
Oracle's PeopleTools PeopleBook

PeopleTools 8.52: PeopleTools Portal Technologies

October 2011

PeopleTools 8.52: PeopleTools Portal Technologies
SKU pt8.52tptr-b1011

Copyright © 1988, 2011, 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

PeopleTools Portal Technologies Preface	xxiii
PeopleTools Portal Technologies	xxiii
PeopleBooks and the PeopleSoft Online Library	xxiii

Part 1

Getting Started with PeopleTools Portal Technologies

Chapter 1

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

Chapter 2

Understanding PeopleSoft Pure Internet Architecture	7
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	14
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	16
PeopleSoft Servlets	17
Oracle Jolt	17
Web Browser	18
Configuration and Deployment Options	19
Configuration Options	20
Deployment Options	21

Chapter 3

Understanding Portal Technology	23
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

Working with PeopleTools Portals

Chapter 4

Working with Navigation Pages	43
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	53
Understanding Navigation Collection Options	53
Common Elements Used in This Section	55
Pages Used to Set Up Navigation Collection 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
Finding Existing Navigation Collections	64
Creating or Editing a Navigation Collection	65
Adding a Folder	68
Selecting a Source Folder	71
Selecting Navigation Collection Images	73
Editing a Folder	74
Adding a Link	75
Selecting a Source Link	77
Editing a Link	79
Publishing Navigation Collections	81
Pages Used to Publish Navigation Collections	81
Publishing a Navigation Collection	81
Publishing Pagelets	84
Pages Used to Publish Pagelets to Portals	85
Publishing Pagelets to Multiple Portals	85
Publishing Multiple Pagelets to a Portal	87
Running Portal Utility Processes	88
Understanding Portal Utility Processes	89

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

Chapter 5

Administering Portals	97
Understanding Portal Administration	97
Understanding Folder Navigation	100
Common Elements Used in This Chapter	102
Administering Portal Definitions	104
Common Elements Used in This Section	104
Pages Used to Administer Portal Definitions	104
Defining Portals	104
Adding and Editing Portal Definitions	106
Changing Default Portals	107
Administering Folders	107
Understanding Folder Administration	108
Pages Used to Administer Folders	111
Defining Folders	111
Setting Folder Permissions	114
Administering SmartNavigation Folders	107
Understanding SmartNavigation Folders	108
Understanding Data Sources: Rowsets	118
Understanding Data Sources: Trees	123
Understanding Data Sources: Dynamic Hierarchy	132
Common Elements Used In This Section	136
Defining SmartNavigation Folders	138
Defining Dynamic Hierarchy Folders	140
Defining Rowset-Based SmartNavigation Folders	141
Defining Tree-Based SmartNavigation Folders	142
Administering Content References	143
Pages Used to Administer Content References	143
Defining Content References	145
Creating Related Links	153
Setting Content Reference Permissions	157
Reviewing Content Reference Permissions	158
Testing Content References	159
Defining Content Reference Links	160
Setting Content Reference Link Permissions	164
Managing Folders and Content References	164
Pages Used to Manage Folders and Content References	165

Copying Folders and Content References	166
Moving Folders and Content References	166
Deleting Folders and Content References	168
Synchronizing Portal Object Security	168
Registering Components in Multiple Portals	170
Registering Components Multiple Times Within a Portal	170
Managing General Portal Settings	171
Page Used to Manage General Portal Settings	171
Managing General Settings for Portals	172
Managing Portal Objects	176
Pages Used to Manage Portal Objects	176
Viewing Portal Objects Folder Settings	177
Understanding Templates	177
Adding and Editing Templates	179
Deleting Templates	179
Reviewing Menu Item Information	179
Common Elements Used in This Section	180
Pages Used to Review Menu Item Information	180
Viewing Menu Item Details	180
Viewing Menu Folder Structure Information	182
Viewing Menu Security Details	184
Building Registry Search Indexes	186
Pages Used to Build Registry Search Indexes	186
Associating Keywords with Content References	187
Building Search Indexes for PeopleSoft Applications	187

Chapter 6

Modifying the Portal Interface	189
Using Predefined Template HTML Objects	189
Using Predefined Homepage HTML Objects	191
Constructing Homepages	191
Constructing Headers	193
Constructing Pagelets	195
Adapting Homepage and Pagelet Objects	195
Using Custom Portal Headers	199
Changing Link Colors	200
Applying Changes	201

Chapter 7

Working with Portal Templates	203
--	------------

Understanding Portal Templates	203
Applying Template Types	203
Understanding Template Types	204
Using Page-Based Static Templates	205
Using Frame-Based Static Templates	205
Using Frame-Based Dynamic Templates	206
Using iFrame-Based Dynamic Templates	210
Developing Portal Templates	216
Template Pagelets Based on Pages	216
Inheritance of Style Classes	217
Template Pagelet Tags	217
Considerations for Non-PeopleSoft Content Providers	218
Considerations for Frame-Based Templates	218
Partial Path Nodes	219
Pagelet Names in Templates	219
Considerations for Forms and Templates	219
Error Message Display	220

Chapter 8

Using Portal Caching Features	221
Understanding Portal Caching	221
Administering Server-Based Caching	222
Understanding Caching Scope	222
Implementing Target Content Caching	223
Implementing Pagelet Caching	224
Implementing Metadata Caching	227
Implementing Proxied JavaScript Caching	229
Administering Browser-Based Caching	229
Understanding Browser-Based Caching	229
Implementing Homepage Caching	230
Controlling Navigation Pagelet Caching	231
Implementing PeopleSoft Page Caching	232
Administering Web Server-Based Navigation Caching	233
Understanding Web Server-Based Navigation Caching	233
Implementing Web Server-Based Navigation Caching	233

Chapter 9

Configuring the Portal Environment	235
Understanding the Authentication Domain	236
Configuring Web Profiles	238

Understanding Web Profiles	238
Pages Used to Configure Web Profiles	241
Configuring General Portal Properties	242
Configuring Portal Security	247
Configuring Virtual Addressing	254
Configuring Cookie Rules	257
Configuring Caching	258
Configuring Trace and Debug Options	264
Configuring Appearance and Character	267
Configuring Custom Properties	274
Viewing Web Profile History	278
Defining Portal Nodes	279
Understanding Portal Nodes	279
Pages Used to Define Portal Nodes	279
Setting Portal Nodes	279
Specifying the Node Type	281
Implementing Single Signon Functionality	281
Understanding Single Signon Functionality	281
Signing In to a Non-Default Portal	282
Redirecting Pages with Sensitive Data	283
Overriding Page Encoding	283
Importing Menu Groups into the Portal Registry	283
Understanding Menu Group Importing	284
Page Used to Import Menu Groups	285
Importing Menu Groups	285
Selecting Components to Include in Portal Navigation	287
Improving Same-Server Performance Under SSL	288
Using SSL Accelerators	289
Using Reverse Proxy Servers	290
Understanding Reverse Proxy Servers	290
Making Reverse Proxy Settings	291
Issuing Requests by Using a Proxy Server	292
Specifying Hosts That Should Not Be Proxied	293
Using Firewalls with Reverse Proxy Servers	293
Using Reverse Proxy Servers with SSL Accelerators	295
Applying Load Balancing	296

Part 3

Working with Homepages, Dashboard Pages, and Pagelets

Chapter 10

Administering Homepages, Dashboard Pages and Pagelets	301
Understanding Homepages and Dashboard Pages	301
Managing Homepages	302
Pages Used to Manage Homepages	303
Registering Homepages in the Portal Registry	303
Setting Homepage Permissions	306
Selecting the Pagelets That Appear on Homepages	308
Arranging Pagelets on Homepages	309
Specifying Homepage Tab Sequence	311
Troubleshooting Homepages	311
Managing Dashboard Pages	312
Pages Used to Manage Dashboard Pages	313
Creating and Managing Dashboard Pages	313
Specifying Dashboard Page Properties	314
Specifying Dashboard Page Security Properties	316
Specifying the Location of Dashboard Page Link	317
Specifying the Pagelets That Appear on Dashboard Pages	318
Arranging the Pagelets on Dashboard Pages	320

Chapter 11

Using Pagelet Wizard to Create and Manage Pagelets	323
Understanding Pagelet Wizard	323
Understanding Pagelet Wizard Data Sources	325
Understanding Pagelet Wizard Display Formats	326
Understanding Pagelet Wizard Display Sizes	332
Understanding Pagelet Wizard Pagelet Types	332
Defining Pagelet Wizard Setup Values	335
Pages Used to Set Up Pagelet Wizard Setup Values	335
Setting Up the PeopleSoft Integration Broker Gateway for Pagelet Wizard	336
Setting Up the Default Gateway for Use with Pagelet Wizard Data Sources	336
Setting Up the Pagelet Wizard IB Connector Data Source Security	336
Setting Up the Pagelet Wizard Integration Broker Data Source	337

