration page for 'My Page'. It displays three categories of portlets:

- QE**: Includes a checked 'Include all?' checkbox and a list of 13 portlets (Display Cookies, QE Portal Links, QE Redirect Test, QE\_COMMAINCOOKIE, QE\_Counter, Signin, Test Global Variables, Who Am I, User-based Cache, Role-based Cache, Application-based Cache) each with an 'Optional' dropdown.
- XYZ Corp**: Highlighted with a red box. Includes a checked 'Include all?' checkbox and a list of 4 portlets (Sun: BookmarkPortlet, Sun: Google Search, Sun: Notepad Portlet, Sun: WeatherPortlet) each with an 'Optional' dropdown.
- News**: Includes an unchecked 'Include all?' checkbox and a list of 2 portlets (Excite Business News, Technology News) each with an 'Optional' dropdown.

### Tab Content page

---

**Note.** This procedure is performed by the administrator.

---

To make newly created remote portlets available to end users:

1. Access the Tab Content page.
2. Select the required portlets, or select the Include all? check box to display all portlets from the portal registry for remote content group box.
3. Click Save.

The selected portlets are now available for end users to display on their homepages.

To display the newly created remote portlets on a homepage:

1. On the homepage, click Personalize Content.
2. Choose the remote portlets that you want to appear on your homepage.
3. Click Personalize Layout to organize your portlets.
4. Click Save to save your selections and return to your homepage.

## Producing Remote Portlets (Producer)

This section discusses how to:

- Specify PeopleSoft content for WSRP production.
- Offer WSRP portlets.
- Work with WSRP content references.
- Set WSRP display mode.
- Map PeopleSoft styles to WSRP styles.
- Validate WSRP-compliance.
- Use PeopleSoft WSRP-produced content.

### Pages Used to Produce Remote Portlets

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Specify Publishing Options	PTPPB_WIZ_PUBOPT	PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard Click the Next button on the Specify Display Options page, or click the icon for step 6.	Specify the manner in which your portlet is published.
Producer Offered Portlets	PT_WSRP_PRO_PORT	PeopleTools, Portal, WSRP Production	Select the portlets you want to offer to third party consumers.
Producer Details	PT_WSRP_PRODUCEDTL	Click the Producer Details link on the Producer Offered Portlets page.	Displays the service description XML that is sent to the consumer for all portlets in the selected portal.
Portlet Details	PT_WSRP_PROPORTDTL	Click the Details button for the desired portlet on the Producer Offered Portlets page.	Displays the generated XML for this portlet that will be sent to a consumer.
Content Ref Administration	PORTAL_CREF_ADM	Click the Path link for the desired portlet on the Producer Offered Portlets page.	Review the content reference that was created automatically for this portlet.

<b>Page Name</b>	<b>Definition Name</b>	<b>Navigation</b>	<b>Usage</b>
Content Ref Security	PORTAL_CREF_SEC	Click the Security tab on the Content Ref Administration page.	Review the security information for this portlet.

## Specifying PeopleSoft Content for WSRP Production

This section discusses how to:

- Specify pagelets for WSRP production.
- Specify content references for WSRP production.

### **Specifying Pagelets for WSRP Production**

Access the Pagelet Wizard - Specify Publishing Options page.

Use Pagelet Wizard to create pagelets that you want to publish as WSRP portlets. On the Specify Publishing Options page, you have the option of selecting WSRP Portlet for Homepage pagelets and Template pagelets. This enables the pagelet's WSRP Producing option and causes the pagelet to appear on the Producer Offered Portlets page so that an administrator can expose it for remote, WSRP portal consumption. Also consider the Enable Caching option on the Specify Publishing Options page. If you select to enable caching, also choose the Caching Type and the Minutes to Cache.

---

**Note.** When you create a pagelet using the Pagelet Wizard and select the WSRP Portal option, the system automatically enables the pagelet for consumption by Oracle portals using the JPDK specification.

---

### **Specifying Components for WSRP Production**

To enable a component for WSRP production, select the WSRP Compliant option on the Internet tab of the Component Properties dialog box.

See *Enterprise PeopleTools 8.50 PeopleBook: PeopleSoft Application Designer Developer's Guide*, "Creating Component Definitions," Setting Internet Properties.

### **Specifying Content References for WSRP Production**

Access the Content Ref Administration page.

While you can also specify pagelets for WSRP production using the Pagelet Wizard, you specify PeopleSoft components and iScripts intended for WSRP production using content reference options.

---

**Note.** Content references for pagelets (portlets) also control whether the pagelet can be WSRP produced. For example, selecting the WSRP Producing option for a pagelet's content reference is equivalent to selecting the WSRP Portlet option on the Specify Publishing Options page in the Pagelet Wizard. This applies also to component-based and iScript-based pagelets

---

Access the Content Ref Administration page (PeopleTools, Portal, Structure and Content.).

General

Security

Root >

## Content Ref Administration

\*Name:

\*Label:

Long Description:

(254 Characters)

Product:

Sequence number:

Object Owner ID

Usage Type:

Target

Storage Type:

Remote by URL

Template Name:

Author:

QEDMO

Parent Folder:

Root

\*Valid from date:

12/28/2005

Valid to date:

Creation Date:

12/28/2005

☐ WSRP Producible

☐ No Template

Content Ref Administration page – WSRP Producible option

The WSRP Producible option only appears for content references with the Usage Type set to *Target* or *Pagelet*, with the URL Type set to one of these values:

- *PeopleSoft Component.*
- *PeopleSoft Generic URL.*
- *PeopleSoft Script.*
- *PeopleSoft Worklist URL.*

The WSRP Producible check box appears just to the right of the Storage Type drop-down list box, and, by default, it is not selected. Selecting this option causes the current content reference to appear on the Producer Offered Portlets page so that an administrator can choose to expose it for remote, WSRP portals. In turn, deselecting the option causes the current content reference to be removed from the list on the Producer Offered Portals page.

Because WSRP producible content must originate from the local node, if a content reference is modified to reference a remote node, the WSRP Producible check box will not appear, and the content reference will not be able to be exposed to remote, WSRP portlets.

WSRP producible content references possess a set of optional, WSRP-specific attributes, differentiating them from other content references. The following table contains descriptions of the WSRP content reference attributes:

Attribute	Attribute Label	Description	Values
WSRP_TITLE	WSRPTitle	Allows the exposed WSRP Title to be exposed to remote WSRP portals.	Entered text.
WSRP_DESCR	WSRPDescription	Allows the exposed WSRP Description to be different than the default content reference description.	Entered text.

<b>Attribute</b>	<b>Attribute Label</b>	<b>Description</b>	<b>Values</b>
WSRP_WINDOWSTATE	WSRPWindowState	Allows the ability to override the default window state behavior for portlets or target pages. The default window state without the content reference attribute override for portlets is <i>normal</i> . The default window state without the content reference attribute override for target content is <i>maximize</i> .	normal maximize

Content reference links created from content references displaying the WSRP Producible check box also display the WSRP Producible check box. By default, the option is not selected when you create a new content reference link, even if the WSRP Producible check box is selected for the associated content reference. As with the content reference, if you select the WSRP Producible option on the content reference link, it will appear on the Producer Offered Portlets page.

When working with WSRP producible content references, also consider the following:

- Portal registry folders do not display the WSRP Producible check box.
- Context Manager template pagelets, considered by the system to be a type of portlet, are WSRP producible. If keys or parameters expected by a supporting transaction are unavailable, the content will appear the same way as keys being unavailable in the PeopleSoft portal.
- If a WSRP producible content reference is moved to a different location in the portal registry, it still retains its ability to be WSRP produced.
- The Test Content Reference link only tests how the content reference will appear in the PeopleSoft portal, not WSRP.
- If you have a content reference that should only be available for WSRP and not through the PeopleSoft portal, hide the content reference from the PeopleSoft navigation system.

## Offering WSRP Portlets

Access the Producer Offered Portlets page (PeopleTools, Portal, WSRP Production).

The Producer Offered Portlets page displays PeopleSoft provider content that has been selected for WSRP production. That is, any pagelet appearing on this page has the WSRP Portlet option selected in the Pagelet Wizard, and any content reference that appears on this page has the WSRP Producible option selected on the Content Ref Administration page.

## WSRP Production

### Producer Offered Portlets

Search Criteria

\*Portal Name:

EMPLOYEE

Search

Portlet Title:

begins with

▼ Web Service Endpoint URL

URL:

http://ple-jcollins00.corp.peoplesoft.com/pspc/pswsdl/ps\_1/EMPLOYEE


▼ Oracle Portal Web Provider URL

URL:

http://ple-jcollins00.corp.peoplesoft.com/pspc/providers/psprovider/ps\_1/EMPLOYEE

Select All

Deselect All

Portlets					
Customize   Find   					
Portal Name	Title	Description	Selected	Details	Path
EMPLOYEE	Demonstration	Demonstration WSRP portlet.	<input checked="" type="checkbox"/>	<a href="#">Details</a>	<a href="#">Root&gt;Portal Objects&gt;Pagelets&gt;QE&gt;</a>

Save

[Producer Details](#)

Producer Offered Portlets page

<b>Portal Name</b>	Choose the portal where the content you want to offer is located, and click Search.
<hr/>	
	<b>Note.</b> To display portlets within all portals, select <i>All Portals</i> , and leave the Portlet Title value empty.
<hr/>	
<b>Portlet Title</b>	Enter the portlet title or a portion of it for increased search options. Use the "%" character for wild card values.
<hr/>	
<b>Web Service Endpoint URL</b>	Displays the URL that the WSRP consumer portal requires to import the offered content.
<hr/>	
<b>Oracle Portal Web Provider URL</b>	Displays the URL that the JPDK-compliant Oracle Portal requires to import the offered content.
<hr/>	
	<b>Note.</b> This value applies only for Pagelet Wizard pagelets.
<hr/>	

## Portlets

This section of the page is named "Portlets," despite the content being exposed could be components, pages, or iScripts. Regardless, the content is exposed and discovered by WSRP consuming portals as "portlets."

**Selected** Select the portlets you want to offer to consumers. Click the Select All button to offer all of the portlets listed. Click the Deselect button to remove all of the selections. If a content item in the list is not selected, it cannot be exposed to consumer portals.

---

**Note.** By default, each new item appearing in the portlets list will have the Selected check box selected.

---

**Details** Click to go to the Portlet Details page to view this portlet's generated XML that will be sent to a consumer.

**Path** Click to go to the Content Ref Administration page to view the CREF information for this portlet.

**Producer Details** Click this link to go to the Producer Details page to view the service description XML that is sent to the consumer for all portlets in this portal.

---

**Note.** If a WSRP producible content reference is deleted from the portal registry, the system removes it from the Producer Offered Portlets page.

---

## Working With WSRP Content References

Use the Content Ref Administration page to view WSRP Content reference information.

---

**Important!** Use the Content Ref Administration page only to select a new parent folder or update the security for the pagelet.

---

## Setting WSRP Display Mode

You can set the WSRP display mode using the system-wide, WSRP Display Mode option on the PeopleTools Options page (PeopleTools, Utilities, Administration, PeopleTools Options). This option enables you to control how PeopleSoft content is rendered when user interaction is initiated from within a WSRP consuming portal. This may override the WSRP Compliant setting within the component. You can select these settings:

- *Display as Portlet.* Renders content accessed from a link in a portlet inline, as a portlet, provided the component has the WSRP Compliant property enabled. As with PeopleSoft applications and new windows, opening a new window, versus replacing the window, is not possible due to session and state management issues.
- *Replace Existing Window.* All content navigated to from a portlet will render in the same browser window but take the user out of the consuming portal and into the PeopleSoft portal to render the content.

If a pagelet is WSRP-produced, and the end user clicks a link on the pagelet to take the user to a different page from PeopleSoft, the following describes the logic flow that determines whether:

- The link is displayed inline in the portlet window of the WSRP consuming portal.
  - The end user is taken out of the WSRP-consuming portal and into the PeopleSoft portal to render the PeopleSoft content.
1. Determine WSRP display mode.
    - If set to *Replace Existing Window*, display the PeopleSoft content in the PeopleSoft portal within the same window.
    - If set to *Display as Portlet* determine if content is WSRP compliant.
  2. Determine WSRP compliance status.
    - If WSRP compliance is not selected, display PeopleSoft content in the PeopleSoft portal within the same window.
    - If WSRP compliance is selected, display the PeopleSoft content inline as a portlet.

### See Also

*Enterprise PeopleTools 8.50 PeopleBook: System and Server Administration*, "Using PeopleTools Utilities," PeopleTools Options

## Mapping PeopleSoft Styles to WSRP Styles

For WSRP content, PeopleSoft style classes are mapped to corresponding WSRP styles. PeopleSoft provides PeopleSoft WSRP produced markup references only WSRP style names, not PeopleSoft styles. If a particular PeopleSoft style is not mapped to a corresponding WSRP style, the style can not be generated in the WSRP markup. The look and feel of the style in the WSRP consuming portal is determined by the WSRP CSS style sheets on the consuming portal.

PeopleSoft-to-WSRP styles exist adhere to WSRP specification 1.0(10.6). To modify the WSRP mapping, use the WSRP Style drop-down list box on the Miscellaneous tab for a style definition in Application Designer.

---

**Note.** Compared to PeopleSoft style options, WSRP style options are limited.

---

## Validating WSRP-Compliance

To ensure that PeopleSoft pages, iScripts, and portlets comply with WSRP standards, use the WSRP validation options in PeopleSoft Application Designer. You can validate a single project or an entire database. Results of the validation appear in the Validate output tab, as with any other validate operation. The first line of the WSRP validation results is "Begin Validate WSRP Compliance," with the last line being "End Validate WSRP Compliance."

The system validates WSRP compliance against the following definitions:

- HTML areas within pages.
- HTML objects.
- JavaScript objects.
- Templates.
- Application classes.
- PeopleCode.
- PeopleCode iScripts.