Defining Pagelet Wizard Transformation Types, Display Formats, and Data Types	338
Pages Used to Define Pagelet Wizard Transformation Types, Display Formats, and Data Types	338
Defining Transformation Types	338
Defining Display Formats	340
Defining Data Types	342
Defining Pagelet Wizard XSL Prototypes	346
Page Used to Define Pagelet Wizard XSL Prototypes	346
Defining XSL Prototypes	346
Defining Pagelet Wizard Headers and Footers	348
Pages Used to Define Pagelet Wizard Headers and Footers	349
Defining Headers	349
Defining Header Images	351
Defining Footers	352
Defining Footer Images	353
Defining Pagelet Wizard Pagelet Categories	354
Page Used to Define Pagelet Wizard Pagelet Categories	354
Defining Pagelet Categories	355
Step 1: Entering Pagelet Identifying Information	356
Page Used to Enter Pagelet Identifying Information	356
Specifying Pagelet Information	356
Step 2: Selecting a Pagelet Data Source	358
Page Used to Select a Pagelet Data Source	359
Selecting an HTML Data Source	359
Selecting an IB Connector Data Source	361
Selecting an Integration Broker Data Source	362
Selecting a Navigation Collection Data Source	364
Selecting an OBIEE Report Data Source	365
Selecting a PeopleSoft Query Data Source	366
Selecting a Pivot Grid Data Source	367
Selecting a Rowset Data Source	368
Selecting a Search Record Data Source	369
Selecting a URL Data Source	371
Step 3: Specifying Pagelet Data Source Parameters	372
Understanding System Variables Supported as Data Source Parameters	373
Common Elements Used in This Section	374
Pages Used to Specify Pagelet Data Source Parameters	377
Specifying HTML Data Source Parameters	377
Specifying IB Connector Data Source Parameters	378
Specifying Integration Broker Data Source Parameters	379
Specifying Navigation Collection Data Source Parameters	380
Specifying OBIEE Data Source Parameters	381
Specifying Pivot Grid Data Source Parameters	382
Specifying Query Data Source Parameters	386
Specifying Rowset Data Source Parameters	388
Specifying Search Record Data Source Parameters	388

Specifying URL Data Source Parameters	391
Specifying User-Defined Data Source Paramters	392
Step 4: Selecting a Pagelet Display Format	393
Page Used to Select a Pagelet Display Format	393
Selecting a Display Format	394
Step 5: Specifying Pagelet Display Options	395
Common Elements Used in This Section	395
Pages Used to Specify Pagelet Display Options	398
Specifying Table Display Options	400
Specifying Passthru Display Options	403
Specifying Search List Display Options	405
Specifying Menu Display Options	406
Specifying List Display Options	409
Specifying Chart Display Options	410
Specifying Custom Display Options	414
Specifying Data Link Details	416
Step 6: Specifying Pagelet Publishing Options	419
Understanding Caching Options	419
Common Elements Used in This Section	420
Pages Used to Specify Pagelet Publishing Options	421
Specifying Pagelet Publication Options	422
Using Pagelet Wizard Post-Transformation Processing	427
Understanding Pagelet Wizard Post-Transformation Processing	427
Inserting Message Catalog Text	428
Formatting Date, Time, DateTime, Currency, and Number Elements	429
Disabling WSRP Style Substitution	430
Inserting Content Reference-Based Links	430
Inserting a Content Reference URL as Text	431
Generating Charts	431
Inserting Pagelet Headers and Footers in Specific Locations	431
Administering Pagelet Wizard Pagelets	432
Common Elements Used in This Section	432
Pages Used to Administer Pagelet Wizard Pagelets	434
Deleting and Copying Pagelet Wizard Pagelets	434
Generating Export and Import Scripts for Pagelet Wizard Pagelets	437
Reviewing Pagelet Wizard Pagelet Details	440

Chapter 12

Developing and Registering Pagelets	443
Understanding Pagelets	443
Prerequisites	443
Pagelets	444

Pagelet Development	444
Pagelet Extensions	446
Pagelet Personalizations	448
Understanding PeopleSoft Portal Architecture	449
PeopleSoft Portal Architecture	450
Client Access	451
The Web Server	451
The Application Server	452
The Database Server	453
The Portal Registry	453
Portal Components	454
Page Assembly	455
Understanding Pagelet Development	457
Pagelet Development Guidelines	457
Pagelet Development Considerations	458
Pagelet Development Tools	467
Pagelet Sizes	469
HTML-Related Design Considerations	471
Branding Considerations	473
Understanding Single Signon and Pagelets	474
Single Signon	475
PeopleSoft Authentication Process	475
The PeopleSoft Authentication API	477
Developing PeopleSoft Component-Based Pagelets	483
Developing iScript-Based Pagelets	484
Developing Java-Based Pagelets	486
Understanding Java Development	486
Describing Java Pagelet Development Considerations	487
Writing Business Logic in Java	490
Using Java to Write Business Logic for a PeopleSoft or Non-PeopleSoft System	494
Invoking PeopleSoft Components from Java	495
Developing Contextual Embeddable Pagelets	498
PeopleCode Sample for Rendering the Context-Based Embeddable Pagelet	499
Administering Pagelets in the Portal Registry	500
Registering Homepage and Template Pagelets	500
Modifying Pagelet Attributes and Security	501
Registering URL-Based Pagelets	503
Registering Pagelet Extensions	505
Using Attributes to Enhance Pagelets	506
Understanding Pagelet and Content Reference Attributes	506
Page Used to Manage Pagelet Attributes	507
Managing Pagelet Attributes	507
Configuring Pagelet Help Links	510
Configuring Pagelet Time-out Settings	510
Configuring Automatic Pagelet Refresh	511

Creating Banner Pagelets	511
Adding Images to Pagelet Headers	512
Handling Special Situations	513
Determining Pagelet Running Location	513
Using Refresh Tags in Pagelets	513

Chapter 13

Configuring OBIEE Report Pagelets	515
Understanding OBIEE Integration Within the PeopleSoft Framework	515
Configuring OBIEE Server Setup	516
Configure the Web Server Files	518
Loading OBIEE Report Definitions	518
Loading OBIEE Reports into PeopleSoft Applications	520
Deleting OBIEE Reports from PeopleSoft Applications	521
Viewing OBIEE Report Details	521
Creating OBIEE Report Pagelets	524
Using Pagelet Wizard to Create OBIEE Report Pagelets	524

Part 4

Working with Related Information

Chapter 14

Configuring Interwindow Communication	527
Understanding Interwindow Communication	527
Prerequisites	529
Common Elements Used in This Chapter	529
Page and Pagelet Types	529
Events and Messages	530
Configuring Interwindow Communication Publish and Subscribe Options	531
Pages Used to Configure Interwindow Communication Publish and Subscribe Options	531
Configuring Fields to Publish	531
Configuring Fields to Subscribe	532

Chapter 15

Developing and Configuring Related Content Services	533
--	------------

Understanding the PeopleSoft Related Content Framework	533
Prerequisites	533
Common Elements Used in This Chapter	534
Framework Flexibility	537
The Related Content Development Process	543
Delivered Related Content Services for PeopleSoft Application Portal	543
Delivered Related Content Services for Oracle WebCenter	544
Defining Related Content Services	546
Understanding Service Types	547
Understanding Service URL Parameters	549
Understanding Service Security	550
Common Elements Used in This Section	552
Pages Used to Define Related Content Services	556
Creating and Maintaining Application Class Related Content Service Definitions	557
Creating and Maintaining Non-PeopleSoft Related Content Service Definitions	560
Creating and Maintaining PeopleSoft Component Related Content Service Definitions	561
Creating and Maintaining Embeddable Pagelet Related Content Service Definitions	562
Creating and Maintaining OBIEE Pagelet Related Content Service Definitions	563
Creating and Maintaining PeopleSoft Query Related Content Service Definitions	564
Creating and Maintaining PeopleSoft Script Related Content Service Definitions	565
Testing Related Content Service Definitions	566
Writing Help Text for Related Content Service Definitions	566
Copying Related Content Service Definitions	568
Assigning and Managing Related Content Services	569
Understanding Related Content Assignment	569
Understanding Related Actions Assignment	570
Understanding Service Configuration	574
Prerequisites	576
Common Elements Used in This Section	576
Pages Used to Assign and Manage Related Content Services	577
Selecting a Content Reference	578
Assigning Component-Level Services to Content References	580
Assigning Page-Level Services to Content References	584
Configuring Service Parameters	588
Managing Related Content Configuration	589
Configuring Menu and Frame Layout	589
Understanding Menu Configuration	589
Understanding the Configuration of Related Content Frames	590
Pages Used to Configure Related Content Service Layout	590
Configuring the Related Actions Menu	591
Configuring the Related Content Menu	593
Configuring Page Field Contextual Menus	594
Configuring the Related Content Frames	596

Chapter 16

Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services	599
Understanding the Oracle WebCenter Integration	599
Understanding WebCenter Spaces	600
Understanding WebCenter Services	600
Understanding Security Synchronization	603
Related Language Considerations	604
Prerequisites	604
Configuring Oracle WebCenter Options	606
Page Used to Configure Oracle WebCenter Options	606
Setting the WebCenter Integration Options	606
Synchronizing Oracle WebCenter and PeopleSoft Application Security	608
Page Used to Synchronize Oracle WebCenter and PeopleSoft Application Security	608
Running the WebCenter Security Sync Batch Process	608
Managing WebCenter Links	609
Pages Used to Manage WebCenter Links	609
Managing Instance-Specific Links	609
Managing Shared Links	610
Adding Group Space Links	611
Adding External URL Links	612

Chapter 17

Creating and Configuring WorkCenter Pages	613
Understanding WorkCenter Pages	613
The Pagelet Area	614
The Target Content Area	616
Page Layout Templates	616
Page Design and Processing Considerations for WorkCenter Pages	618
Understanding iScript and Component-Based Pagelets	620
Navigation Collection-Based Pagelets	620
PeopleSoft Query-Based Pagelets	622
Component-Based Pagelets	624
WorkCenter Page Development and Implementation Process Overview	629
Prerequisites	630
Defining Page Layout Templates	631
Page Used to Define Page Layout Templates	631
Configuring WorkCenter Page Layout	631
Managing WorkCenter Page Configurations	633
Pages Used to Manage WorkCenter Page Configurations	633