The validation generates errors after identifying the existence of any of the following keywords. The use of these keywords is not recommended:

- href="http://
- frameset
- src="http://
- .location
- action="http://
- top.
- parent.
- parent.frames
- window.
- IFRAME
- ILAYER

### ***Performing WSRP Validation For a Project***

To perform WSRP validation for a project:

1. In Application Designer, open the project containing the WSRP definitions.
2. Select Tools, Options and click the Validate tab.
3. On the Validate tab, select the Validate WSRP compliance check box, and click OK.

---

**Note.** By default, this option is not selected.

---

4. Select Tools, Validate project.

### ***Performing WSRP Validation For a Database***

To perform WSRP validation for a database:

1. Launch Application Designer, and signon to the appropriate database.
2. Select Tools, Validate Database for WSRP.

## Using PeopleSoft WSRP-Produced Content

This section discusses the differences between the behavior of PeopleSoft content within the PeopleSoft portal and a consuming WSRP portal:

- The main PeopleSoft menu portlet does not provide navigation of WSRP consumed content. The PeopleSoft WSRP content needs to be discovered through the consumer portlet's navigation.
- Hot keys (such as, CTRL + J for the System Information page) are available as long as they do not conflict with any hot keys defined on the consuming system.
- Pagebar functionality is not included in PeopleSoft WSRP produced content.

For example, the New Window hyperlink or the Multi-Language drop-down list will not display (so, multi-language functionality is *not* supported for WSRP content). The Copy URL functionality is available on the System Information page only.

- Personalizations, such as "Tab-Over," made through the My Personalizations page apply to WSRP content.
- Only PeopleSoft portlets and iScripts support Edit capability, not components.

---

**Note.** PeopleSoft portlets including an Edit URL do not support multiple instances of the of the portlet in the consuming portal.

---

## Chapter 18

# Working with JSR 168 Portlets

This chapter provides overviews of JSR 168 and PeopleSoft compliance with JSR 168, and discusses how to:

- Configure the portlet container.
- Create portlet applications for Oracle application server deployment.
- Describe the portlet entity registry.

---

## Understanding JSR 168

JSR 168 is a standard for portals running in a Java 2 Platform, Enterprise Edition (J2EE) architecture. It specifies content definitions much like pagelets, called portlets, that present portal content defined according to the JSR 168 standard. Portlets are Java-based web components that you can register in the PeopleSoft portal just as you register pagelets. Both HTML templates and frame templates support portlet content.

Portlets generate fragments of markup (such as HTML, Extensible HyperText Markup Language, and Wireless Markup Language). The PeopleSoft portlet engine manages and invokes portlets by using Java programs. This combines markup fragments that are generated by different portlets into a portal page.

You develop your own Java-based JSR 168 compliant portlets, and then register them by using the PeopleSoft portal registration components.

You expose your locally installed portlets to remote WSRP consumers by creating entries in the portal entity registry.

See [Chapter 18, "Working with JSR 168 Portlets," Describing the Portlet Entity Registry, page 541.](#)

---

## Understanding PeopleSoft Compliance with JSR 168

This section discusses:

- Java Portlet API
- Window states.
- Content caching.
- Portlet modes.

## Java Portlet API

Portlets obtain their context via the Portlet API- functions like access to user profile information for the current user, participation in the portal window and action event model, access to web client information, sharing of information with other portlets and a standard way of storing and retrieving per-user/per-instance portlet data persistently. PeopleTools portlet container supports the Java Portlet API 1.0. The Java Portlet API provides a URL-rewriting mechanism for creating links to trigger actions within a portlet without requiring knowledge on how URLs are structured in the particular web application. Portlets are grouped in a Portal Application by bundling them in a single web application archive file (WAR) with a Portlet deployment descriptor file. In addition, the API provides a mean for sharing data among portlets of the same Portal Application.

## Window States

The PeopleSoft JSR 168 engine accepts all of the required as well as any custom window states. However, although we allow any window state, PeopleSoft does not react to them. Portlets always appear to have a *Normal* amount of screen real-estate.

## Content Caching

In the current release, the PeopleSoft JSR 168 engine doesn't support the caching of portlet content based on caching elements that are embedded in the portlet; any caching tags in the portlet are ignored. However, the engine does support caching by using the `PSCACHECONTROL` content reference attribute.

See [Chapter 9, "Using Portal Caching Features," Implementing Pagelet Caching, page 202](#).

## Portlet Modes

The PeopleSoft JSR 168 engine supports all of the required portlet modes, which are:

- VIEW
- EDIT
- HELP

---

## Configuring the Portlet Container

If the `logClientIPOnMalformedRequest` initialization parameter is set to *true*, it will log the address of clients making failed attempts to access the Java portlets (due to malformed URLs).

If the `AccessJavaPortletsFromPeopleSoftPortalOnly` initialization parameter is set to *true*, only clients who have their `REMOTE_USER` setting will be allowed access. By default, all accesses from the users homepage supply the `REMOTE_USER` setting.

---

**Note.** You can secure all of the Java portlets on your system using J2EE security and role based permissions within your portlet logic.

---

The parameter can be found in the PortletContainerServlet definition in the web.xml of the /pspc application

The following is the relevant section from the /pspc/WEB-INF/web.xml within the domain:

```
<servlet>
  <servlet-name>PortletContainerServlet</servlet-name>
  <display-name>PeopleSoft PortletContainer Servlet</display-name>
  <servlet-class>com.peoplesoft.pt.portlet.container.ContainerServlet</servlet-
class>
  <!-- <load-on-startup>0</load-on-startup> -->
  <init-param>
    <param-name>logClientIPOnMalformedRequest</param-name>
    <param-value>true</param-value>
  </init-param>
  <init-param>
    <param-name>AccessJavaPortletsFromPeopleSoftPortalOnly</param-name>
    <param-value>false</param-value>
  </init-param>
</servlet>
```

---

## Describing the Portlet Entity Registry

The portlet entity registry is the source for both the PeopleSoft portlet content references and the portlets produced through WSRP.

## Choosing Portlets to Produce For WSRP

You can choose which portlets to produce for WSRP by including them as <portlet> elements in portletentityregistry.xml.

To produce portlets for WSRP:

1. Edit portletcontainer/WEB-INF/data/portletentityregistry.xml to add an <application> element for your new portlet application.

The following is the portletentityregistry.xml file as delivered, showing the production of four portlets for WSRP consumption, WSRPTestPortlet of the wsrpctest portlet application, TestPortlet1 and TestPortlet2 of the testsuite portlet application, and the HelloPortlet of the helloportletapp portlet application.

```
<?xml version="1.0" encoding="UTF-8"?>
<portlet-entity-registry>
  <application id="">
    <definition-id>helloportletapp</definition-id>
    <portlet id="">
      <definition-id>helloportletapp.HelloPortlet</definition-id>
    </portlet>
  </application>
  <application id="">
    <definition-id>wsrpctest</definition-id>
    <portlet id="">
      <definition-id>wsrpctest.WSRPTestPortlet</definition-id>
    </portlet>
    <portlet id="">
      <definition-id>wsrpctest.WSRPTestPortlet</definition-id>
    </portlet>
  </application>
  <application id="">
    <definition-id>testsuite</definition-id>
    <portlet id="">
      <definition-id>testsuite.TestPortlet2</definition-id>
    </portlet>
    <portlet id="">
      <definition-id>testsuite.TestPortlet1</definition-id>
    </portlet>
  </application>
</portlet-entity-registry>
```

The portletentityregistry.xml requires the following:

- Your portlet application must have a new <application> element in which the <definition-id> is the name of your portlet application directory and the id is unique within the file.
- Any portlets you wish to produce from your portlet application must have their own <portlet> elements within your new <application> element. Use the delivered portletentityregistry.xml as a guide.

---

**Note.** Each portlet must have a unique id within the <application>. For example, refer to the same definition multiple times —each having different preferences and its own unique entity ID.

---

- The <definition-id> of each portlet is a period-separated fully qualified name for the portlet of the form: "<portletApplicationName>.<portletName>" as shown. The portletApplicationName portion must match the <definition-id> of the <application>. The portletName portion must match the <portlet-name> element for the <portlet> definition in your portlet.xml file.
2. Restart your web server. Your Java portlets are now produced via WSRP technology. All WSRP Consumers you may choose to use will need to retrieve the new service description, which will include the new Java portlets.

## Appendix A

# Consuming PeopleSoft Components as a Web Service in OBIEE

PeopleSoft applications have an internal API that generates a PeopleSoft Component URL for any transaction page. This API is exposed as a standard web service, `PeopleSoftComponents`, which, when consumed, enables you to load PeopleSoft metadata into the OBIEE system. With the metadata loaded, you can configure action links from OBIEE Answers or Dashboards to a PeopleSoft transaction page without hard coding the PeopleSoft URL.

In addition to generating the PeopleSoft transaction page URL, the web service can also expose the key structure information and append to the URL a query string that uniquely identifies a transaction row within a PeopleSoft table. When you embed the URL as an action link, you enable the OBIEE application user to navigate to a PeopleSoft transaction page which shows a specific row of data.

---

## Understanding the OBIEE and the PeopleSoftComponents Web Service Interaction

When you create a PeopleSoft Component Action Type, the following background processing, which is transparent to the end-user, occurs between the OBIEE and PeopleSoft systems. The system invokes the OBIEE plug-in and calls the following service operations to load the component metadata.

There are two options:

- The system invokes the following service operations in sequence:
  1. `getComponentList()`
  2. `getComponentBaseUrl()`
  3. `getComponentKeys()`
- The system invokes the `getComponentMetadata()` service operation, which is a bulk-loading of the metadata

This table describes the `PeopleSoftComponents` Service Operations:

<i>Service Operation</i>	<i>Definition</i>	<i>Input Parameter</i>
<code>getComponentList()</code>	Returns the list of all the component names in the source PeopleSoft system.	None

<b>Service Operation</b>	<b>Definition</b>	<b>Input Parameter</b>
<code>getComponentKeys(string ComponentName)</code>	Returns all the search keys and the alternate search keys for the given component	Component name as string
<code>getComponentBaseURL(string ComponentName)</code>	Returns the PeopleSoft component URL for the given component name available within the PeopleSoft source system.	Component name as string
<code>getComponentMetadata()</code>	Returns PeopleSoft component URLs of all the components available within the PeopleSoft source system.  This service operation is used for bulk load of metadata that returns all components, their URLs, the component search keys and the alternate search keys.	None

The result set is stored in local OBIEE metadata tables. The PeopleSoft component metadata is displayed in a hierarchical tree structure which is available as Target URLs for a application when embedding the Action Links to various report columns.

---

**Note.** Any PeopleSoft component can be registered under various portal names. For any given component, there could be multiple associated URLs. In the OBIEE Report Designer, you must select the appropriate component URL using the tree browser.

---

## Setting Up OBIEE to Consume PeopleSoft Components as a Web Service

To set up OBIEE to consume PeopleSoft components as a web service, an OBIEE administrator must perform the following tasks as a one-time activity:

1. Consume the PeopleSoftComponents web service.

An administrator with the appropriate permissions uses the OBIEE EM Integration Console to define URLs for web services registries (UDDI/WSIL) and consume the WSDLs. Multiple PeopleSoft application installations, such as CRM, HCM, SCM, EPM, and so on can co-exist and share a single OBIEE system.

2. Load the PeopleSoft component metadata into the OBIEE system.

Select Load PeopleSoft Meta Data from the menu.

---

**Note.** These activities need be performed only once prior to developing OBIEE reports and again after a PeopleTools upgrade to reload the PeopleTools metadata.

---

## Appendix B

# Language Support For Consuming and Producing Remote Portlets

The ServiceDescription provides translations for the following strings: title, shortTitle, description, displayName, and keywords. The main language value is placed inside <offeredPortlets> tag, other language strings are inside <resources> tags at the end of xml file.

PeopleSoft stores the mapping of 3-character PeopleSoft language codes such as *ENG*, *FRA*, *CFR*, and *DUT*, as well as ISO locales such as *en*, *en-US*, *fr*, *fr-ca*, and *nl*.

---

## Language Support For Consuming Remote Portlets

When we consume remote portlets we request service description XML from the remote producer. We store the definition of the remote producer in the PeopleSoft database. We then parse service description XML and store the definitions of the remote portlets. If there is multilanguage data in the service description, we map ISO locales to PeopleSoft language codes and store the values in our database along with the correct PeopleSoft language codes.

---

**Note.** We are interested only in such locales that correspond to installed PeopleSoft language codes. All other locales are ignored.

---

When parsing the remote ServiceDescription, we look for matches between ISO locales and installed PeopleSoft language codes. We accept the string values when matches are found. Here are the rules:

- Accept if an exact match is found. For example, if *fr-ca* (CFR) is installed and *fr-ca* is provided in ServiceDescription.
- If the child is installed and they provide only the parent, we accept parent for child. For example, if *fr* (FRA) and *fr-ca* (CFR) are installed and only *fr* is provided, it is matched to both *fr* and *fr-ca*.
- If the parent locale is installed, but only children are provided, then accept the child for the parent. For example, if *fr* (FRA) is installed and *fr-ca* (CFR) is provided, then accept it for FRA.

---

## Language Support Rules For Producing Remote Portlets

When we produce remote portlets, we look at the definitions of the appropriate content references and generate the ServiceDescription. If there are any multilanguage strings (like cref label, description and keywords), we may have to include them into ServiceDescription. The remote producer sends us the list of "desired locales" he is interested in and in response we determine whether these desired locales have matching installed PeopleSoft language codes.

When generating the ServiceDescription, PeopleSoft will try to find matches for the desired locales to PeopleSoft installed languages using the following rules:

- If the locale is installed, return it.
- If only the parent is installed, return the parent. For example, if the desired locale is *fr-ca* and only *fr* is found, return *fr*.
- If no match is found, return *en* (English).

## Appendix C

# Understanding Changes in Portal Configuration Settings

In the current release, PeopleSoft portal configuration has moved from a file-based environment to a PeopleSoft Pure Internet Architecture interface. This appendix provides a complete listing that maps PeopleTools 8.43 and earlier portal settings to the current PeopleTools web profile settings. Within each of several broad functional categories, the release 8.43 properties are listed alphabetically, alongside the corresponding web profile fields that replace them.