Administering WorkCenter Pages	634
Selecting the WorkCenter Starting Page	635
Configuring WorkCenter Pages	636

Part 5

Working with Portlet Technologies

Chapter 18

Understanding Web Services for Remote Portlets (WSRP)	643
Understanding WSRP	643
Terminology	645
Implementing WSRP Protocol Scenario	646
Considerations for WSRP and Server Cluster Configuration	648

Chapter 19

Configuring WS-Security for WSRP Consumption and Production	649
Determining Security Requirements	649
Understanding the PeopleSoft WS-Security for WSRP	650
Setting Up WSRP Security Options	657
Using HTTPS Protocol to Communicate Between Producer and Consumer	657
Installing Certificates for Encryption and Signature	658
Setting Up SAML	661
Creating the SAML Administrator	661
Importing Digital Certificates	661
Configuring the SAML Inbound Setup	662
Running the RedeployWSRP.cmd Executable	663
Configuring WS-Security for PeopleSoft as a WSRP Producer	663
Defining Nodes with WS-Security	664
Pages Used to Define Nodes With WS-Security	665
Configuring WS-Security for WSRP Consumers	665

Chapter 20

Using WSRP to Consume and Produce Remote Portlets	669
Setting Up the Local Gateway URL (Consumer)	669
Page Used to Set Up the Local Gateway URL	669

Setting Up the Local Gateway URL	670
Using the WSRP Consumption Wizard (Consumer)	670
Understanding the WSRP Consumption Wizard Data	671
Common Elements Used in This Section	671
Pages Used to Navigate WSRP Consumption Wizard	672
Importing Producer Information	672
Registering Producers	674
Creating Content References	676
Displaying Remote Portlets on a Homepage (Consumer)	677
Pages Used to Display Remote Portlets	678
Reviewing Content References	678
Displaying Remote Portlets on Homepages	680
Producing Remote Portlets (Producer)	681
Pages Used to Produce Remote Portlets	681
Specifying PeopleSoft Content for WSRP Production	682
Offering WSRP Portlets	684
Working with WSRP Content References	686
Setting WSRP Display Mode	686
Mapping PeopleSoft Styles to WSRP Styles	687
Validating WSRP Compliance	688
Using PeopleSoft WSRP-Produced Content	689

Chapter 21

Working with JSR 168 Portlets	691
Understanding JSR 168	691
Understanding PeopleSoft Compliance with JSR 168	691
Java Portlet API	692
Window States	692
Content Caching	692
Portlet Modes	692
Configuring the Portlet Container	692
Describing the Portlet Entity Registry	693
Choosing Portlets to Produce for WSRP	693

Appendix A

Understanding Changes in Portal Configuration Settings	695
General Properties	695
Security Properties	696
Virtual Addressing Properties	697
Cookie Rules	698

Caching Settings	698
Trace and Debug Properties	700
Look and Feel Settings	701

Appendix B

Configuring the Application for 8.4x Navigation	703
Reconfiguring the Default Template	703
Reconfiguring the Navigation Pagelet	706

Appendix C

Converting Frames-Based Templates to iFrame-Based Templates	709
Converting the eProcurement Shopping Cart Template	709
Gathering Information About the Current Template	709
Creating the iFrame Template iScript	710
Creating the iFrame Template HTML Definition	710
Setting Security for the Template iScript	714
Converting the Template Content Reference	715
Troubleshooting Errors	715

Appendix D

Replacing SmartNavigation Images	717
Understanding SmartNavigation Icons	717
Creating Replacement Images	718
Defining New Style Classes	718
Modifying Folder Settings	719
Writing PeopleCode	719
Rowset Data Source	719
Tree Data Source	720

Appendix E

Related Content Reserved Words and Service Parameters	723
Reserved Words Default Mapping	723

Appendix F

Configuring UPK Topics as Related Content	725
Understanding Oracle UPK and Related Content	725
Prerequisites	726
UPK Player Package Links	727
Service Definitions	729
Related Content Menus and Frames	730
Creating UPK-Based Related Content Service Definitions	732
Modifying the UPK Player Package Link	732
Creating the UPK-Based Service Definition	732
Assigning UPK-Based Services as Related Content	734

Appendix G

Configuring OBIEE-Related Content Services	737
OBIEE Configuration Prerequisites	737

Appendix H

Consuming PeopleSoft Components as a Web Service in OBIEE	739
Understanding the OBIEE and the PeopleSoftComponents Web Service Interaction	739
Setting Up OBIEE to Consume PeopleSoft Components as a Web Service	740

Appendix I

Enabling Locale Switching Through Query Parameters in Oracle WebCenter	741
Registering Provider Classes	741

Appendix J

Configuring OBIEE Admin User Permissions in Enterprise Manager 11g	743
Configuring OBIEE Server Admin User Permissions in Enterprise Manager 11g	743

Appendix K

Language Support for Consuming and Producing Remote Portlets	745
Language Support for Consuming Remote Portlets	745
Language Support Rules for Producing Remote Portlets	745

Appendix L

PeopleTools Web Libraries	747
Understanding Web Libraries	747
PeopleTools Web Libraries	747

Index	751
--------------------	------------

PeopleTools Portal Technologies Preface

This preface discusses:

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

PeopleTools Portal Technologies

This book describes PeopleSoft Pure Internet Architecture and other technologies that enable you to perform tasks associated with the use and maintenance of the PeopleTools portal.

The terms *PeopleTools portal technology*, *PeopleTools portal*, *portal technologies*, and *portal* are used interchangeably throughout this PeopleBook to refer to the core portal technology PeopleTools.

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

PeopleBooks and the PeopleSoft Online Library

A companion PeopleBook called *PeopleBooks and the PeopleSoft Online 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 locally installed PeopleSoft online library, including web site folders.
- Understanding documentation integration and how to integrate customized documentation into the library.

- Application abbreviations found in application fields.

You can find *PeopleBooks and the PeopleSoft Online Library* in the online PeopleBooks Library for your PeopleTools release.

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.
- PeopleTools portal technologies implementation.

PeopleTools Portal Technologies Overview

PeopleTools portal technologies consists of PeopleSoft Pure Internet Architecture and the various systems, code, and standards 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 portals are built on top of PeopleSoft Pure Internet Architecture and enable you to easily access and administer content from a variety of providers, such as PeopleSoft FSCM, CRM, and HCM applications; other Oracle applications; and many third party content suppliers. PeopleTools portals enable you to combine content from these multiple sources and deliver the result to users in a unified, consistent, and intuitive 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, content search and management, navigation, and homepage or dashboard personalization.

PeopleTools Portal Technologies Implementation

To implement PeopleTools portals, 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.

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

See *PeopleTools 8.52 Installation Guides*, <https://support.oracle.com>.

See Also

PeopleTools 8.52: Getting Started with PeopleTools, "Getting Started with Enterprise PeopleTools"

Designing and Configuring the Portal Environment

In the designing phase of your implementation, take advantage of all Oracle resources for PeopleSoft applications including installation guides, hardware and software guides, certification requirements, red papers, and PeopleBooks.

Build the physical network and server architecture that is necessary to support your implementation. Ensure that you consider the various facets of network management and security, including creating subnets with appropriate routers and switches; creating firewalls, proxy servers, and reverse proxy servers, and using Secure Sockets Layer/Transport Layer Security.

During implementation, you should research and understand how to:

Step	Reference
Size servers appropriately	See your hardware configuration guides and PeopleSoft minimum requirements.
Configure software and hardware to optimize speed and performance	See your network administration guides.
Set up optimal load balancing	See your server configuration guides and industry best practices.

Installing PeopleSoft Pure Internet Architecture

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

Step	Reference
Run the PeopleSoft Pure Internet Architecture setup program for your database.	See <i>PeopleTools 8.52 Installation Guide</i> , https://support.oracle.com .

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 no existing profile meets your needs. To configure your selected web profile, you perform the following step:

Step	Reference
Configure your selected web profile.	<p>See Chapter 9, "Configuring the Portal Environment," Configuring Web Profiles, page 238.</p> <p>See <i>PeopleTools 8.52: System and Server Administration</i>, "Working with Oracle WebLogic."</p> <p>See <i>PeopleTools 8.52: System and Server Administration</i>, "Working with IBM WebSphere."</p>

Configuring Application Security

Applications come with roles and permissions that you can configure or use as delivered. Roles define the permission lists that are assigned to groups of people. A permission list defines 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:

Step	Reference
Configure permission lists, roles, and user profiles.	See <i>PeopleTools 8.52: Security Administration</i> .
(Optional) Set up single sign-on.	<p>See <i>PeopleTools 8.52: Security Administration</i>, "Implementing Single Signon."</p> <p>See Chapter 9, "Configuring the Portal Environment," Implementing Single Signon Functionality, page 281.</p> <p>See <i>PeopleTools 8.52: Security Administration</i>, "Employing LDAP Directory Services."</p>

Setting Up the Portal

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

Step	Reference
Manage portal definitions.	See Chapter 5, "Administering Portals," Administering Portal Definitions, page 104.

Step	Reference
Define folders and set folder security.	See Chapter 5, "Administering Portals," Administering Folders, page 107.
Define content references, create related links, and set content reference security.	See Chapter 5, "Administering Portals," Administering Content References, page 143.
Add templates and set template security.	See Chapter 5, "Administering Portals," Managing Portal Objects, page 176.
Create, configure, and secure homepages and dashboards.	See Chapter 10, "Administering Homepages, Dashboard Pages and Pagelets," page 301.

(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:

Step	Reference
Implement JSR 168 portlets.	See Chapter 21, "Working with JSR 168 Portlets," Describing the Portlet Entity Registry, page 693.

Configuring Performance Features

Properly configuring caching options significantly increases portal performance. To administer caching features for the portal, perform the following steps:

Step	Reference
Configure server-based caching.	See Chapter 8, "Using Portal Caching Features," Administering Server-Based Caching, page 222.
Configure browser-based caching.	See Chapter 8, "Using Portal Caching Features," Administering Browser-Based Caching, page 229.
Configure web server-based navigation caching.	See Chapter 8, "Using Portal Caching Features," Administering Web Server-Based Navigation Caching, page 233.

Chapter 2

Understanding PeopleSoft Pure Internet Architecture

This chapter discusses:

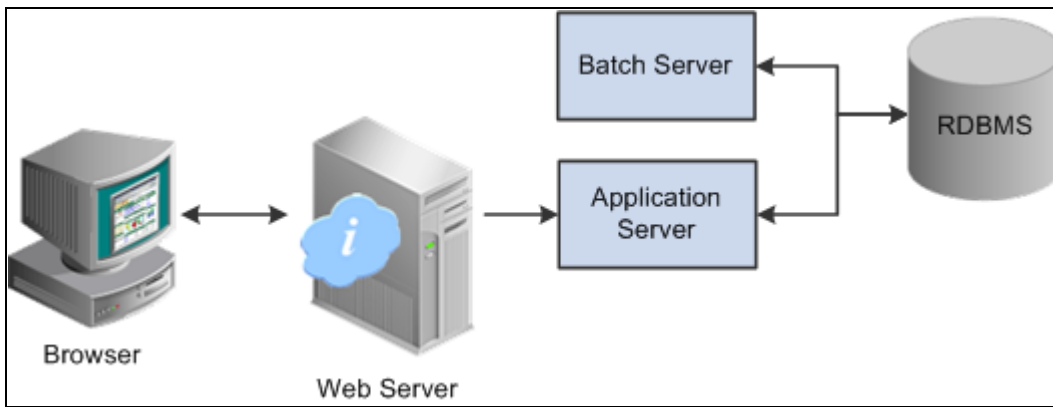
- PeopleSoft Pure Internet Architecture fundamentals.
- Database server.
- Application servers.
- Batch server environment.
- Web server.
- Web browser.
- Configuration and deployment options.

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.

You need 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 overview of 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 *PeopleTools 8.52 Installation Guides*, support.oracle.com.

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

PeopleTools 8.52: 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.

A one-to-one or a many-to-one relationship can exist 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 can configure multiple application server domains, with each domain connecting to the same PeopleSoft database. The opposite is not true; a single application server domain cannot be used to connect to multiple PeopleSoft databases.

For example, suppose that 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 in 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 that 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.

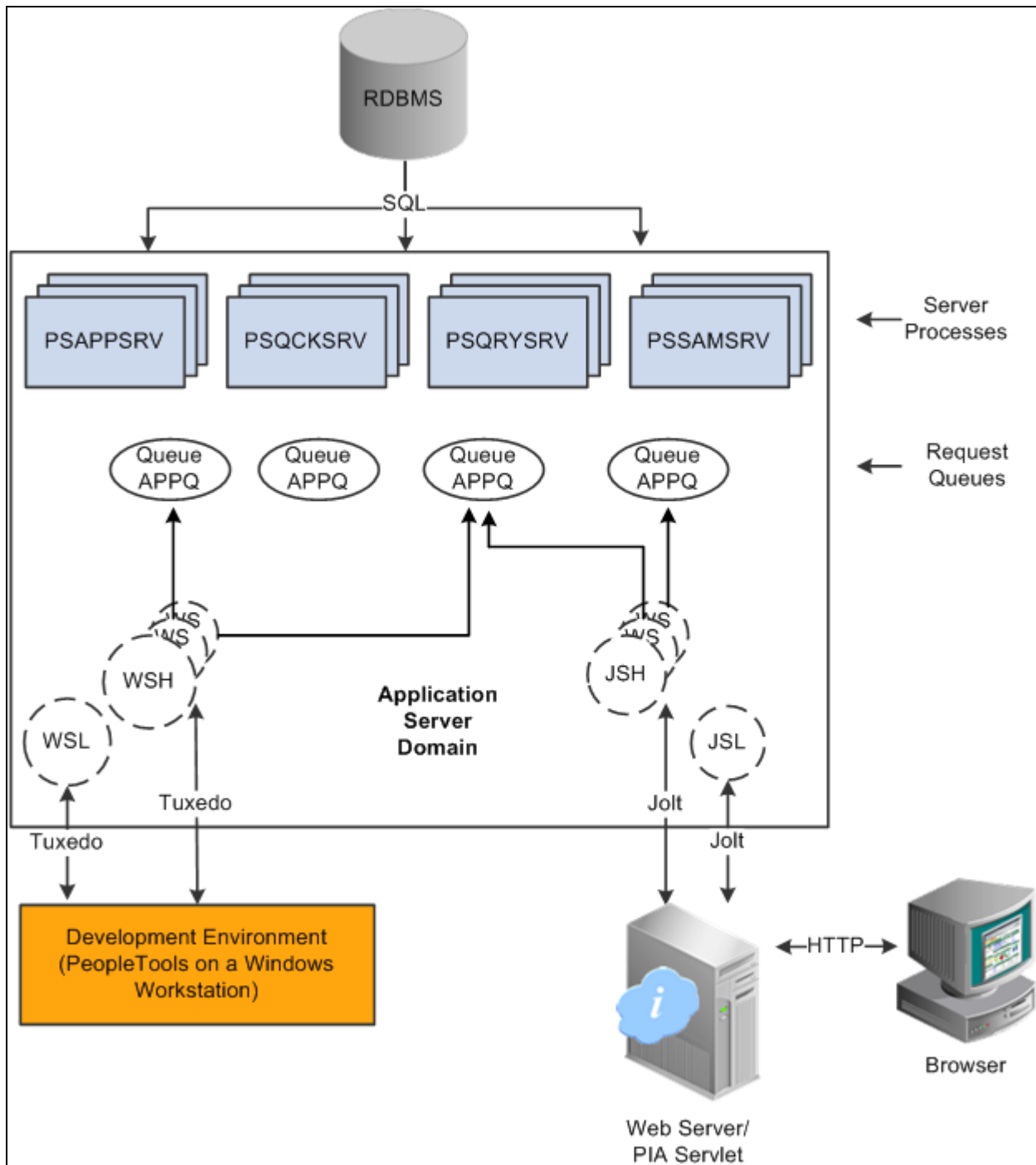
Starting an application server domain starts the set of server processes associated with that domain, such as PSAPPSRV, PSQCKSRV, and PSSAMSRV. Each server process establishes a persistent connection to a PeopleSoft database, and this connection acts as a generic SQL pipeline that the server process 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 you need to know how to configure multiple Java server handlers, you also need 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 omitted.

Note. When PeopleSoft architecture mechanics is discussed, the term *service* is 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:

<i>Item</i>	<i>Description</i>
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 that 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 an 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 that 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 and PSQCKSRV on QCKQ). 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 finish, then returns information to the device that initiated the request, such as a browser. While a server process waits for a service to finish, other transaction requests wait in a queue until the current service finishes. A service may take a fraction of a second to finish or several seconds, depending on the type and complexity of the service. When the service finishes, 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 start the application server domain. This helps you handle predicted workloads. Furthermore, Oracle Tuxedo can dynamically generate incremental server processes to handle increasing numbers of transaction requests. The capability to configure multiple server processes and generate 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.

- 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 starts, 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

PeopleTools 8.52: System and Server Administration, "Using the PSADMIN Utility," Using PSADMIN Configuration Files

Oracle Middleware

PeopleSoft software uses Oracle Tuxedo, a middleware framework and transaction monitor, to manage database transactions. PeopleSoft software 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 you are troubleshooting.

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

See *PeopleTools 8.52 Installation Guides*, support.oracle.com.

Database Connectivity

Application servers require database connectivity software to be 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 discusses:

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

Batch Server Environment

The batch server environment is where PeopleSoft Process Scheduler is 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 COBOL and Structured Query Report (SQR) executables are installed.

In a multiserver environment, you can decide where your site batch server environment resides. In PeopleSoft software, you can install the batch server on a separate server, or you can install the batch server on either the application server or the database server.

See Also

PeopleTools 8.52: 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 you need to choose a location that's supported in the PeopleSoft environment. At least two options are available 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 start, 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 manage the web server software, such as Oracle 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 manages all of the requests and formatting when users access PeopleSoft applications through the PeopleTools portal. It manages content searches, content access, 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.