The portal configuration files still exist in the current PeopleTools release, but they now contain only a small number of properties, and they're not meant to be edited. Don't modify any of the following files unless directed to do so by a PeopleSoft representative:

- Configuration.properties
- Pstools.properties
- Browserprops.xml
- Cookierules.xml

### See Also

Chapter 10, "Configuring the Portal Environment," Configuring Web Profiles, page 216

---

## General Properties

You can now set these properties from the configuration.properties file by using the Web Profile Configuration - General page:

<b>Release 8.43 Configuration Property</b>	<b>Corresponding Web Profile General Field</b>
AuthTokenDomain	Authentication Domain
compressCacheFiles	Compress Response References
compressResponse	Compress Responses
compressMimeTypes	Compress MIME Types (compress Multipurpose Internet Mail Extensions types)

<b>Release 8.43 Configuration Property</b>	<b>Corresponding Web Profile General Field</b>
CompressReportOutput	Compress Report Output field in the Reports group box.
CompressReportOutputNetscape	Compress Report Output field in the Reports group box.
enableNewWindow	Enable New Window
enableProcessingWait	Enable Processing Message
enableReportRepository	Enable Report Repository field in the Reports group box.
helpUrl	Help URL
physicalpath	Non-standard Base Path
ReportRepositoryPath	Report Repository Path field in the Reports group box.
saveConfirmDisplayTime	Save Confirmation Display Time
singleThreadNS	Single Thread Netscape
threadDelay	Single Thread Delay

**See Also**

[Chapter 10, "Configuring the Portal Environment," Configuring General Portal Properties, page 220](#)

---

## Security Properties

You can now set these properties from the configuration.properties file by using the Web Profile Configuration - Security page:

<b>Release 8.43 Configuration Property</b>	<b>Corresponding Web Profile Security Fields</b>
byPassSignOn	Allow Public Access field in the Public Users group box.
defaultPWD	Password field in the Public Users group box.
defaultUSERID	User ID field in the Public Users group box.
defaultXMLLinkUSERID	User ID field in the XML Link group box.
defaultXMLLinkPWD	Password field in the XML Link group box.
portalUseHttpForSameServer	PIA use HTTP Same Server (PeopleSoft Pure Internet Architecture use HTTP same server)

<b>Release 8.43 Configuration Property</b>	<b>Corresponding Web Profile Security Fields</b>
sessionTimeout	Authenticated Users - Inactivity Logout field in the Authenticated Users group box.
SSLRequired	Secured Access Only field in the SSL group box.
timeoutWarningScript	Timeout Warning Script and Override fields in the Authenticated Users group box.
userIDCookieAge	Days to Autofill User ID
UseSecureCookieWithSSL	Secure Cookie with SSL field in the SSL group box.
warningTimeout	Inactivity Warning field in the Authenticated Users group box.
XMLLinkUseHttpForSameServer	XML Link Use HTTP Same Server field in the XML Link group box.

You can now set these properties from the pstools.properties file by using the Web Profile Configuration - Security page:

<b>Release 8.43 Pstools Property</b>	<b>Corresponding Web Profile Security Field</b>
tuxedo_network_disconnect_timeout	Disconnect Timeout field in the Web Server Jolt Settings group box.
tuxedo_receive_timeout	Receive Timeout field in the Web Server Jolt Settings group box.
tuxedo_send_timeout	Send Timeout field in the Web Server Jolt Settings group box.

### See Also

[Chapter 10, "Configuring the Portal Environment," Configuring Portal Security, page 225](#)

---

## Virtual Addressing Properties

You can now set these properties from the configuration.properties file by using the Web Profile Configuration - Virtual Addressing page:

<b>Release 8.43 Configuration Property</b>	<b>Corresponding Web Profile Virtual Addressing Fields</b>
defaultPort	Port field in the Default Addressing group box.
defaultScheme	Protocol field in the Default Addressing group box.

<b>Release 8.43 Configuration Property</b>	<b>Corresponding Web Profile Virtual Addressing Fields</b>
pswebservername	Name field in the Default Addressing group box.
relativeURL	Generate Relative URLs
RPS	Protocol, Host, HTTP, and HTTPS fields in the Reverse Proxy Server List group box.

**See Also**

[Chapter 10, "Configuring the Portal Environment," Configuring Virtual Addressing, page 232](#)

---

## Cookie Rules

You can now set these elements and attributes from the cookierules.xml file by using the Web Profile Configuration - Cookie Rules page:

<b>Release 8.43 Cookierules.xml Element or Attribute</b>	<b>Corresponding Web Profile Cookie Rules Field</b>
block domain	Cookies Not Passed to Server
cookie name	Cookie Pattern
delete_on_logout	Delete Cookie on Logout
forward domain	Cookies Passed to Server
proxied	Proxied
secure	Secure

**See Also**

[Chapter 10, "Configuring the Portal Environment," Configuring Cookie Rules, page 234](#)

---

## Caching Settings

You can now set these properties from the configuration.properties file by using the Web Profile Configuration - Caching page:

<b>Release 8.43 Configuration Property</b>	<b>Corresponding Web Profile Caching Field</b>
CachePurgeAllHitCount	Cache Purge All Hit Count field in the On the Web Server group box.
CacheTargetContent	Cache Target Content field in the On the Web Server group box.
chartdirphys	Chart Directory field in the On the Web Server - Directories group box.
chartdirweb	Chart Web Directory field in the On the Web Server - Directories group box.
cssdirphys	CSS Directory (cascading style sheet directory) field in the On the Web Server - Directories group box.
cssdirweb	CSS Web Directory (cascading style sheet web director) field in the On the Web Server - Directories group box.
enableBrowserCache	Cache Generated HTML field in the On the Browser group box.
enableNoVersion	Copy Image/CSS (No Versioning) (copy image/cascading style sheet (no versioning) field in the On the Web Server - Directories group box.
imagedirphys	Image Directory field in the On the Web Server - Directories group box.
imagedirweb	Image Web Directory field in the On the Web Server - Directories group box.
jsdirphys	JavaScript Directory field in the On the Web Server - Directories group box.
jsdirweb	JavaScript Web Directory field in the On the Web Server - Directories group box.
maxSavedState	Number of States Supported field in the On the Browser group box.
PortalCacheHomepageOnBrowser	Cache Homepage field in the On the Browser group box.
PortalCacheObjects	Cache Portal Objects field in the On the Web Server group box.
PortalCacheStaleInterval	Cache Stale Interval field in the On the Web Server group box.
PortalHomepageStaleInterval	Homepage Stale Interval field in the On the Browser group box.
portalUseCachedProxiedJS	Cache Proxied JavaScripts field in the On the Web Server group box.

You can now set these elements and attributes from the browserprops.xml file by using the Web Profile Configuration - Caching page:

<b>Release 8.43 Browserprops.xml Element or Attribute</b>	<b>Corresponding Web Profile Caching Field</b>
property name="CacheHomePage"	Cache Home Page field in the Browsers group box.
useragent id	User Agent ID field in the Browsers group box.

**See Also**

Chapter 10, "Configuring the Portal Environment," Configuring Caching, page 235

---

## Trace and Debug Properties

You can now set these properties from the configuration.properties file by using the Web Profile Configuration - Debugging page:

<b>Release 8.43 Configuration Property</b>	<b>Corresponding Web Profile Debugging Field</b>
connectionInformation	Show Connection Information
debug_inlinejavascript	Show JavaScript Inline HTML
debug_inlinestylesheet	Show StyleSheet Inline HTML
debug_overlap	Show Overlapping Fields
debug_savefile	Create File from PeopleSoft Pure Internet Architecture HTML Page
debug_showlayout	Show Layout
enableDebugDumpFile	Write Dump File
enableTrace	Show Trace Link at Signon
testing	Generate HTML for Testing

**See Also**

Chapter 10, "Configuring the Portal Environment," Configuring Trace and Debug Options, page 240

## Look and Feel Settings

You can now set these properties from the configuration.properties file by using the Web Profile Configuration - Look and Feel page:

<b>8.43 Configuration Property</b>	<b>Corresponding Web Profile Look and Feel Fields</b>
authtokenenabled_page	Auth Token Enable Page field in the Other Pages group box.
chgPwdOnExpire	Change Password On Expire and Override fields in the Password group box.
chgPwdOnWarn	Change Password On Warning and Override fields in the Password group box.
cookiesrequired_page	Cookies Required Page field in the Other Pages group box.
exception_page	Exception Page field in the Error Pages group box.
expire_page	Page field in the Expire Page group box.
expirePage_ContentName	Content Name field in the Expire Page group box.
logout_page	Logout Page field in the Signon/Logout Pages group box.
mcfAuthFailure_page	MCF Auth Failure Page field in the Error Pages group box.
passwordexpired_page	Password Expired Page field in the Password group box.
passwordwarning_page	Password Warning Page field in the Password group box.
signon_page	Signon Page field in the Signon/Logout Pages group box.
signonError_page	Signon Error Page field in the Signon/Logout Pages group box.
signonresultdoc_page	Signon Result Doc Page field in the Signon/Logout Pages group box.
signontrace_page	Enable Trace Page field in the Other Pages group box.
sslrequired_page	SSL Required Page field in the Other Pages group box.
start_page	Page field in the Start Page group box.
startPageScript	Script and Override fields in the Start Page group box.
userprofile_page	User Profile Page field in the Other Pages group box.

You now specify the locale properties from the `pstools.properties` file as default language mappings on the Manage Installed Languages page. You can specify additional mappings from International Organization for Standardization (ISO) locale codes to PeopleSoft language codes by using the Web Profile Configuration - Look and Feel page.

**See Also**

Chapter 10, "Configuring the Portal Environment," Configuring Trace and Debug Options, page 240

*Enterprise PeopleTools 8.50 PeopleBook: Global Technology, "Adding New Languages"*

## Appendix D

# PeopleTools Web Libraries

This appendix provides an overview of web libraries and discusses PeopleTools web libraries.

---

## Understanding Web Libraries

A web library is a derived or work record whose name starts with WEBLIB\_. All PeopleSoft iScripts are embedded in records of this type. An iScript is a specialized PeopleCode function that generates dynamic web content. Administrators should make sure that users have the proper access to web libraries.

For example, the default navigation system for PeopleSoft Pure Internet Architecture users is implemented by using a web library. If users do not have the proper authorization to the web library and its associated scripts, then they won't have proper access to the system. If users are not authorized for a particular web library or script, then they can't invoke it. After you add a web library, you set the access for each script function individually. Invoking an iScript requires the assembly of a URL. Developers assemble the URL by using PeopleCode.

---

## PeopleTools Web Libraries

This table lists the primary PeopleTools web libraries:

<b>Web Library Name</b>	<b>Description</b>
WEBLIB_CTI	Generates the Computer Telephony Integration (CTI) console and the CTI free-seat extension page.
WEBLIB_IB	Generates PeopleSoft Integration Broker functions.
WEBLIB_MCF	Generates the Multichannel Framework (MCF) console, Agent to Customer chat window, Customer to Agent chat window, Agent to Agent chat window, and MCF tracer window.
WEBLIB_MSGWSDL	This is the record behind the page for the enterprise integration point (EIP) Web Services Description Language (WSDL) generation.
WEBLIB_NAVMAIN	
WEBLIB_OPT	Generates Optimization framework functions.

<b>Web Library Name</b>	<b>Description</b>
WEBLIB_PORTAL	<p>Contains the following six fields with FieldFormula iScript PeopleCode, each of which relate to a functional area of the portal:</p> <p>PORTAL_HOMEPAGE: Support for homepage runtime interaction, including the homepage version of the menu navigation.</p> <p>PORTAL_NAV: Main support routines for side navigation.</p> <p>PORTAL_HEADER: Support for the header portion of the page and some generic routines.</p> <p>PORTAL_DYN_TEMP: Support for the dynamic template.</p> <p>PORTAL_PGLT_PREV: Support for the pagelet preview functionality.</p> <p>PT_WSRP: Used to sign onto PeopleSoft when it is defined as a producer.</p>
WEBLIB_PPM	Generates Performance Monitor functions.
WEBLIB_PTAF WEBLIB_PTAFEMC	Generates Approval functions.
WEBLIB_PTCBD	Contains iScripts for displaying common Dashboard components.
WEBLIB_PTCXM	Contains iScripts for displaying Context Manager pagelets.
WEBLIB_PTDIAG	Contains iScripts for displaying PeopleTools Diagnostics pages.
WEBLIB_PTOBIEE	Contains iScripts for displaying Business Intelligence (Analytics) pagelets.
WEBLIB_PTPP_SC	Renders navigation pages and Main Menu navigation pagelet.
WELIB_PTPP_PGT	Renders navigation pagelets without Pagelet Wizard.
WEBLIB_PTPPB	Contains iScripts for displaying Pagelet Wizard pagelets as homepage pagelets and template pagelets
WEBLIB_PTPPB1	Contains an iScript for displaying Pagelet Wizard pagelets as WSRP portlets.
WEBLIB_PTDIAG	Generates Diagnostic framework functions.

<b>Web Library Name</b>	<b>Description</b>
WEBLIB_PT_NAV	Contains iScripts for the menu pagelet, drop-down navigation, and left-hand navigation for transaction pages.
WEBLIB_PTRTE	Contains iScripts for uploading inline images in rich text.
WEBLIB_PT_TPL	Contains iScripts for Portal template processing.
WEBLIB_QUERY	Contains the PeopleCode to generate a URL for running queries.
WEBLIB_RPT	Contains iScript for the Run report to window output option. Supports access to the new browser window.
WEBLIB_SDK	Generates Software development kit functions.
WEBLIB_SDK_BI	Contains sample code showing a call to the PeopleSoft Business Interlinks PSHttpEnable function to return information from a remote site that is formatted into the HTML display.
WEBLIB_SOAPTOCI	This is the entry point for ExcelToCI and the WSDL Discovery.
WEBLIB_TIMEOUT	Generates an inactivity timeout warning script.
WEBLIB_XMLLINK	Generates PeopleSoft Business Interlinks XML functions.



## Appendix E

# Configuring the Application for 8.4x Navigation

The 8.50 navigation is based on iframe templates and features drop-down navigation. The 8.4x navigation is based on frame templates and uses menu pagelet navigation. This appendix describes how to reconfigure an 8.50 application for 8.4x navigation.

This appendix discusses how to reconfigure the application for 8.4x navigation. To reconfigure the application you must complete the following tasks:

- Reconfigure the default template.
- Reconfigure the navigation pagelet.

---

**Important!** You must stop and restart the application and web servers after completing these tasks.

---

---

## Reconfiguring the Default Template

To reconfigure the default template:

1. Select PeopleTools, Portal, Structure and Content.
2. In the Folders group box, click Portal Objects, and then click Templates.
3. Click the Edit link for the 8.4x default template.
4. In the Field Name field, delete the existing value and enter *DUMMY*.

5. Save the component.

The template will look like this:

GeneralSecurity

Root > Portal Objects > Templates >

### Content Ref Administration

Name:PT\_84X\_DEFAULT\_TEMPLATE
\*Label:8.4x default template
Long Description:(254 Characters)8.4x default frames template
Product:PT
Sequence number:
Object Owner ID:PPT PeopleTools
Usage Type:Frame template
Storage Type:Remote by URL

Author:PTDMO
Parent Folder:Templates
Copy objectSelect New Parent Folder
\*Valid from date:05/18/2000
Valid to date:
Creation Date:05/18/2000

Add Content Reference

URL Information

\*Node NameLOCAL\_NODE
URL Type:PeopleSoft Script

iScript Parameters

\*Record (Table) Name:WEBLIB\_PT\_NAV
\*Field Name:DUMMY
\*PeopleCode Event Name:FieldFormula
\*PeopleCode Function Name:IScript\_PT\_NAV\_TPL\_FRAME
Additional Parameters:
Example: name1=value1&name2=value2

Producer:
Portlet:

Producer Details

Content Reference Attributes

Name:
Label:
Attribute value:

Delete
☒ Translate
Attribute Information

Add

#### 8.5X DEFAULT\_TEMPLATE Template after reconfiguration

6. In the breadcrumbs, click the Templates link.
7. Click the Edit link for the 8.50 default template.
8. In the Field Name field, replace *ISCRPT1* with *ISCRPT2*.

9. Change the template Usage Type from *Inline frame template* to *Frame template*.

## 10. Save the component.

The template will look like this:

GeneralSecurity

Root > Portal Objects > Templates >

Content Ref Administration

Name:

DEFAULT\_TEMPLATE

Author:

PTDMO

\*Label:

8.50 default template

Parent Folder:

Templates

Copy object

Select New Parent Folder

Long Description:  
(254 Characters)

8.50 default template

Product:

PT

\*Valid from date:

05/18/2000

31

Sequence number:

Valid to date:

31

Object Owner ID

PPT

PeopleTools

Usage Type:

Frame template

Creation Date:

05/18/2000

Storage Type:

Remote by URL

Add Content Reference

URL Information

\*Node Name

LOCAL\_NODE

URL Type:

PeopleSoft Script

iScript Parameters

\*Record (Table) Name:

WEBLIB\_PT\_NAV

\*Field Name:

ISCRIP2

\*PeopleCode Event Name:

FieldFormula

\*PeopleCode Function Name:

IScript\_PT\_NAV\_TPL\_FRAME

Additional Parameters:

Example: name1=value1&name2=value2

Producer:

Portlet:

Producer Details

Content Reference Attributes

Name:

Delete

Label:

☒ Translate

Attribute Information

Attribute value:

Add

8.5X DEFAULT\_TEMPLATE Template after reconfiguration

## Reconfiguring the Navigation Pagelet

To reconfigure the navigation template:

562

Copyright © 1988, 2009, Oracle and/or its affiliates. All Rights Reserved.

1. Select PeopleTools, Portal, Structure and Content.
2. In the Folders group box, click Portal Objects, and then click Pagelets.
3. Click the PeopleSoft Applications link, and then click the Edit link for Menu 2.0.
4. In the Field Name field, delete the existing value and enter *ISCRIP2*.
5. Change the Label field value from *Menu 2.0* to *Menu*.
6. Save the component.

The content reference will look like this:

**General** **Security**

[Root](#) > [Portal Objects](#) > [Pagelets](#) > [PeopleSoft Applications](#) >

### Content Ref Administration

**Name:** MENU **Author:** PTDMO

**\*Label:** Menu **Parent Folder:** PeopleSoft Applications  
[Copy object](#) [Select New Parent Folder](#)

**Long Description:** (254 Characters) 8.50 menu

**Product:** PT

**Sequence number:**

**Object Owner ID:** PPT [PeopleTools](#)

**Usage Type:** Pagelet **Creation Date:** 05/15/2000

**Storage Type:** Remote by URL ☐ WSRP Producible

[Add Content Reference](#)

---

**URL Information**

**\*Node Name:** LOCAL\_NODE

**URL Type:** PeopleSoft Script

**iScript Parameters**

**\*Record (Table) Name:** WEBLIB\_PT\_NAV **\*Field Name:** ISCRIP2

**\*PeopleCode Event Name:** FieldFormula **\*PeopleCode Function Name:** IScript\_PT\_NAV\_PAGELET

**Additional Parameters:** templatype=html

Example: name1=value1&name2=value2

MENU 2.0 pagelet after reconfiguration

7. In the breadcrumbs, click the PeopleSoft Applications link.
8. Click the Edit link for Menu.
9. In the Field Name field, replace *ISCRIP2* with *ISCRIP1*.

10. Save the component.

The content reference will look like this:

GeneralSecurity

Root > Portal Objects > Pagelets > PeopleSoft Applications >

Content Ref Administration

Name:PT\_84X\_MENU

\*Label:Menu

Long Description:  
(254 Characters)8.4x menu pagelet

Product:PT

Sequence number:

Object Owner IDPPTPeopleTools

Usage Type:Pagelet

Storage Type:Remote by URL

Author:PTDMO

Parent Folder:PeopleSoft Applications

Copy objectSelect New Parent Folder

Creation Date:05/15/2000

☐ WSRP Producible

[Add Content Reference](#)

URL Information

\*Node NameLOCAL\_NODE

URL Type:PeopleSoft Script

iScript Parameters

\*Record (Table) Name:WEBLIB\_PT\_NAV

\*Field Name:ISCRIP1

\*PeopleCode Event Name:FieldFormula

\*PeopleCode Function Name:IScript\_PT\_NAV\_PAGELET

Additional Parameters:

Example: name1=value1&name2=value2

Menu after reconfiguration

## Appendix F

# Converting Frames-Based Templates to iFrame-Based Templates

This appendix provides a detailed example of converting an existing frames-based template (eProcurement Shopping Cart template) to the 8.50 drop-down navigation template.

When you convert a frames-based template to an iframe-based template, it enables you to take advantage of the new features in the iframe, drop-down navigation template: related content; AJAX-based, drop-down navigation; modal, key-specific favorites improvements; and the recently used menu functionality.

---

**Important!** The following conversion example assumes the current frames-based template that you are converting uses a storage type of *Local in HTML (catalog)*. If your frames template is a *Remote by URL* template, your steps will be slightly different.

---

---

## Converting the eProcurement Shopping Cart Template

This section describes how to:

- Gather information about the current template.
- Create the iframe template iScript.
- Create the iframe template HTML definition.
- Set security for the template iScript.
- Convert the template content reference.
- Troubleshoot errors.

## Gathering Information About the Current Template

To gather information about the current template:

1. Select PeopleTools, Portal, Structure and Content.
2. Within Structure and Content, click Portal Objects, then click Templates.
3. Click the Edit link next to the ePro Shopping Cart Template.
4. Copy the text in the HTML Area field. Save it to a text file. After you have changed the template type this HTML will be removed.

5. Click the Security tab and note the current security for this content reference.

This step is important if there are specific permissions or roles that have been set. Later, you apply these permissions to the new iScript that will be created.

## Creating the iFrame Template iScript

To create the iframe template iScript:

1. Open the WEBLIB\_PV.ISCRIPT.FieldFormula PeopleCode in edit mode.
2. At the bottom of the PeopleCode, enter the following code:

```
Declare Function ptIframeTemplateHNav PeopleCode WEBLIB_PT_NAV.ISCRIPT1 Field=>
Formula;

/*
 * builds the inline frame template with the drop-down navigation
 */
Function IScript_PVShopCart_Template

    /* parameter 1: HTML object name that will be used to generate the inline frame=>
    template */
    %Response.Write(ptIframeTemplateHNav( "PV_SHOP_CART_HNAV_TEMPLATE" ));

End-Function;
```

---

**Note.** For *Remote by URL* templates, use the existing web library iScript and just modify the PeopleCode as seen above.

---

## Creating the iFrame Template HTML Definition

To create the iframe template HTML definition:

1. In Application Designer, open the PT\_HNAV\_TEMPLATE HTML definition.
2. Clone the definition and save it as the the value that was used in parameter 1 of the new iScript.
3. Open the text file that contains the original template HTML and locate the pagelets that you need to move to the new template, for example:

```
<frame name="CART" id="CART" frameborder="no" scrolling="auto" noresize src="">
  style="border-top:ridge #CCCCCC 5px;">
    <IClientComponent Name="Links">
      <Source Node="ERP" Pagelet="EP_PV_REQ_LINKS_PGLT">dummy</Source>
    </IClientComponent>
</frame>
```

4. Change the HTML so that it looks like this:

```
<div id="pvcartpagelet">
  <iframe name="CART" id="CART" frameborder="no" src="">
    <pagelet name="Links">
      <source Node="ERP" Pagelet="EP_PV_REQ_LINKS_PGLT"/>
    </pagelet>
  </iframe>
</div>
```

---

**Note.** The iframe template does not support the `IClientComponent` template meta tag. These meta tags are deprecated and are replaced by the `pagelet` meta tag.

---

5. Locate where you want to position the Links pagelet in the `PV_SHOP_CART_HNAV_TEMPLATE` iframe template. If the pagelet is to exist to the right or left of the target content, place the pagelet inside of the iframe content container `<div>`. In this scenario, the pagelet will be located to the right of the target content.

```
<div id="ptifrmcontent">
  <div id="ptifrmtarget">
    <iframe id="ptifrmtgtframe" name="TargetContent" title="%bind(:6)" frameborder=>
=>
=>
"0" scrolling="auto" onload="ptrc.onLoadTC()" src="">
      <target Name="target" params="%bind(:7)"/>
    </iframe>
  </div>
  <div id="pvcartpagelet">
    <iframe name="CART" id="CART" frameborder="no" src="">
      <pagelet name="Links">
        <source Node="ERP" Pagelet="EP_PV_REQ_LINKS_PGLT"/>
      </pagelet>
    </iframe>
  </div>
  %bind(:8)
  <script type="text/javascript" src="%bind(:9)"></script>
</div>
```

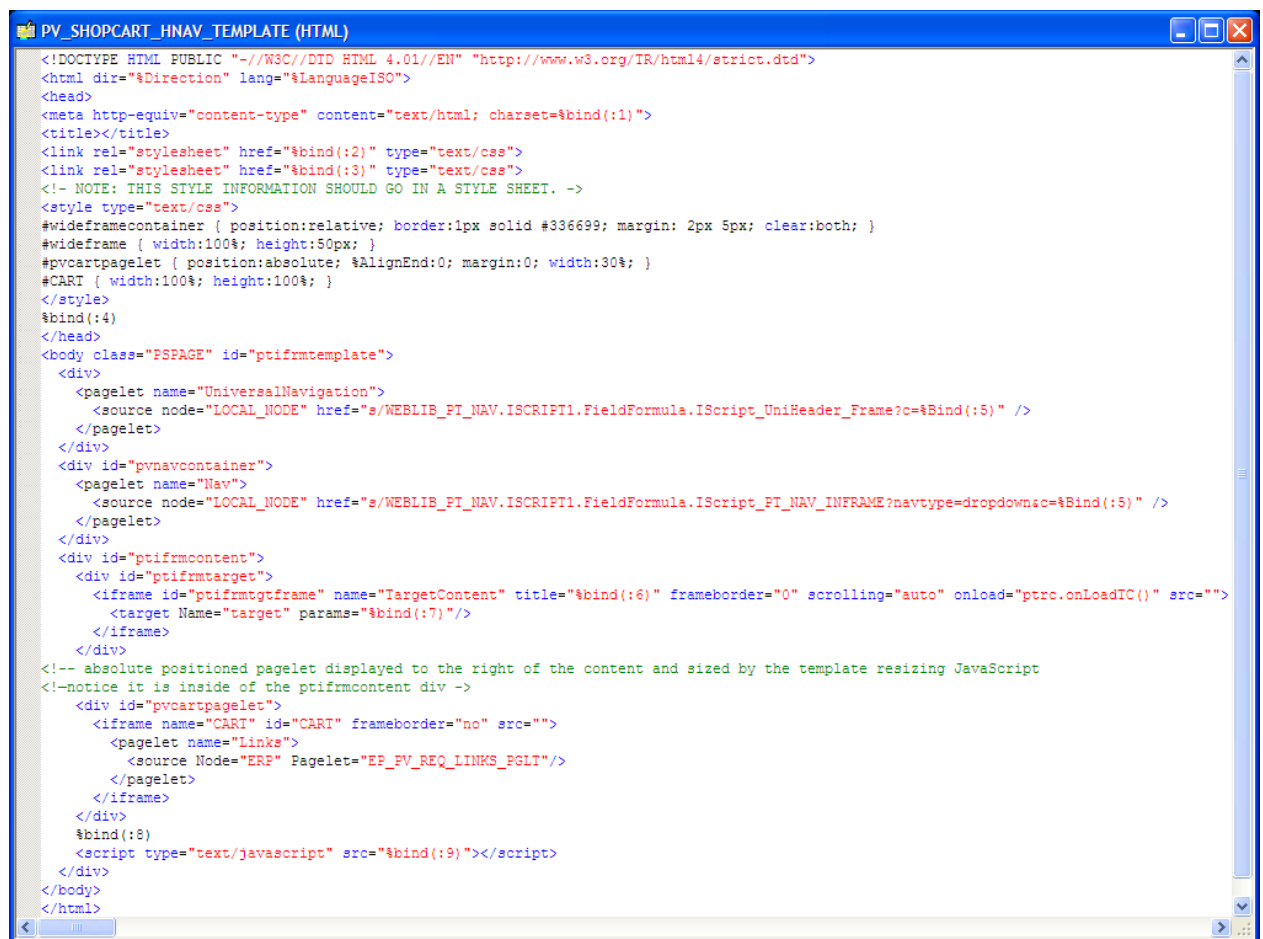
6. Add the style information for the new pagelet. Ideally, these styles should go into a style sheet to take advantage of style sheet caching on the web server and browser. For this example, the embedded style approach places the style in the head section of the template.

```
<style type="text/css">
/* style for the CART pagelet container div */
#pvcartpagelet {
    position:absolute; /* required: the position must be fixed so the iframe can be=
    sized properly */

    %AlignEnd:0;          /* meta CSS to position the location of the pagelet */
    margin:0;
    width:30%;            /* required: must be a percentage, value so the correct width=
    is rendered when the browser is resized */
}

/* style for the CART iframe */
#CART { width:100%; height:100%; }
</style>
```