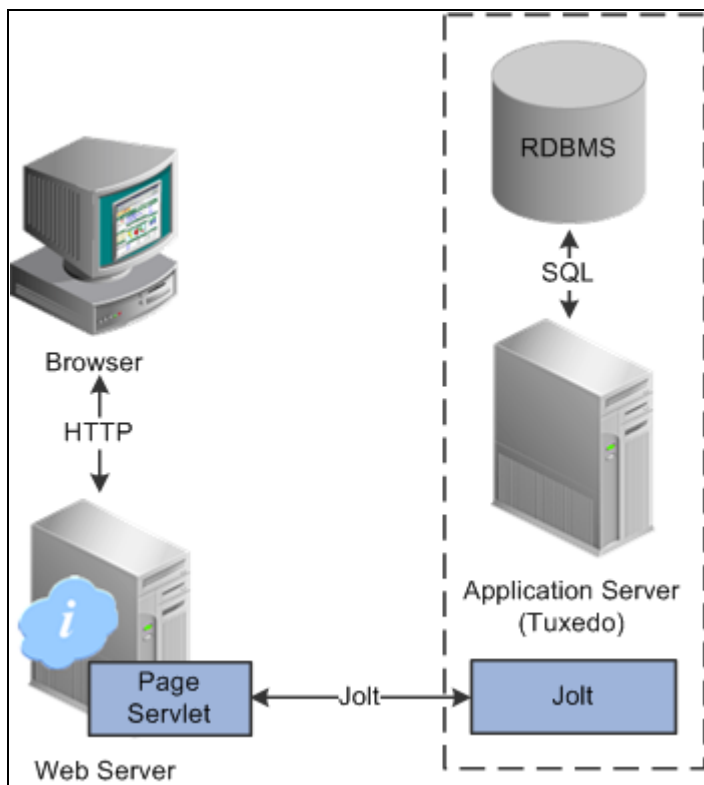
- Report repository servlet.

This servlet enables users to easily access and distribute the output of batch reports, such as Crystal and SQR reports, that are 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 an 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 items 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 uses a checksum to prevent tampering.

See *Troubleshooting Browser Limitations*, support.oracle.com.

Note. With PeopleSoft Pure Internet Architecture, no client software installation is necessary. Most processing occurs at the server level. You install the PeopleTools development applications in a Microsoft Windows environment. Development applications, such as PeopleSoft Application Designer and Data Mover, are usually intended for application developers and system administrators who need to access data other than application data. These users require workstations running Microsoft Windows.

Web Browser Cookies Setting

For you 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

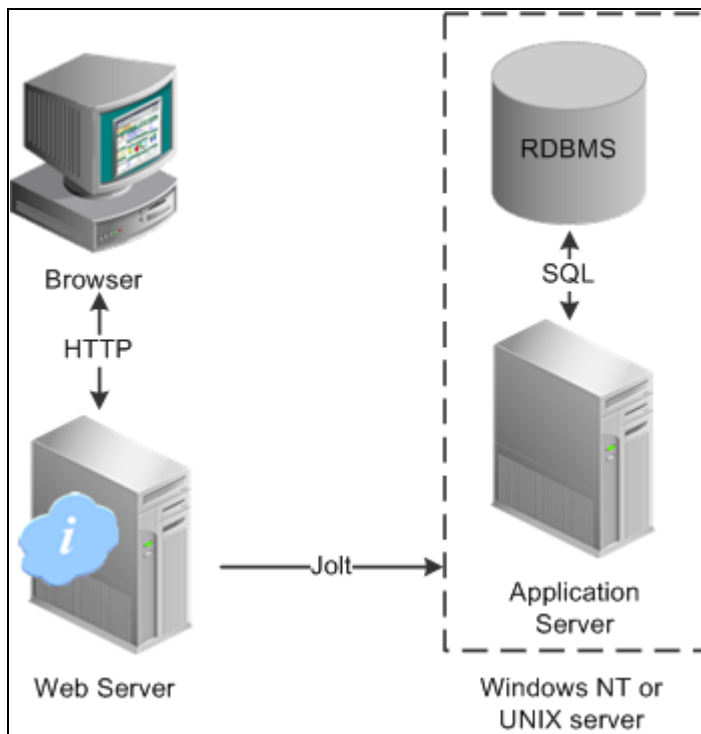
You can configure your environment to support either a physical or a logical application server configuration. In some cases, the PeopleSoft standard installation procedure recommends one or the other depending on the combination of database and operating system at your site.

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 is one in which 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

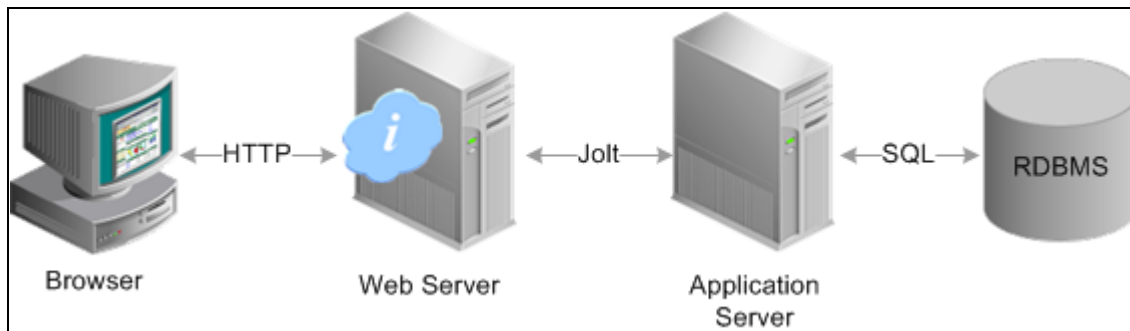
The dashed line surrounding the application server and the database server represents one physical machine. In this case, a logical application server configuration is possible only when both the database server and the application server are supported on a particular operating system. This shows that 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 is one in which each component resides on a separate machine. The following diagram depicts a physical application server configuration:



Physical application server configuration

Deployment Options

A variety of user deployment options exist.

PeopleTools Portal

You can use the PeopleTools 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 be standalone, or you can integrate it with any application 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 by 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. Oracle 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

PeopleTools 8.52: PeopleSoft Integration Broker, "Understanding PeopleSoft Integration Broker"

PeopleTools 8.52: PeopleSoft Component Interfaces, "Understanding Component Interfaces"

Chapter 3

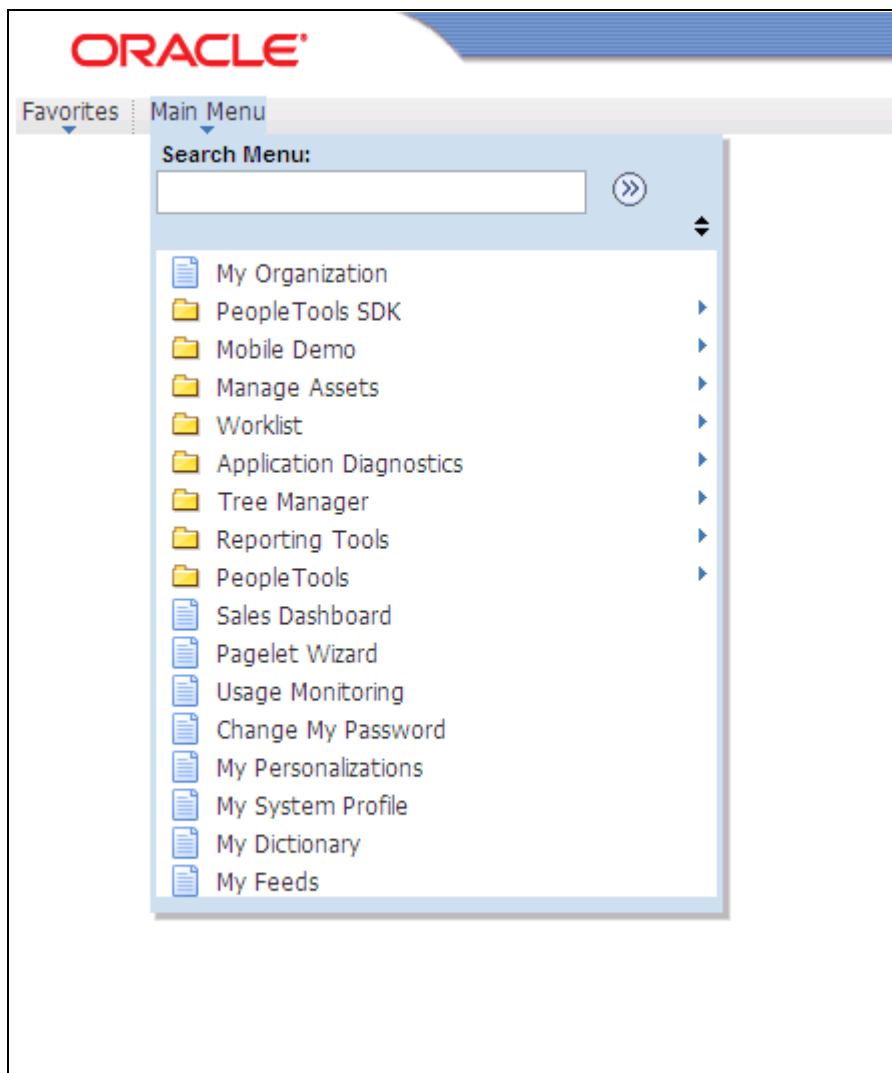
Understanding Portal Technology

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:



Example homepage with drop-down navigation and search

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-sign on 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 content or external references, and assemble pages.

The use of nodes simplifies PeopleCode programming because 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

PeopleTools 8.52: PeopleCode API Reference, "Portal Registry Classes," Node Class

PeopleTools 8.52: PeopleSoft Integration Broker Administration, "Adding and Configuring Nodes"

Portal Templates and Template Pagelets

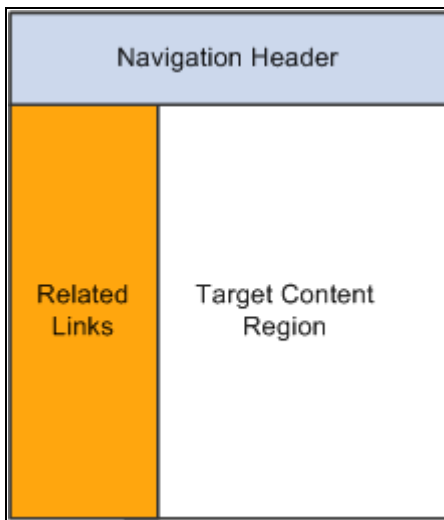
Developers create portal templates in PeopleSoft Application Designer. At runtime, a template is transformed into a web page by PeopleTools portal technology. Each template is made from various template pagelets.