This is the full HTML definition:



Test HTML object: PV\_SHOPCART\_HNAV\_TEMPLATE with shopping cart pagelet

If a pagelet needs to expand to the width of the page, the following HTML and style information can be used:

```

<style type="text/css">
#wideframecontainer {
    position:relative;                /* required: */
    border:1px solid #336699;
    margin: 2px 5px;
    clear:both;
}
#wideframe { width:100%; height:50px; }
</style>
...
<!--must be located outside of the content container div-->
<div id="wideframecontainer">
    <iframe name="wide" id="wideframe" frameborder="no" src="">
        <pagelet name="Test">
            <source Node="LOCAL_NODE" href="s/WEBLIB_PV.ISCRIPT1.FieldFormula.IScript_>
PT_TEST"/>
        </pagelet>
    </iframe>
</div>

```

This is the full HTML definition that contains a wide pagelet:

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html dir="%$Direction" lang="%$LanguageISO">
<head>
<meta http-equiv="content-type" content="text/html; charset=%bind(:1)">
<title></title>
<link rel="stylesheet" href="%bind(:2)" type="text/css">
<link rel="stylesheet" href="%bind(:3)" type="text/css">
<!-- NOTE: THIS STYLE INFORMATION SHOULD GO IN A STYLE SHEET. -->
<style type="text/css">
#wideframecontainer { position:relative; border:1px solid #336699; margin: 2px 5px; clear:both; }
#wideframe { width:100%; height:50px; }
#pvcartpagelet { position:absolute; %AlignEnd:0; margin:0; width:30%; }
#CART { width:100%; height:100%; }
</style>
%bind(:4)
</head>
<body class="PSPAGE" id="ptifrmtemplate">
<div>
<pagelet name="UniversalNavigation">
<source node="LOCAL_NODE" href="/s/WEBLIB_PT_NAV.ISCRIPT1.FieldFormula.IScript_UniHeader_Frame?c=%bind(:5)" />
</pagelet>
</div>
<div id="pvnavcontainer">
<pagelet name="Nav">
<source node="LOCAL_NODE" href="/s/WEBLIB_PT_NAV.ISCRIPT1.FieldFormula.IScript_PT_NAV_INFRAME?navtype=dropdown&c=%bind(:5)" />
</pagelet>
</div>
<!-- relative positioned pagelet which gets the entire width of the page and is resized automatically by the browser -->
<!--notice it is outside of the ptifrmcontent div -->
<div id="wideframecontainer">
<iframe name="wide" id="wideframe" frameborder="no" src="">
<pagelet name="Test">
<source Node="LOCAL_NODE" href="/s/WEBLIB_FV.ISCRIPT1.FieldFormula.IScript_PT_TEST"/>
</pagelet>
</iframe>
</div>
<div id="ptifrmcontent">
<div id="ptifrmtarget">
<iframe id="ptifrmtgtframe" name="TargetContent" title="%bind(:6)" frameborder="0" scrolling="auto" onload="ptro.onLoadTC()" src="">
<target Name="target" params="%bind(:7)"/>
</iframe>
</div>
<!-- absolute positioned pagelet displayed to the right of the content and sized by the template resizing JavaScript -->
<!--notice it is inside of the ptifrmcontent div -->
<div id="pvcartpagelet">
<iframe name="CART" id="CART" frameborder="no" src="">
<pagelet name="Links">
<source Node="ERP" Pagelet="EP_FV_REQ_LINKS_PG1T"/>
</pagelet>
</iframe>
</div>
%bind(:8)
<script type="text/javascript" src="%bind(:9)"/></script>
</div>
</body>
</html>

```

Test HTML object: PV\_SHOPCART\_HNAV\_TEMPLATE with shopping cart and wide pagelet

## Setting Security for the Template iScript

To set security for the template iScript:

1. Select PeopleTools, Security, Permissions & Roles, Permission Lists.
2. Enter the permission list that you noted when you gathered information about the template earlier.
3. Enable Web Library security for the permission list.
4. Continue until all permission lists and roles are added.

---

**Note.** If the template content reference security is public, only one permission list is needed.

---



---

**Note.** For *Remote by URL* templates, this step is optional, because the iScript already exists.

---

## Converting the Template Content Reference

To convert the template content reference:

1. Select PeopleTools, Portal, Structure and Content.
2. Within Structure and Content, click Portal Objects, then click Templates.
3. Click the Edit link next to the ePro Shopping Cart Template.
4. Change the Usage Type to *inline frame template*.
5. Change the Storage Type to *Remote By URL*.
6. Enter the iScript record, field, and PeopleCode function names.
7. Click the Save button.
8. Stop and restart the application and web servers.
9. Test the template.

---

**Note.** For *Remote by URL* templates, omit steps 5, 6, and 7.

---

This is the newly configured eProcurement application in an iframe-based template:

## Troubleshooting Errors

This table describes some common errors and possible causes:

<b>Error</b>	<b>Solution</b>
JavaScript errors	Check existing pagelet JavaScript. Make sure to correct statements like this: <pre>parent.parent.frames["UniversalHeader"] /* no longer valid in an iframe template */ parent.frames[3]. /* use a iframe name or an iframe id */</pre>
Display errors	



## Appendix G

# Configuring OBIEE Related Content Services

You can configure related content services to display OBIEE analytics reports. The OBIEE servlet must be installed and OBIEE must be configured to enable OBIEE related content. In addition to configuring the analytics servlet, there are prerequisites that must be fulfilled before assigning OBIEE related content to application pages.

---

## OBIEE Configuration Prerequisites

When using OBIEE analytics as related content services, note the following configuration requirements:

- The OBIEE database needs to be on release 10.1.3.3 or greater.
- The PeopleSoft application database and the OBIEE database must be configured with the same users to access the reports.
- XML needs to be generated for the OBIEE report definition that you make available as related content. To automatically generate the XML, navigate to the advanced tab to on the OBIEE report definition and save the report.
- The sizing of reports need to be handled on OBIEE database. Pending further investigation, only the chart object can be resized. The associated grid, width of the page, and the centering of the objects can not be adjusted (other then the OBIEE dashboard itself).
- Set up the Integration Broker Gateway on the PIA database.
- Encrypt the *secureFileKeystorePasswd*.
  1. Select PeopleTools, Integration Broker, Configuration, Gateways and click the Gateway Setup Properties link.
  2. Enter *administrator* in the User ID field and *password* in the Password field and click the OK button.
  3. Click the Advanced Properties Page link and search for *secureFileKeystorePasswd*.
  4. Copy the password and encrypt it using the Password Encryption Utility at the bottom of the page.
  5. Replace the existing plain text password with the encrypted value.
  6. Click the OK button.
  7. Save the page and click the OK button.

- Deploy the analytics.war file from the OBIEE server on the PIA web server.  
,
- 1. Remove the file from the <obiee server>\OracleBI\web directory.
- 2. Copy the file to a temporary directory, open a command window and point to the temporary directory.
- 3. Set the PATH environment variable with the JDK bin directory path.
- 4. Extract the Java files from analytics.war with "jar -xvf analytics.war".
- 5. Once extracted, open the resulting web.xml file from the WEB-INF directory and enter the OBIEE server and port information.
- 6. Recompile the files back into an analytics.war file using "jar -cvf analytics.war <path>."
- 7. Deploy the new analytics.war file on the web server.
- Test the deployment by entering this URL in your browser: *http://<webservername:port>/analytics*.
- After installing analytics.war, update the OBIEE server setup with the weblogic URI: *http://<webservername:port>/analytics* and re(load) the report definitions.

## Appendix H

# Related Content Reserved Words and Service Parameters

When you have a service using any of the following reserved words as service parameters, these will have some default mapping done when you first time access the Map Fields page. Here is the list of supported related content specific reserved word service parameters and their default mapping data.

---

## Reserved Words Default Mapping

This table lists the reserved words and their default mapping:

<b><i>Reserved Word</i></b>	<b><i>Description</i></b>	<b><i>Default Map Type</i></b>
PTCS_CREFNAME	Current content reference (cref) name for which the service assignment and mapping is being done.	Fixed value.
PTCS_CREFLABEL	Current related content transaction cref label.	Fixed value.
PTCS_NODENAME	%Node.	System variable.
PTCS_PORTALNAME	%Portal.	System variable.
PTCS_KEYFIELD1	Component level 0 key1 if available.	Key field.
PTCS_KEYFIELD2	Component level 0 key2 if available.	Key field.
PTCS_KEYFIELD3	Component level 0 key3 if available.	Key field.
PTCS_KEYFIELD4	Component level 0 key4 if available.	Key field.
PTCS_PAGENAME	%Page.	System variable.
PTCS_SERVICEID	Current related content service ID.	Fixed value.
PTCS_RELURL	Relative URL of the content reference.	Fixed value.



# Index

## Numerics/Symbols

<%=resultDoc%> session variable 248

## A

accelerators, SSL *See* SSL accelerators

action bars

creating 168

hiding 156

HTML objects for 172

Add Folder page 68

Add Link page 74

addressing, virtual *See* virtual addressing

anchor tags 39

API

Java Portlet 540

application classes, pagelet search 298

Application Designer

changing menu definitions 113

creating/storing application objects 8

designing pagelets based on PIA pages 425

inheriting style classes 194

understanding 8

viewing iScripts 184

Application Engine programs

PORTAL\_CSS 133

PORTAL\_INDEX 150

PORTAL\_MENU

*See Also* Menu Import process

application objects 8

application server domains

server processes *See Also* server processes

understanding 9

application servers

configuring 20

database connectivity software 15

domains *See Also* application server domains

handlers handlers

installing Process Scheduler 16

Jolt *See Also* Jolt

listeners listeners

managing memory consumption 206

processes *See Also* server processes

queues queues

relationship to database servers 8

server processes *See Also* server processes

services services

Tuxedo Tuxedo

understanding 9, 16

ASF 493

Assign Related Content page 471

authentication domains

defining 214

selecting for portals 221

understanding 214

authentication token *See* PS\_TOKEN

AuthTokenDomain 235

## B

base-level authentication domains 214

batch servers 15

bind variables

displaying error messages 246

using for images, tabs and pagelets 175

using in templates 182, 187, 191

using the PORTAL\_HP\_COMPONENT

object 168

breadcrumbs 52

browsers

applying timeouts to individual windows 237

browser-based caching 207

caching 199

compressing communication with web

servers 222

configuring caching 236

setting state options 210, 237

understanding 18

BuildStaticHPTemplate function 168

Business Interlinks

using 22

Business Interlinks web library 557

## C

Cache Enabled Value 58

caching

browser-based caching, understanding 207

browsers, configuring on 235

compressing cache files 222

configuring directories 239

configuring for portals 6, 199, 550

configuring on the web server 238

homepage 208

homepage, default 200

JSR 168 portlet content 540

metadata 205

navigation pagelet 209

page 210

pagelet *See Also* pagelet caching

pagelet refreshing, disabling 163

proxied JavaScript 207

server-based 200

target content 117, 201

understanding 199

user IDs 226

web server-based navigation 211, 238

web server memory 201

caching, enabling for navigation pages 47

cascading style sheet (CSS) *See* style sheets

certificates, installing 508

character set encoding

overriding for content references 117

overriding for portal pages 259

Clean Portal Project page 92

Clean Portal Project process

running 92

- understanding 89
  - collaborative content 451
  - colors
    - changing for links 178
    - enabling color attributes in table layouts for pages 241
  - COM 28
  - common fields
    - display options 457
    - service security options 458
    - service URL parameters 457
    - URL information 456
  - Component Interfaces 22
  - Component Object Model (COM) 28
  - components
    - adding default 168
    - including in portal navigation 263
    - registering in multiple portals 134
    - registering multiple times 134
    - URL format for 34, 36
  - Computer Telephony Integration (CTI) console web library 555
  - connector 290
  - consumer properties 523
  - consuming remote portlets
    - language support for 545
  - Content Home navigation pagelet 49
  - Content Ref Administration page 526, 535
    - adding/editing templates 141
    - content references, defining 112
    - content references/folders, moving 131
    - pagelet attributes, managing 161
    - pagelet Help, configuring 164
    - pagelet timeouts, configuring 164
    - tabs, adding 154
    - tabs, creating 173
  - Content Reference Link Administration page 125
  - content reference links
    - copying attributes 128
    - defining 125
    - defining attributes 128
    - deleting 127
    - setting link information 127
    - setting security 128
    - setting target information 127
    - understanding 126
    - using URLs for 126
  - content references
    - accessing 31
    - adding internal PeopleSoft 113
    - copying 130
    - creating 524
    - creating content reference groups 120
    - defining 112
    - deleting 124
    - links *See Also* content reference links
    - moving 131
    - registering 31
    - related links *See Also* related links
    - reviewing 526
    - reviewing permission list security 123
    - reviewing role security 124
    - reviewing security 123
    - setting attributes 116
    - setting security 122
    - setting URL information 115
    - specifying pagelet caching 203
    - synchronizing portal object security 132
    - testing 124
    - type parameters 117
    - understanding 27
    - using the KEYWORD attribute 150
    - viewing for a folder 145
    - viewing for WSRP portlets 535
    - viewing permission list, role and user access 147
  - Content Reference Security page 122
  - content servlets
    - accessing unwrapped content 35
    - proxying and relative URLs 37
  - context manager 451
  - Context Manager Home custom navigation page 46
  - Context Manager Home navigation page 49
  - cookies
    - allowing session cookies 19
    - authentication 18
    - checking for duplicate 251
    - configuring rules 234
    - configuring rules for PeopleSoft portal 550
    - deleting 235
    - passing to servers 234
    - proxying 235
    - PS\_TOKEN *See Also* PS\_TOKEN
    - securing 235
    - selecting the cookies required page 247
    - sharing 214
  - Copy Related Content Service Definition page 469
  - CSS *See* style sheets
  - CTI console web library 555
  - custom navigation pages
    - describing 46
    - describing folder and link sequence on 47
- ## D
- databases
    - application server domains
      - See Also* application server domains
    - connectivity software 15
    - defining portals 102
    - importing menus 261
    - servers *See Also* database servers
    - supporting portal registries 28
  - database servers
    - configuring 20
    - logical server configuration 20
    - understanding 8
  - data source
    - selecting an IB 316
    - selecting a URL 323
    - selecting IB connector 315
    - setting up default gateway 290
    - setting up Pagelet Wizard IB 291
    - setting up security 290
    - specifying IB Connector parameters 330
    - specifying URL parameters 337
  - debugging options, configuring 240, 552
  - Define IB Connector Security - Connectors page 290
  - Define Related Content Service page
    - non-PeopleSoft URL type 460
    - PeopleSoft component URL type 461
    - PeopleSoft embeddable pagelet URL type 462
    - PeopleSoft OBIEE pagelet URL type 463

- PeopleSoft Query URL type 464
- PeopleSoft script URL type 465
- definitions
  - application object 8
  - menu 113, 260
  - menu group 263
  - node 25, 255
  - portal *See Also* portal definitions
- Delete Empty Folders page 91
- Delete Empty Folders process
  - running 91
  - understanding 89
- Delete Registry Override
  - defining at registry level 61
- DEV web profile 216
- Diagnostic Framework web library 556
- disconnect URL format 35
- display options, fields 457
- DNS 215
- Domain Name Server (DNS) 215
- domains
  - application server
    - See Also* application server domains
  - authentication authentication domains
- drop-down navigation
  - example 23
- dynamic templates
  - frame-based example 184
  - iframe-based example 188
  - understanding 182

## E

- Edit Collection 52
- Edit folder 52
- Edit Link page 78
- EIP WSDL web library 555
- Enable Drill Down Cache 58
- encryption and signature 508
- encryption enabling 508
- Enterprise Portal
  - importing menus 261
- errors
  - accessing content references 31
  - checking for duplicate cookies 251
  - configuring error pages 245
  - displaying/hiding for pagelets 197
  - selecting error pages for unavailable pagelets 245
  - selecting error pages for unavailable target content 245
  - selecting pages for MCF authorization failure 245
  - selecting pages for user authentication failure 248
  - selecting signon error pages 248
  - sending PUT requests to portal servlets 29
  - setting authentication domains 214
  - setting target parameters for template pagelets 193
  - setting the base physical path 224
  - understanding causes 245
  - unlocking web profile accounts 217
  - using SSL accelerators 265
- Example for Images page 50
- exceptions

- logging for Jolt 241
- selecting pages for Java exceptions 245
- using mutual authentication 251
- expiration page 245
- extended authentication domains 214
- external URL format 34

## F

- favorites, adding 27
- file URL format 35
- Find an Existing Collection page 63
- firewalls
  - configuring web servers 21
  - using with reverse proxy servers 269
- Folder Administration page 105
- folders
  - copying 130
  - defining 105
  - defining attributes 107
  - deleting 107
  - enabling navigation 104
  - moving 131
  - organizational structure 100
  - Portal Objects folder
    - See Also* Portal Objects folder, Portal Objects folder
  - root folder 28
  - setting menu visibility 148
  - setting security 108
  - synchronizing portal object security 132
  - understanding 27
  - viewing menu folder structure information 145
  - viewing permission list, role and user access 147
- Folder Security page 108
- forms
  - combining with frames 197
  - converting form tags 39
  - posting data from frame-based templates 196
- frame-based templates
  - constructing URL references 195
  - dynamic example 184
  - handling 30
  - posting form data 196
  - static example 183
  - understanding 182
- frames
  - browser caching 207
  - combining with forms 197
  - frame-based templates
    - See Also* frame-based templates, frame-based templates
  - iframe-based templates
    - iframe-based templates, iframe-based templates
  - using single signon 258
- frames-based templates
  - converting to iframes 565
- functions
  - BuildStaticHPTemplate 168
  - displaying JavaScript used in pages 241
  - forcing refreshes 179
  - generating URL references 195
  - web libraries 555

## G

Gateways page 518  
 General Settings page 136  
 GenerateXXXXPortalRelURL function 195  
 GenerateXXXXPortalURL function 195  
 global variables 214

## H

handlers  
   Jolt server 12  
   understanding 10  
   workstation 12  
 headers  
   custom response 195  
   hiding pagelet buttons 163, 205  
   HTTP header caching information 208  
   PortalRegisteredURL 195  
   universal navigation header  
     *See Also* universal navigation header  
   using custom portal 177  
   using UsesPortalRelativeURL 37  
 Help  
   configuring for pagelets 164  
   creating pagelet 117  
   text for related content 467  
 help, online  
   accessing homepage help 156, 162  
   setting the URL 222  
 Homepage Generation Service 175  
 homepages  
   adding default components 168  
   administering 153  
   caching 208  
   caching, default 200  
   caching on browsers 238  
   constructing 169  
   constructing headers 171  
   creating tabbed 153  
   creating three-column 168  
   creating two-column 168  
   Homepage Generation Service 175  
   inserting custom tab images 117  
   navigating 23  
   opening 26  
   overriding images/objects 117, 173  
   tabs *See Also* tabs  
   URL format for 35  
   using predefined HTML objects 169  
 hosts file  
   configuring for firewalls 270  
   configuring for load balancing 273  
   configuring for SSL accelerators 272  
 HTML  
   areas 182  
   caching target content 238  
   catalog 140  
   content supported for content references 117  
   generating for testing 241  
   objects *See Also* HTML objects  
   pagelets pagelets  
   template pagelets template pagelets  
   templates templates  
   using tables for page layout 182

  viewing/debugging for pages 242  
 HTML objects  
   available 168  
   constructing headers 171  
   constructing pagelets 172  
   creating tabs 173  
   overriding default 117, 173  
   using predefined homepage 169  
   using predefined template 167  
 HTTP\_PROXY\_NONPROXY\_HTTPHOSTS  
   environment variable 269  
 HTTP\_PROXY\_NONPROXY\_HTTPSHOSTS  
   environment variable 269  
 HTTPS  
   improving same-server performance under  
     SSL 264  
   overriding the protocol for URL references  
     233  
   specifying for portals 227  
   using SSL accelerators 265  
 HTTPS protocol  
   using to communicate 506  
 hyperlinks *See* links

## I

IB Integration Broker  
   selecting a data source 316  
   setting up Pagelet Wizard 291  
   setting up security 290  
 IB Connector  
   defining security 290  
   selecting data source 315  
   setting up security 290  
   specifying data source parameters 330  
 IBM WebSphere *See* WebSphere  
 iframe 182  
 iframe-based templates  
   converting from frames-based templates 565  
   dynamic example 188  
   handling 30  
 images  
   adding to tabs 175  
   configuring cache directories 239  
   inserting on homepage tabs 117  
   objects 173  
   overriding default homepage 117, 173  
 images, navigation collection 50  
 Import Producer Information page 520  
 indexes, registry search 149  
 information, producer 520  
 inline frames 182  
 installing certificates 508  
 Integration Broker, using 22  
 Integration Broker web library 555  
 International Standards Organization (ISO) locale  
   codes 249  
 Internet scripts *See* iScripts  
 internet technology portal technology  
 iScripts  
   configuring start pages 244  
   designing pagelets based on 426  
   navigation 37  
   setting parameters for content references  
     115, 141  
   setting parameters for pagelets 163

- understanding web libraries 555
- using dynamic templates 182
- using frame-based templates 184
- using iframe-based templates 188

ISO locale codes 249

## J

### Java

- applying load balancing 273
- controlling proxy server settings 268
- encrypting passwords (PSCipher) 219
- JavaScripts *See Also* JavaScripts
- JSR 168 JSR 168
- portlets 539
- selecting pages for exceptions 245
- servlets *See Also* Java servlets
- setting Java Virtual Machine properties 268
- web servers *See Also* web servers

### Java Portlet 493

### Java Portlet API 1.0 540

### JavaScripts

- caching proxied 207, 238
- configuring cache directories 239
- converting URL references 40
- displaying functions used in pages 241
- setting authentication domains 214

### Java servlets

- portal servlets *See Also* portal servlets
- understanding 17

### JCP 493

### Jolt

- logging exceptions 241
- server listeners/handlers 12
- setting web server options in web profiles 231
- transmitting requests and data 17
- understanding 15

### JSH 12

### JSL 12

### JSR 493

### JSR 168 493

- configuring the portal container 540
- describing the portlet entity registry 541
- Java Portlet API 540
- PeopleSoft compliance 539
- portlets *See Also* JSR 168 portlets
- setting up 6
- understanding 539

### JSR 168 portlets

- caching content 540
- understanding 539
- understanding modes 540
- window states 540

### JVM 268

## K

### Keystore Password 509

### keywords

- associating to content references 150
- using keywords to search for CREFs 116

### KIOSK web profile 217

## L

### labels

- labelling content references 116
- labelling folders 107
- labelling portals 101
- labelling tabs 155, 156

### language support 249

- rules for consuming and producing remote portlets 545

### Large Collection Image

- defining at registry level 61
- defining at system level 59

### Large Content Image

- defining at registry level 61
- defining at system level 59

### Large Folder Image

- defining at registry level 61
- defining at system level 59

### LDAP

- using mutual authentication 251

### libraries

- web 555

### licensing

- homepages 153
- portal technology products xix

### links

- changing colors 178
- content reference
  - See Also* content reference links
- proxying 38
- related *See Also* related links
- XML 231

### listeners

- Jolt server 12
- understanding 10
- workstation 12

### load balancing

- applying 272
- overriding the port/server name for URL references 233

### Local Gateway URL 518

## M

### Main Menu Collection Name 54

### Manage Related Content Configuration page 470, 475

### mapping service parameters 473

### mashup 451

### Maximum Child Links 55

### Maximum Columns 55

### MCF

- selecting pages for authorization failure 245

### MCF web library 555

### Menu Folder Structure page 145

### Menu Import page 261

### Menu Import process

- importing menu groups 261
- monitoring the status 263
- selecting components to include in portal navigation 263
- understanding 260
- using with Enterprise Portal 261

### Menu Item Details page 143

- menus
  - enabling navigation 104
  - importing groups into the portal registry
    - See Also* Menu Import process
  - navigation 26
  - reviewing item details 143
  - setting visibility 148
  - viewing associated permission lists 149
  - viewing folder structure information 145
  - viewing permission lists, roles and users with
    - menu item access 145
  - viewing security details 147
- Menu Security page 147
- message signature, enabling 509
- metadata
  - caching 205, 238
  - understanding 8
- Microsoft Windows *See* Windows
- Move to New Parent Folder 74
- Multi-channel Framework (MCF) *See* MCF

## N

- navigation
  - drop-down 23
  - enabling for folders 104
  - iScripts 37
  - selecting components to include 263
  - understanding 26
  - URL format 35
  - web server-based caching 211, 238
- navigation collection options
  - understanding 53
  - understanding default values for 53
- navigation collections
  - adding folders to 68
  - adding links to 74
  - creating 64
  - editing 64
  - editing folders in 73
  - editing links in 78
  - maintaining 61
  - publishing 80
  - publishing as a page 49
  - publishing as a pagelet 49
  - searching for 63
  - selecting a source folder 71
  - selecting a source link 76
  - selecting images for 72
  - understanding 48
  - using images 50
- navigation pagelets
  - caching 209
  - enabling folder navigation 104
- navigation pagelets web library 556
- navigation pages
  - describing common elements on 43
  - describing common elements used on 52
  - enabling caching for 47
  - understanding 43
  - using breadcrumbs on 43
- navigation pages web library 556
- navigation web library 557
- Node Definitions page 257
- Node Name 58
- nodes
  - applying templates for partial path 196
  - configuring node-specific template
    - information 138
  - setting 255
  - specifying types 257
  - understanding 25, 255
- non-PeopleSoft URL types
  - creating 460
  - example 461
  - service URL parameters 461
  - URL information 461
- numeric message variables 246

## O

- OBIEE
  - configuring for related content 573
- OBIEE pagelets
  - related content 463
- objects
  - application 8
  - Component Object Model (COM) 28
  - content references *See Also* content references
  - folder navigation 101
  - HTML *See Also* HTML objects
  - image 173
  - PeopleTools 106
  - portal *See Also* portal objects
  - Portal Objects folder Portal Objects folder
- online help *See* help, online
- Optimization Framework
  - PSOPTENG server process 14
- Optimization Framework web library 555
- options 53
- Oracle Jolt *See* Jolt
- Oracle Tuxedo Tuxedo
- Oracle WebLogic WebLogic
- Owner ID 58

## P

- page-based templates
  - assembling pages 30
  - handling 29
  - proxying 38
  - static example 183
  - understanding 182
- pagelet caching
  - caching personalized pagelets 204
  - disabling manual pagelet refresh 205
  - implementing 117
  - specifying in content references 203
  - specifying in templates 202
  - understanding 202
- pagelet display options, search box 343
- pagelets
  - accessing online help 162
  - action bars *See Also* action bars
  - administering 153
  - building 388
  - caching *See Also* pagelet caching
  - configuring Help 164
  - configuring timeouts 164
  - constructing 167, 168, 172

- description 388
- designing based on iScripts 426
- designing based on PIA pages 425
- displaying/hiding error messages for 197
- extensions 389
- forcing the refresh 163
- hiding the minimize button 163
- hiding the refresh button 163
- homepages, deleting from 160
- homepages, selecting for 157
- managing attributes 161
- Pagelet tag 31
- personalization page, selecting the 163
- publishing to another portal 86
- publishing to multiple portals 84
- publishing to other portals 83
- running location 449
- selecting error pages for unavailable pagelets 245