Each template (and each template pagelet) is designed individually and stored as an HTML object in the application database. Oracle 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 composing 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.

Search	Enter a term or phrase in the Search field and click the Go button to search for a registered content reference within the system.
Home	Click to return to the homepage.
Worklist	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 appear to all users. See <i>PeopleTools 8.52: PeopleSoft Applications User's Guide</i> , "Using Workflow," Using Worklists to Receive Notifications.
MultiChannel Console	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. See <i>PeopleTools 8.52 : MultiChannel Framework</i> , "Configuring PeopleSoft MCF Agents" and <i>PeopleTools 8.52 : MultiChannel Framework</i> , "Managing Tasks and Using Chat in PeopleSoft MultiChannel Framework."

Performance Trace	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 appear to all users.
Add to Favorites	Click to add the current page to your favorites list for easy accessibility.
Sign out	Click to terminate the user session and return to the Sign In page.

See Also

PeopleTools 8.52: System and Server Administration, "Configuring Verity Search and Building Verity Search Indexes"

PeopleTools 8.52: PeopleSoft Applications User's Guide

PeopleTools 8.52: 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, that is 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: homepages, target content, templates, and pagelets. Although each type of content reference differs slightly from the other types, every content reference includes information, such as 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 website 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 basic PeopleTools security through the assignment of permission lists and roles.

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

Every portal registry contains a root folder and a Portal Objects folder. The Portal Objects folder includes the following folders:

- Homepage
- Navigation Collections
- Pagelets
- Templates

- Template Pagelets
- WSRP Produced Portlets.

In addition to these standard folders, one folder per PeopleSoft application is 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 taxonomy, or classification and navigation system, in which content references can be registered and managed for all portal users. PeopleTools provides a set of portal administration pages 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

Several ways exist to access and maintain the portal registry:

- Use the component registration wizard in PeopleSoft Application Designer to register and assign permissions to content references .
- Use the Menu Import utility to upgrade custom menu group definitions.
- Use the portal administration pages to manage portal and node definitions, folders, content references, templates, pagelets, navigation collections, and portlets.
- Use the security synchronization process to update the portal registry security based on the menu and script security.
- Use the portal registry API for programmatic access to the registry.

See Also

Chapter 9, "Configuring the Portal Environment," Importing Menu Groups into the Portal Registry, page 283

PeopleTools 8.52: Security Administration

PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide, "Using the Registration Wizard"

PeopleTools 8.52: PeopleCode API Reference, "Portal Registry Classes," Using the Portal Registry API

PeopleTools 8.52: PeopleSoft Integration Broker Administration, "Adding and Configuring Nodes"

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

Chapter 9, "Configuring the Portal Environment," Specifying the Node Type, page 281

PeopleTools 8.52: PeopleCode API Reference, "Portal Registry Classes," Node Class

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, envelopes the page with additional content, 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 accept only 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 incorporate 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 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 8.52 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 based on the query string parameter that was inserted there when the referring page was proxied.

2. The portal servlet checks the portal registry to determine whether a content reference exists for the target URL.

If a content reference exists 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 no node is associated with the content reference, then the default template for the portal is used. If no default portal template exists, 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 the request for target content is fulfilled. 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

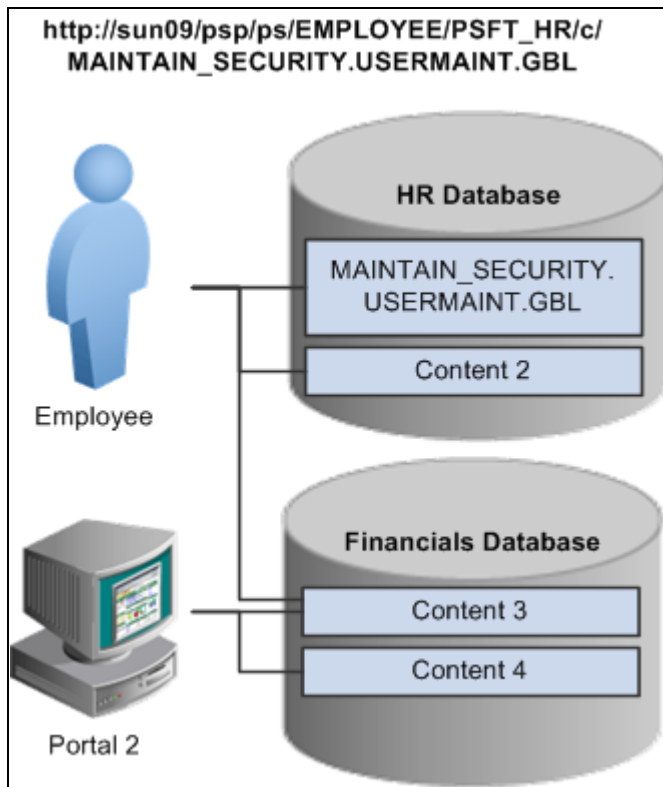
URLs provide the addresses for content so that it can be located and correctly identified. The portal servlet needs three pieces of information to present a page to the user. These are integral parts of a PeopleSoft portal URL:

- 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 determine 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 communicates with 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 with 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`

The following table describes the different sections of the URL:

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

Section	Description
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>Important! 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 that 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:

Content Type	URL Format	Example
Component	<i>/c/menu.component.market/?Page=page&Action=action &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

Content Type	URL Format	Example
Homepage	<i>/h/?tab=homepage_tab</i>	http://sun09/psp/ps/EMPLOYEE/PSFT_HR/h/?tab=HR homepage tab Note. This homepage URL tells the portal servlet to display 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 Note. 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 that 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 by means of 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 websites.

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 because 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. This sample HTML 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 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 contain 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. Note. The system detects whether the user is still active during the cmd=expire process on the server. The system ignores the session after detecting that the user is still active so that ongoing transactions in the active window are not disturbed. However, if another session is 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 to the sign-in page.

<i>URL</i>	<i>Description</i>
http://sun09/psp/ps/?cmd=login	Tells the servlet to return to the sign-in 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 that 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, you would 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 includes 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://sun0server/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 adds the following HTML to the response:

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

The absolute URL looks like this:

```
http://sunserver/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 be used to change a portal or node only if the HTML is being accessed through an HTML template. It won't work with a frame template because 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

PeopleTools 8.52: 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 applies only 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 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.

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 no ambiguity exists 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 whether 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%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

Working with PeopleTools Portals

Chapter 4

Working with Navigation Pages

Chapter 5

Administering Portals

Chapter 6

Modifying the Portal Interface

Chapter 7

Working with Portal Templates

Chapter 8

Using Portal Caching Features

Chapter 9

Configuring the Portal Environment

Chapter 4

Working with Navigation Pages

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

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

Note. The sample pages in this chapter use the default PeopleTools style sheet. Your page style might be different depending on your style sheet settings. Regardless of the style sheet, all page elements 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.

Three types of navigation pages are available:

- Standard navigation pages.

These pages are generated dynamically 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 facilitate 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.

- SmartNavigation pages.

These pages are generated dynamically based on hierarchical data and appear not as folders and links, but as an organization chart, which users can traverse to access transactions. The pages and processes that you use to configure SmartNavigation pages are not the same as those that you use to configure standard and custom navigation pages.

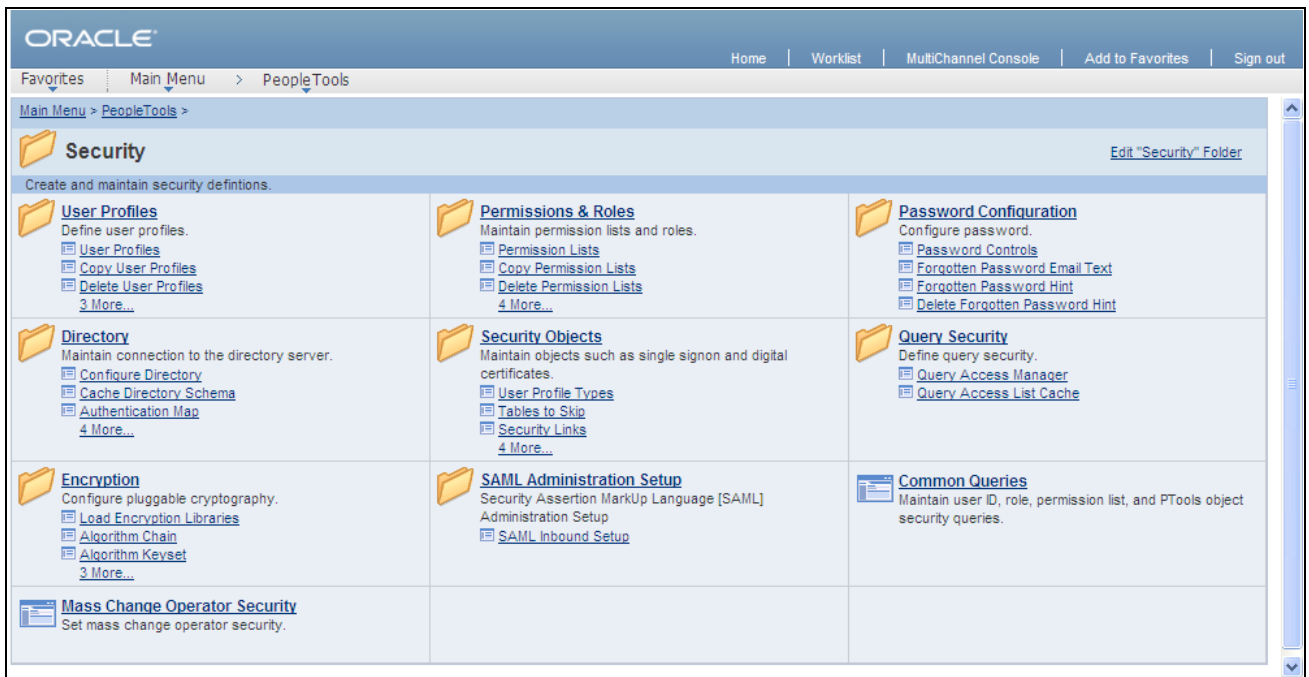
See [Chapter 5, "Administering Portals," Administering Folders, page 107](#) and [Chapter 5, "Administering Portals," SmartNavigation Pages, page 117](#).

The following sections further describe standard and custom navigation pages and discuss navigation page display caching.

Standard Navigation Pages

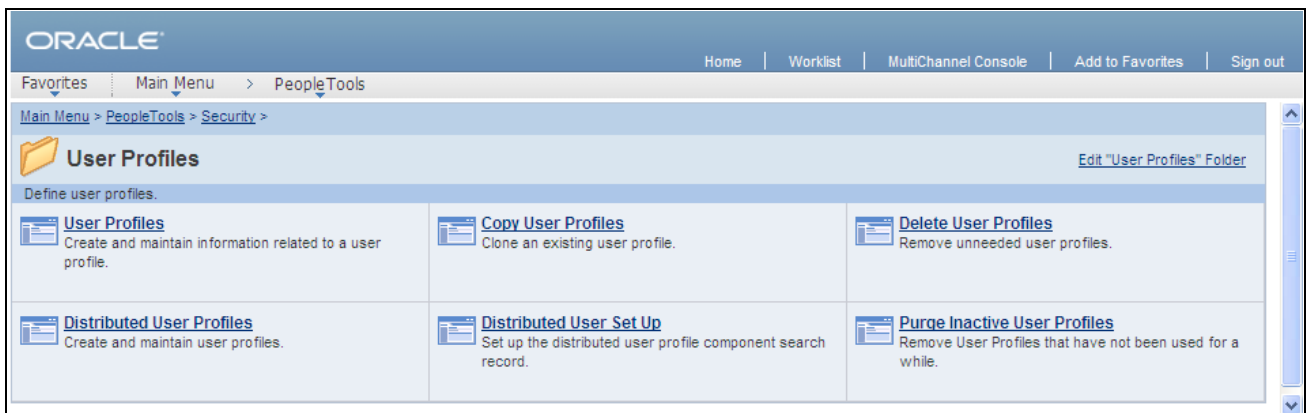
Standard navigation pages are available for every folder that appears in your menu navigation that has also been configured to display a standard navigation page. You enable the display of standard navigation pages at the registry level on the General Settings page. You can then disable the display of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



User Profiles standard navigation page

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

See [Chapter 4, "Working with Navigation Pages," Setting Up Navigation Collection Options, page 53.](#)

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 in to, select PeopleTools, Portal, Portal Definitions, then click Edit next to the portal that 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 that you designate as the folder navigation object is launched when the folder label is selected.

See [Chapter 5, "Administering Portals," Administering Folders, page 107](#) and [Chapter 5, "Administering Portals," Administering Portal Definitions, page 104](#).

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 the content of the menu navigation folder selected to display the page, 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 by 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 107](#) and [Chapter 5, "Administering Portals," Administering Content References, page 143](#).

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 using links in the menu navigation. When you click a menu navigation link to a custom navigation page, the page appears to the right of the menu navigation. For example, selecting the Context Manager Center link in the menu navigation displays the Context Manager Center custom navigation page as shown here:



Context Manager Center 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 component to which the content reference points.

See [Chapter 4, "Working with Navigation Pages," Creating or Editing a Navigation Collection, page 65.](#)

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 made in the navigation collection definition. Do not modify custom navigation pages using the Structure and Content component.

Note. When you save the navigation collection, modifications to the Placement Properties sequences are 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 74](#) and [Chapter 4, "Working with Navigation Pages," Editing a Link, page 79.](#)

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 via 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 the 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 navigation path 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 that you access by means of 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 53](#); [Chapter 4, "Working with Navigation Pages," Publishing Navigation Collections, page 81](#) and [Chapter 5, "Administering Portals," Administering Content References, page 143](#).

Understanding Navigation Collections

This section discusses:

- Navigation collections
- Navigation collection images

Navigation Collections

Navigation collections provide 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.

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.

After being defined, a navigation collection can be published as a navigation page or navigation pagelet for use on a homepage, dashboard page, or WorkCenter page.

When you publish navigation collections, 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 choose one of these two options for setting security access to published navigation collections:

- An automatic security synchronization process that will apply content reference and folder security as it is defined in the portal registry to the published navigation page or pagelet content reference.
- A manual process that entails you configuring the security access to published pages and pagelets

Note. Public access is allowed by default.

Navigation collections enable 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 that their user communities need to access on a frequent basis. They should also know where to find links to this content, and whether the content is accessed by portal navigation or external URLs.

Navigation collections can contain these elements:

- Links to existing nonhidden content references.
- Links to existing nonhidden 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 previously.

Navigation Collection Published as a Navigation Page

The following example shows a navigation collection that is published as a navigation page:



Context Manager Home navigation page

Top-level folders in the navigation collection appear as bold and linked section headings that organize the folders and content references they contain, as shown in these examples. However, the appearance of your published navigation collections depends on the assigned style sheet. The assigned style sheet must use style classes contained in the PTPP_SCSTYLEDEF style class definition.

Click the top-level folder link to access another navigation page that contains any folders and content references nested in the parent folder.

Note. Click the Edit <collection name> Collection link in the upper right corner of the navigation page to access the navigation collection definition on the Maintain Collection page. This link appears 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 a navigation collection that is published as a navigation pagelet:



Context Manager Center navigation pagelet

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

See Also

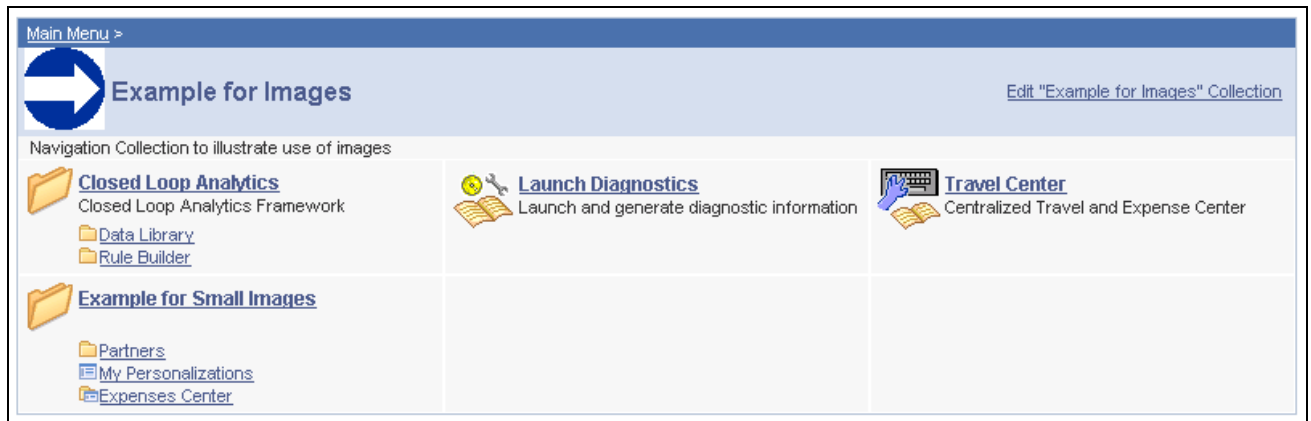
[Chapter 4, "Working with Navigation Pages," Publishing Navigation Collections, page 81](#)

[Chapter 4, "Working with Navigation Pages," Understanding Portal Utility Processes, page 89](#)

Navigation Collection Images

The images that 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 sample navigation collection page publication illustrates the different ways in which images can be displayed:



Example for Images page

In the sample 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 appear 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 appear 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 appear for navigation collections at the top level of a navigation collection publication.



This image was defined as the small folder image. Small folder images appear for folders nested within top-level folders on a navigation collection publication.



This image was defined as the small content image. Small content images appear 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 appear 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 a small image 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 appear until the element is accessed as a top-level 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 appears with its unique large image.

Common Elements Used on Navigation Pages

Breadcrumbs

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

See [Chapter 4, "Working with Navigation Pages," Setting Up Navigation Collection Options, page 53.](#)

Edit <folder name> Folder

The Edit <folder name> Folder link appears 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 111.](#)

Edit <Navigation Collection name> Collection

The Edit <Navigation Collection name> Collection link appears 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 *PeopleTools 8.52: PeopleSoft Applications User's Guide*, "Using PeopleSoft Application Pages," Using Spell Check.

Setting Up Navigation Collection Options

This section provides an overview of navigation collection options, lists common elements used in this section, and discusses how to:

- Define Portal Utilities system options.
- Define Portal Utilities registry options.

Note. These options are used by navigation collections, Feeds, and Pagelet Wizard pagelets.

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.

PeopleTools 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:

Options	System (default)	Registry (overrides system default)	Source Reference (overrides system and registry values)	Collection Reference (overrides system, registry, and source reference values)
Registry object prefix	X	None available	None available	None available
Show breadcrumbs	X	X	None available	None available
Show main menu breadcrumb	X	X	None available	None available
Registry node name	X	None available	None available	None available
Large (parent) image	X	X	X (with attribute name PTPP_IMAGE)	X
Small (child) image	X	X	None available	None available
Style sheet	X	X	None available	X
Show images	X	X	None available	X
Maximum child links	X	X	None available	X

Options	System (default)	Registry (overrides system default)	Source Reference (overrides system and registry values)	Collection Reference (overrides system, registry, and source reference values)
Maximum columns	X	X	None available	None available
Owner ID	X	None available	None available	X

You define and set navigation collection options on these pages:

- Set system options on the System Options page.
- Set registry options on the Registry Options page.
- Set source reference options through the Structure and Content pages.
- Set 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 97](#)

[Chapter 5, "Administering Portals," Managing General Portal Settings, page 171](#)

Common Elements Used in This Section

Show Breadcrumbs