- setting display options 158
- setting names in templates 196
- setting the number of columns 162
- specifying layout 159
- template *See Also* template pagelets
- understanding URLs 36
- using refresh tags 449
- pagelet search
  - application classes 298
  - setting default options 343
- Pagelet tag 31
- Pagelet Wizard
  - copying pagelets 379
  - defining pagelet categories 308
  - defining transformation types 292
  - defining XSL prototypes 300
  - deleting pagelets 379
  - describing display sizes 286
  - generating pagelet export and import scripts 382
  - overview 277
  - publishing a pagelet to multiple portals 84
  - publishing pagelets to another portal 86
  - publishing pagelets to other portals 83
  - reviewing pagelet details 384
  - setting up data source 291
  - setting up default gateway 290
  - setting up IB connector data source security 290
  - setting up the Integration Broker 290
  - specifying pagelet information 310
  - specifying publication options 365
- Pagelet Wizard data source parameters
  - specifying 324
  - specifying for Navigation Collections 332
  - specifying for queries 334
  - specifying for search records 336
  - using system variables as 325
- Pagelet Wizard data sources
  - describing 278
  - selecting 312
  - selecting a Navigation Collection 317
  - selecting a query 319
  - selecting a rowset 320
  - selecting a search record 321
  - selecting HTML 313
- Pagelet Wizard data types, defining 292, 296
- Pagelet Wizard display formats
  - defining 292, 294
  - describing 280
  - describing the chart format 281
  - describing the custom format 285
  - describing the list format 281
  - describing the menu format 282
  - describing the passthru format 284
  - describing the search list format 283
  - describing the table format 280
  - selecting 340
- Pagelet Wizard display options
  - specifying 342
  - specifying data links 362
  - specifying for charts 356
  - specifying for customs 360
  - specifying for lists 355
  - specifying for menus 352
  - specifying for passthru 349
  - specifying for search lists 351
  - specifying for tables 346
- Pagelet Wizard footers
  - defining 302, 306
  - defining images 307
- Pagelet Wizard headers
  - defining 302, 303
  - defining images 305
- Pagelet Wizard pagelet types
  - describing 286
  - describing the embeddable type 288
  - describing the homepage type 286
  - describing the template type 287
- pagelet wizard portlets web library 556
- Pagelet Wizard post-transformation processing
  - formatting currency elements 374
  - formatting date elements 374
  - formatting datetime elements 374
  - formatting number elements 374
  - formatting time elements 374
  - generating charts 376
  - inserting content reference links 375
  - inserting content reference URLs 376
  - inserting footers 376
  - inserting headers 376
  - inserting message catalog text 373
  - understanding 372
- Pagelet Wizard search options
  - specifying 343
- Pagelet Wizard - Specify Publishing Options page 531
- pagelet wizard web library 556
- pages
  - assembling via HTML objects 167
  - assembling via templates 25, 29, 30
  - caching 210, 236
  - combining forms and frames 197
  - configuring error pages 245
  - configuring expiration pages 245
  - configuring start pages 244
  - creating via HTML objects 168
  - diagnosing layout problems 241
  - enabling border/color attributes in table layouts 241
  - inheriting style classes 194
  - inserting style sheets into page HTML 241
  - overriding character set encoding 259
  - PTCS\_FIELDMAPS\_SEC 473
  - PTCS\_HANDLER 477
  - PTCS\_SRVCFG\_SRCH 470, 475
  - selecting for authentication token enable 247

- selecting for passwords 248
  - selecting for setting trace parameters 247
  - selecting for signon/signoff 247
  - selecting for user authentication failure 248
  - selecting the cookies required page 247
  - selecting the SSL required pages 247
  - selecting the user profile page 247
  - template pagelets based on 193
  - viewing/debugging HTML 242
- Parent Folder 74
- passwords
  - default for web profiles 217
  - encrypting for web profiles 219
  - selecting pages for 248
  - setting for web profiles 231
  - setting for XML Link users 231
- PeopleSoft Application Designer
  - See* Application Designer
- PeopleSoft Business Interlinks Business Interlinks
- PeopleSoft Component Interfaces 22
- PeopleSoft component URL types
  - component parameters 462
  - creating 461
  - example 462
- PeopleSoft embeddable pagelet URL types
  - creating 462
  - example 463
  - service URL parameters 463
  - URL information 463
- PeopleSoft Enterprise Portal
  - Related Discussion service 454
  - Related Links service 455
  - Related Tags service 455
- PeopleSoft Integration Broker
  - See* Integration Broker, Integration Broker
- PeopleSoft OBIEE pagelet URL types
  - creating 463
  - example 464
  - service URL parameters 464
  - URL information 464
- PeopleSoft Optimization Framework
  - See* Optimization Framework
- PeopleSoft Performance Monitor
  - Performance Monitor
- PeopleSoft portal
  - configuring 213, 547
  - configuring caching 550
  - configuring cookie rules 550
  - configuring general properties 547
  - configuring look and feel 553
  - configuring security properties 548
  - configuring tracing/debugging 552
  - configuring virtual addressing 549
  - deploying applications 21
  - installing sites 4
  - URL formats 32
- PeopleSoft Process Scheduler 15
- PeopleSoft Pure Internet Architecture
  - application servers
    - See Also* application servers
  - batch server environment 15
  - browsers *See Also* browsers
  - configuring 20
  - database servers *See Also* database servers
  - deploying applications 21
  - improving same-server performance under
    - SSL 264
  - installing 4
  - overriding the port/protocol/server name for
    - URL references 233
  - single signon *See Also* single signon
  - understanding 3, 7
  - unwrapped content URL formats 35
  - URL formats 34
  - using SSL accelerators 265
  - web servers *See Also* web servers
- PeopleSoft Query URL types
  - creating 464
  - example 465
  - service URL parameters 465
  - URL information 465
- PeopleSoft script, related content 465
- PeopleSoft script URL types
  - creating 465
  - example 466
  - iScript parameters 466
  - service URL parameters 466
- PeopleSoft tags 29, 181, 194
- PeopleTools
  - accessing 21
  - base portal *See Also* PeopleSoft portal
  - naming objects 106
  - portal technology *See Also* portal technology
  - web libraries 555
- PeopleToolsBasicSecurity, default handler 476
- performance issues
  - applying load balancing 272
  - applying timeouts to browser windows 237
  - caching, browser-based 207
  - caching, configuring portal 6
  - caching, page 210
  - caching, understanding portal 199
  - caching, web server-based navigation 211
  - caching metadata 205
  - creating content reference links 127
- Performance Monitor
  - See Also* Performance Monitor
- portal homepage pagelets 153
- proxying 37
- server memory consumption, managing 206
- servers, configuring application/database 20
- SSL, improving same-server performance
  - under 264
- SSL accelerators, using 265
- tracing 242
- Performance Monitor
  - enabling agents 223
  - enabling agent tracing 240
  - logging debug information 240
  - setting the monitor buffer maximum 223
- Performance Monitor web library 556
- permission lists
  - menu items, accessing 145
  - menu items, viewing permission lists
    - associated with 149
  - menu security, setting 147
  - reviewing content reference security 123
  - setting folder security 108
- personalization
  - caching personalized pagelets 204
  - selecting personalization pages for pagelets
    - 163
- Placement in Folder 74
- PORTAL\_CSS program 133
- PORTAL\_DEFAULT template 142
- PORTAL\_HP\_2COL\_LAYOUT object 168

- PORTAL\_HP\_3COL\_LAYOUT object 168
- PORTAL\_HP\_ACTIONBAR\_EDIT object 173
- PORTAL\_HP\_ACTIONBAR\_HELP object 173
- PORTAL\_HP\_ACTIONBAR\_REFRESH object 172
- PORTAL\_HP\_ACTIONBAR\_REMOVE object 172
- PORTAL\_HP\_ACTIONBARIMAGE object 168
- PORTAL\_HP\_ACTIONBAR object 168, 172
- PORTAL\_HP\_CODMPONENT object 173
- PORTAL\_HP\_COMPONENT object 168
- PORTAL\_HP\_TABS object 171, 172
- PORTAL\_HP\_USER\_TEMPLATE object 168, 170, 171
- PORTAL\_INDEX program 150
- PORTAL\_MENU process
  - See Also* Menu Import process
- PORTAL\_UNI\_HEADER\_NEW object 168
- portal definitions
  - adding/editing 103
  - adding portals to other databases 102
  - administering 101
  - changing default portals 104
  - enabling folder navigation 104
- Portal Definitions - Add Portal page 103
- portal objects
  - caching 206, 238
  - managing 139
  - Portal Objects folder
    - See Also* Portal Objects folder
  - synchronizing security 132
- Portal Objects folder
  - changing/viewing settings 139
  - understanding 28
- Portal page 255
- PortalRegisteredURL response header 195
- portal registry
  - administering 99
  - building search indexes 149
  - database support 28
  - importing menu groups
    - See Also* Menu Import process
  - maintaining 28
  - navigating 26
  - understanding 27
- portals
  - adding to databases 102
  - administering content references 109
  - administering folders 105
  - allowing unregistered content 227
  - authentication domains
    - See Also* authentication domains
  - caching caching
  - changing default 104
  - configuring general properties 220
  - configuring security 5, 225
  - definitions *See Also* portal definitions
  - homepages homepages
  - improving same-server performance under SSL 264
  - JSR 168 *See Also* JSR 168
  - load balancing 272
  - managing settings 136
  - multiple applications on 215
  - nodes *See Also* nodes
  - objects portal objects
  - overriding character set encoding for pages 259
  - overriding the protocol/server name for URL references 233
  - PeopleSoft portal *See Also* PeopleSoft portal
  - PeopleSoft Pure Internet Architecture
    - PeopleSoft Pure Internet Architecture
  - portal-aware applications 116
  - Portal Objects folder
    - See Also* Portal Objects folder
  - portal servlets portal servlets
  - portal technology portal technology
  - registering components 134
  - registry *See Also* portal registry
  - selecting components to include in navigation 263
  - setting up 5
  - signing on to non-default 258
  - templates *See Also* templates
  - using custom headers 177
  - using HTTPS 227
  - viewing portal attributes for web servers 253
  - web profiles *See Also* web profiles
- Portal Security Synchronization page 132
- portal servlets
  - assembling pages 30
  - converting anchor tags 39
  - handling frame-based templates 30
  - handling iframe-based templates 30
  - handling page-based templates 29, 38
  - proxying URLs *See Also* proxying
  - selecting templates 262
  - setting up 30
  - understanding 17, 29
- portal technology
  - architecture 23
  - implementing 3
  - understanding 3
- portal utilities
  - creating navigation collections 61
  - defining at the registry level 59
  - defining at the system level 56
  - publishing navigation collections 80
  - running 88
- portal utilities options
  - defining at the registry level 59
  - defining at the system level 56
- portal web library 556
- portlet container
  - configuring for JSR 168 540
- portlet entity registry 541
- portlets, JSR 168 *See* JSR 168 portlets
- ports, overriding 233
- process flow
  - related content 453
- Process Scheduler 15
- producer
  - importing information 520
  - registering 522
- Producer Offered Portlets page 524
- producing remote portlets
  - language support for 545
- PROD web profile 217
- protocol
  - understanding WSRP 491
  - using HTTPS 506
- protocols, specifying default 233
- proxying
  - caching proxied JavaScript 207
  - converting anchor tags 39

- converting form tags 39
- converting JavaScripts 40
- converting URLs from relative to absolute 39
- cookies 235
- handling page-based templates 38
- proxy servers *See Also* proxy servers
- reverse proxy servers reverse proxy servers
- specifying hosts not to proxy 269
- understanding 30
- understanding relative URLs 37
- proxy servers
  - issuing requests 268
  - reverse proxy servers
    - See Also* reverse proxy servers
  - specifying hosts not to proxy 269
- PS\_TOKEN
  - securing 228
  - selecting authorization token enable pages 247
  - understanding authentication domains 214
  - understanding single signon 257
- PSADMIN
  - managing application/Process Scheduler servers 16
  - understanding 9
- PSAPPSRV server process 13
- PSCipher 219, 509
- PSMSGDSP server process 14
- PSMSGHND server process 14
- PSOPTENG server process 14
- PSPUBDSP server process 14
- PSPUBHND server process 14
- PSQCKSRV server process 13
- PSQRYSRV server process 14
- PSSAMSRV server process 14
- PSSTYLEDEF style sheet 178
- PSSUBDSP server process 14
- PSSUBHND server process 14
- PTCS\_SECURITY application class, default handler 476
- PTPP\_CPPROJ 89
- PTPP\_FD\_DEL 89
- PTPP\_SCSYNC 89
- Publish Collection page 80

## Q

- queries
  - adding query strings for template pagelets 193
  - reviewing content reference security 123
  - URL format 35
- queries web library 557
- query URL format 35
- queues
  - request 12
  - understanding 10

## R

- redirecting pages 259
- refreshing
  - automatic cache refresh 203
  - disabling for pagelets 163
  - disabling manual pagelet 205
  - forcing for pagelets 163

- metadata caches 206
- updating pages 179
- registration
  - registering components in multiple portals 134
  - registering components multiple times 134
  - registering content references 31
  - registering related link URLs 121
  - Registration Wizard
    - See Also* Registration Wizard
- Registration page 522
- registration properties 523
- Registration Wizard
  - adding internal PeopleSoft content references 113
  - registering components 134, 261
  - registering menus 261
- Registry Name 60
- Registry Object Prefix 58
- registry options, defining for navigation collections 59
- Registry Options page 53
- related content 451
  - assigning services to application pages 469, 470
  - business process flow 453
  - cloning 469
  - common elements 455
  - configuring OBIEE 573
  - copying service definitions 469
  - defining services 459
  - elements used to assign/manage 475
  - example 452
  - iScripts 465
  - managing 469, 470
  - managing configuration 475
  - mapping service parameters 473
  - non-PeopleSoft URL 460
  - PeopleSoft component URL 461
  - PeopleSoft embeddable pagelet URL 462
  - PeopleSoft OBIEE URL 463
  - PeopleSoft Query URL 464
  - PeopleSoft script URL 465
  - Security Handlers page 477
  - service types 453
  - steps to test service parameters 467
  - URL testing 466
  - writing help information 467
- Related Content Framework
  - diagram of business process flow 453
  - supported service types 459
- Related Content - Security Handlers page 477
- Related Content Service Definition page
  - help information 467
- Related Content Service - Map Fields page 473
- related content services 451
  - delivered in PeopleTools 454
  - PeopleSoft Enterprise Portal 454
  - Related Discussion service 454
  - Related Links service 455
  - Related Tags service 455
  - understanding 451
- Related Discussion service 454
- related links
  - creating 119
  - creating groups 120
  - defining the RELINK attribute 120
  - testing 121
  - using the RELINK attribute 117

- Related Links service 455
- Related Tags service 455
- remote portlets 525, 530
- remote portlets, setting up 513
- reporting web library 557
- reports
  - report repository servlet 17
  - selecting properties 224
- request queues 12
- reverse proxy servers
  - overriding the port/protocol/server name for
    - URL references 233
  - proxying and relative URLs 37
  - proxy servers *See Also* proxy servers
  - retrieving portal content 233
  - selecting 267
  - setting up the hosts file 270
  - understanding 266
  - using firewalls 269
  - using SSL accelerators 271
- rich text editor image upload web library 557
- roles
  - accessing menu items 145
  - reviewing content reference security 124
  - server-based caching 200
  - setting folder security 108
  - setting menu security 147
- root folder 28
- RPS *See* reverse proxy servers
- rules, configuring cookie 234

## S

- SAML 493
- scripts
  - encrypting passwords (PSCipher) 219
  - iScripts *See Also* iScripts
  - JavaScripts JavaScripts
  - security settings for content references 113
  - timeout warning 229, 557
  - URL format 34, 36
- script URL format 34, 36
- SDK web library 557
- search
  - building registry search indexes 149
  - using 26
- Secure Sockets Layer (SSL) *See* SSL
- securing data 259
- security
  - accessing portal administration pages 100
  - accessing web libraries 555
  - configuring application 5
  - configuring for PeopleSoft portal 548
  - configuring for portals 5
  - configuring portal 225
  - configuring web servers 21
  - modifying for web libraries 427
  - new session upon signon 227
  - reviewing for content references 123
  - setting for content reference links 128
  - setting for content references 122
  - setting for folders 108
  - SSL *See Also* SSL
  - synchronizing for portal objects 132
  - viewing menu details 147
  - viewing permission lists, roles and users with

- menu item access 145
  - web profiles 217
- security handler default
  - PeopleToolsBasicSecurity 476
  - PTCS\_SECURITY application class 476
- security handlers
  - defaults 476
  - defining 477
  - understanding 476
- security requirements for WSRP
  - determining 497
- Security standard navigation page 44
- Select a Content Reference or Content Reference
  - Link page 471
- Select Source Folder page 71
- Select Source Link page 76
- sensitive data pages 259
- server processes
  - application server domains
    - See Also* application server domains
  - listeners, handlers, and queues 10
  - list of 13
  - understanding 13
- servers
  - application *See Also* application servers
  - batch 15
  - database *See Also* database servers
  - Domain Name Server (DNS) 215
  - improving same-server performance under
    - SSL 264
  - monitoring 14
  - overriding the server name for URL
    - references 233
  - passing cookies 235
  - Process Scheduler 16
  - proxy servers *See Also* proxy servers
  - reverse proxy reverse proxy servers
  - server-based caching 200
  - web *See Also* web servers
- service
  - security options 475
- service definitions, steps to copy 469
- services
  - assigning to application pages 475
  - Homepage Generation Service 175
  - server processes *See Also* server processes
  - single signon authentication 257
  - understanding 14
  - web 17
- service security options, fields 458
- service types
  - related content 453
  - supported by Related Content Framework 459
- service URL, parameter fields 457
- servlets
  - integration gateway 17
  - Java *See Also* Java servlets
  - portal portal servlets
  - report repository 17
  - servlet engine 17
  - setting timeouts 231
  - understanding Jolt and Tuxedo 17
- session cookies, allowing *See* cookies
- session variables 246, 248
- Show Breadcrumbs 54
- Show Images
  - defining at registry level 61

- defining at system level 59
- Show Main Menu Breadcrumb 54
- signing out 27
- signon
  - checking for duplicate cookies 251
  - displaying the trace setting link 241
  - selecting error pages for 248
  - selecting pages for 247
  - signing on to non-default portals 258
  - single *See Also* single signon
- single signon
  - accessing frame-based content 258
  - authentication domains
    - See Also* authentication domains
  - securing PS\_TOKEN 228
  - understanding 257
- Small Collection Image
  - defining at registry level 61
  - defining at system level 59
- Small Content Image
  - defining at registry level 61
  - defining at system level 59
- Small Folder Image
  - defining at registry level 61
  - defining at system level 59
- SOAP 493
- soap, web library 557
- Specify Data Source Parameters page 330
- SQL
  - application server domains
    - See Also* application server domains
  - server processes 13
- SSL
  - accelerators *See Also* SSL accelerators
  - improving same-server performance 264
  - overriding the port/protocol for URL
    - references 233
  - selecting the SSL required pages 247
  - setting web profile options 227
  - specifying HTTPS for portals 227
- SSL accelerators
  - applying load balancing 272
  - using 265
  - using reverse proxy servers 271
- standard navigation pages
  - describing folder and link sequence on 45
  - enabling 45
  - understanding 44
- states
  - browser 210, 237
  - menu 209
  - window 540
- static templates
  - frame-based example 183
  - page-based example 183
  - understanding 182
- Status 523
- Structure & Content for Portal Objects page 139
- Style Sheet Name 55
- style sheets
  - assembling pages 31
  - changing link colors 178
  - configuring cache directories 239
  - inheriting classes 194
  - inserting style sheets into page HTML 241
- Synchronize Collection Security page 90
- Synchronize Collection Security process
  - running 90

- understanding 89
- system options, defining for Navigation Collections 56
- System Options page 53, 56

## T

- Tab Content page 157
- Tab Layout page 158
- tables, HTML 182
- tabs
  - accessing online help 156
  - adding 154
  - adding images 175
  - changing order 160
  - creating 173
  - hiding action bars *See Also* action bars
  - HTML objects for 171
  - labelling 155, 156
  - naming 155, 156
  - overriding images/objects 173
  - selecting content 157
  - setting accessibility by date 156
  - specifying layout 158
- tags
  - anchor 39
  - form 39
  - PeopleSoft 29, 181, 194
  - template pagelet HTML 194
- Target tag 31
- template pagelets
  - based on pages 193
  - understanding 25
  - understanding HTML tags for 194
- templates
  - adding 141
  - applying to unregistered URLs 196
  - configuring node-specific information 138
  - conflicting style classes 194
  - deleting 142
  - developing 193
  - dynamic *See Also* dynamic templates
  - editing 141
  - frame-based *See Also* frame-based templates
  - iframe-based iframe-based templates
  - managing 140
  - page-based *See Also* page-based templates
  - pagelets template pagelets
  - PORTAL\_DEFAULT template 142
  - selecting for portal servlets 262
  - setting pagelet names in 196
  - specifying pagelet caching 202
  - static *See Also* static templates
  - storing (HTML catalog) 140
  - types of 181
  - understanding 25, 181
  - using non-PeopleSoft content providers 195
  - using predefined template HTML objects 167
- template types
  - applying 181
- testing
  - content references 124
  - generating HTML for 241
  - related content service 466
  - related content service parameters 467
  - related links 121

- using the TEST web profile 216
- Test Related Content Service page 466, 467
- TEST web profile 216
- timeouts
  - applying to browser windows 237
  - configuring expiration pages 245
  - configuring for pagelets 164
  - setting for authenticated users 228
  - setting for Jolt 231
  - setting for pagelet loading 117
  - setting for public users 231
  - setting for servlets 231
- timeouts web library 557
- tracing
  - configuring 240, 552
  - selecting pages for setting trace parameters 247
- transactions
  - managing *See Also* Tuxedo
  - understanding server processes 13
  - web servers *See Also* web servers
- Tuxedo
  - transmitting requests and data 17
  - understanding 15
  - workstation listeners/handlers 12
- type ahead
  - disabling 251

## U

- uniform resource locators (URLs) *See* URLs
- universal navigation header
  - constructing 167, 168, 171
  - understanding the buttons 26
- UNIX
  - monitoring servers 14
  - using PSCipher 219
- URL
  - selecting a data source 323
  - specifying data source parameters 337
- URL Information 528
- URL information, common fields 456
- URLs
  - applying templates for partial path nodes 196
  - changing default portals 105
  - constructing references for frame-based templates 195
  - content reference links 126
  - converting from relative to absolute 38
  - defining content references 115
  - generating relative 232
  - invoking iScripts 555
  - nodes *See Also* nodes
  - overriding the port/protocol/server name for URL references 233
  - PeopleSoft portal URL formats 32
  - PeopleSoft Pure Internet Architecture content types 34
  - proxying *See Also* proxying
  - retrieving non-PeopleSoft content 195
  - understanding pagelet 36
  - understanding relative 37
  - understanding system 36
  - understanding the portal registry 27
  - unwrapped content formats 35
  - using authentication domains 215

- URL type
  - creating PeopleSoft OBIEE pagelet 463
  - creating PeopleSoft Query 464
  - creating PeopleSoft script 465
  - PeopleSoft embeddable pagelet 462
- URL types
  - creating non-PeopleSoft 460
  - creating PeopleSoft component 461
  - understanding 459
- User Profiles standard navigation page 44
- users
  - accessing menu items 145
  - reviewing content reference security 123
  - selecting the user profile page 247
  - server-based caching 200, 203, 204
  - setting access options for web profiles 230
- utilities
  - PSADMIN 9
  - PSCipher 219

## V

- variables
  - bind *See Also* bind variables
  - global 214
  - numeric message 246
  - session 248
  - session message 246
  - sharing between components 214
  - system 235
- virtual addressing
  - configuring 232, 549
  - running portals under SSL 264
  - setting default 233
  - using firewalls with reverse proxy servers 270
  - using reverse proxy servers with SSL accelerators 271
  - using SSL accelerators 266

## W

- web libraries 555
- WebLogic
  - applying load balancing 272
  - preventing cookie overwrites 234
  - running portals under SSL 264
  - specifying hosts not to proxy 269
  - using proxy servers to issue requests 268
  - using SSL accelerators 265
- Web Profile Configuration component
  - Caching page 235
  - Cookie Rules page 234
  - Custom Properties page 249
  - Debugging page 240
  - General page 220
  - Look and Feel page 242
  - Security page 225
  - Virtual Addressing page 232
- Web Profile History page 253
- web profiles
  - changing 218
  - configuring 4
  - configuring browser caching 236

- configuring cache directories 239
  - configuring caching 550
  - configuring cookie rules 234, 550
  - configuring custom properties 249
  - configuring error pages 245
  - configuring expiration pages 245
  - configuring general portal properties 220
  - configuring general properties 547
  - configuring look and feel 242, 553
  - configuring password pages 248
  - configuring portal security 225
  - configuring security 548
  - configuring security/tracing pages 247
  - configuring signon/signoff pages 247
  - configuring tracing/debugging 552
  - configuring tracing/debugging options 240
  - configuring virtual addressing 232, 549
  - configuring web server caching 238
  - encrypting user IDs/passwords 219
  - improving same-server performance under SSL 264
  - mapping language codes 249
  - new session upon signon 227
  - selecting reverse proxy servers 233, 267
  - setting default addressing 233
  - setting SSL options 227
  - setting timeout options 228
  - setting user access options 230
  - setting web server Jolt options 231
  - specifying initial 217
  - understanding 216
  - unlocking accounts 217
  - using firewalls with reverse proxy servers 269
  - using reverse proxy servers with SSL accelerators 271
  - using SSL accelerators 265
  - viewing histories 253
  - web servers
    - applying load balancing 272
    - browsers *See Also* browsers
    - caching, configuring 238
    - caching pagelet content 202
    - caching target content 201
    - configuring 20
    - Jolt *See Also* Jolt
    - managing memory consumption 206
    - maximum name length 33
    - PeopleSoft servlets 17
    - portal servlets *See Also* portal servlets
    - setting up the hosts file 270
    - software elements 17
    - Tuxedo *See Also* Tuxedo
    - understanding 16
    - viewing portal attributes 253
    - WebLogic *See Also* WebLogic
    - WebSphere WebSphere
  - Web Service Endpoint URL 521
  - web services 17
  - web services, understanding 491
  - Web Services Description Language (WSDL) web library 555
  - WebSphere
    - preventing cookie overwrites 234
    - specifying hosts not to proxy 269
    - using proxy servers to issue requests 268
  - Windows
    - accessing online help 222
    - monitoring servers 14
    - using the PeopleTools development environment 8, 21
  - window states, JSR 168 540
  - worklists, accessing 26
  - worklist URL format 35
  - workstation handlers (WSH) 12
  - workstation listeners (WSL) 12
  - WSDL 494
  - WSDL web library 555
  - WSH 12
  - WSL 12
  - WSRP 494
    - configuring WS-Security 512
    - configuring WS-Security for 497
    - defining interfaces and operations 492
    - defining terminologies 493
    - implementing protocol scenario 494
    - language support for 545
    - offering portlets 533
    - setting up security options 506
    - specifying pagelets for production 531
    - understanding 491
    - understanding the Consumption Wizard Data 519
    - using the consumption wizard 518
    - viewing content references 535
  - WSRP4J 494
  - WSRP Consumption Wizard
    - understanding the data 519
    - using 518
  - WSS 494
  - WS-Security
    - configuring 512
    - configuring for WSRP consumers 514
    - understanding 498
  - WS-Security SOAP Message Structure 498
- ## X
- XML 494
  - XML Link 231
  - xmllinks web library 557