Select to indicate that you want navigation path to appear 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 because the left-navigation menu is not updated to reflect the user's navigation through navigation pages.

The first breadcrumb in the Main Menu collection 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 displayed depend on the left-navigation menu folders that the user opened to access the navigation collection page.

Subsequent breadcrumbs displayed depend on the folder path that the user accesses on the navigation collection page.

Show Main Menu Breadcrumb

If you 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 deselect this option, the first entry in the breadcrumb path depends on the level-1 left-navigation folder that the user opens to access the navigation collection page. The Main Menu link does not appear as the first link in the breadcrumbs.

Main Menu Collection Name

This field appears when the Show Main Menu Breadcrumb option is selected. The default value is *Main Menu*, 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 that you have created.

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.

Style Sheet Name

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.

This setting applies to both navigation pages and navigation pagelets.

Note. You must use the same styleclass names that are found on the stylesheet PTPP_STYLEDEFN.

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Creating Style Sheet Definitions."

Maximum Columns

Enter the maximum number of columns to display on a navigation page.

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.

This setting applies only to navigation pages.

See *PeopleTools 8.52: PeopleSoft Applications User's Guide*, "Setting User Preferences," Personalizing Your Homepage.

Maximum Child Links

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, an 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.

For example, if you set Maximum Child Links to 4 and the navigation collection folder contains five links, three links appear on the navigation collection page along with the X More... link.

When you select the X More... link, a page containing all five links in the navigation collection folder appears.

This setting applies to both navigation pages and navigation pagelets.

Pages Used to Set Up Navigation Collection Options

Page Name	Definition Name	Navigation	Usage
System Options	PTPP_OPTIONS	PeopleTools, Portal, Portal Utilities, System Options	Define system-level default values for navigation collection options. These defaults may possibly be overridden 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. (Select 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

PeopleTools 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 value that the system prepends to registry structure object names (content and folder references) that are created when you publish navigation collections, template pagelets, and feed definitions.

For example, the object name of a published navigation pagelet has this format: <prefix>_SC_PGT_<object label>. The object name of a published navigation page has this format: <prefix>_SC_SP_<object 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 listed default node names:

PeopleSoft Enterprise Financials/SCM: *ERP*

PeopleSoft Enterprise CRM: *CRM*

PeopleSoft Enterprise HCM: *HRMS*

PeopleSoft Application 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 of the portal registry tables.

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 means of 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

Show Images	<p>Select one of these values:</p> <p><i>Always Show Images:</i> Select if you always want images to appear on navigation pages and pagelets.</p> <p><i>Never Show Images:</i> Select if you never want images to appear on navigation pages and pagelets.</p> <p><i>Only Show Defined Images:</i> Select if you want only specifically defined images to appear 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-level and registry-level default images are not displayed.</p>
Large Folder Image	Select the default image that you want to display for top-level folders on a navigation page or pagelet.
Large Content Image	Select the default image that you want to display for top-level content references on a navigation page or pagelet.
Large Collection Image	Select the default image that you want to display for top-level navigation collection references on a navigation page or pagelet.
Small Folder Image	Select the default image that you want to display for a folder within a folder on a Navigation page or pagelet.
Small Content Image	Select the default image that you want to display for a content reference within a folder on a Navigation page or pagelet.
Small Collection Image	Select the default image that you want to display for a navigation collection within a folder on a Navigation page or pagelet.

Pagelet Wizard

Package Name	Specify the application package name to use for the display format.
Path	Enter the names—delimited by colons—of each subpackage in the hierarchy that drills down to the class that contains the method.
Application Class ID	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. (Select 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: 


*Style Sheet Name: 

*Maximum Columns:

*Maximum Child Links:

Navigation Image Defaults


*Show Images: 

*Large Folder Image: 



*Large Content Image: 




*Large Collection Image: 




*Small Folder Image: 



*Small Content Image: 



*Small Collection Image: 



[Delete registry Override](#)

Registry Options page

All options that 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.](#)

Registry Name

Select the portal registry for which you want to define navigation collection options that will override system-level defaults.

Navigation Image Defaults

Show Images

Always Show Images: Select if you always want images to appear on navigation pages and pagelets accessed by means of the selected portal registry.

Never Show Images: Select if you never want images to appear on navigation pages and pagelets accessed by means of the selected portal registry.

Only Show Defined Images: Select if you want only specifically defined images to appear on navigation pages and pagelets accessed by means of 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-level and registry-level default images are not displayed.

Large Folder Image

Select the image that you want to display for top-level folders on a Navigation page or pagelet accessed by means of the selected portal registry.

Large Content Image

Select the image that you want to display for top-level content references on a Navigation page or pagelet accessed by means of the selected portal registry.

Large Collection Image

Select the image that you want to display for top-level navigation collection references on a Navigation page or pagelet accessed by means of the selected portal registry.

Small Folder Image

Select the image that you want to display for a folder within a folder on a Navigation page or pagelet accessed by means of the selected portal registry.

Small Content Image

Select the image that you want to display for a content reference within a folder on a Navigation page or pagelet accessed by means of the selected portal registry.

Small Collection Image

Select the image that you want to display for a navigation collection within a folder on a Navigation page or pagelet accessed by means of the selected portal registry.

Delete Registry Override

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"> • PeopleTools, Portal, Portal Utilities, Navigation Collections • Click the Return to Search link on the Maintain Collections page. 	Find existing navigation collections.
Maintain Collection	PTPP_SCMAINTCOLL	<ul style="list-style-type: none"> • Click the Edit link for an existing navigation collection on the Find an Existing Collection page. • Click the Add Collection link on the Find an Existing Collection page. • Click the Edit <Navigation Collection name> Collection link on a custom navigation page. 	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.

Page Name	Definition Name	Navigation	Usage
Select Source Folder	PTPP_SCBROWSEREG	<ul style="list-style-type: none"> Click the Browse Portal Registry icon on the Add Folder page. Click the Browse Portal Registry button on the Edit Folder page. 	View a graphical display of a selected portal registry and select the content reference that you want to add to your navigation collection.
Browse Image Catalog	PTPP_SCIMGSRCH	<ul style="list-style-type: none"> Click the Browse Image Catalog button on the Add Folder page. Click the Browse Image Catalog button on the Edit Folder page. Click the Browse Image Catalog button on the Add Link page. Click the Browse Image Catalog button on the Edit Link page. 	Search for and select an image to associate with the selected folder or content reference. This image appears 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"> Click the Browse Portal Registry button on the Add Link page. Click the Browse Portal Registry button on the Edit Link page. Click the Move button on the Publish Collections page. 	<p>View a graphical display of a portal registry.</p> <p>When this page is 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.

Finding Existing Navigation Collections

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

Navigation Collections

Find an Existing Collection

Search By:

[Add Collection](#)

Search Results	Personalize Find First 1 of 1 Last
BI Publisher	Edit Delete

Find an Existing Collection page

Search by

Select one of these values:

Description: Select to conduct your search based on navigation collection description text.

Name: Select to conduct your search based on navigation collection name text.

Use the *begins with* and *contains* options to limit your search results.

Add Collection

Select to access the Maintain Collection page, where you can define a new navigation collection.

Search Results

Edit

Select to access the Maintain Collection page, where you can edit an existing navigation collection.

Delete

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.

Note. The Delete button does not appear for the Main Menu navigation collection. While you can edit this navigation collection, you should not delete it because 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.

Creating or Editing a Navigation Collection

Access the Navigation Collections page. (Click the Edit link for an existing navigation collection on the Find an Existing Collection page).

The screenshot shows the 'Navigation Collections' page with the 'Maintain Collection' tab selected. The page title is 'Navigation Collections'. Below the title is a descriptive paragraph: '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.'

The 'Collection Properties' section contains the following fields:

- *Name:** BI Publisher
- Description:** (254 Characters)
- *Valid from date:** 01/01/1900
- Valid to date:**
- Owner ID:** PeopleTools

The 'Override Default Options' section shows a tree view with the following structure:

- BI Publisher
 - Register Data Sources
 - Define BIP Reports
 - Run BIP Reports
 - Complete Setup

Buttons for 'Add Link' and 'Add Folder' are located next to the tree view. At the bottom of the page is a link labeled '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

Name Enter a name for the navigation collection. This text appears as the label on the published navigation page or pagelet.

Description

Enter a description of the navigation collection. This text can appear in several locations.

If the navigation collection is published as a navigation page, the description text appears when the mouse cursor is moved over the menu item link that is used to access the page. The description text also appears on the navigation page.

If the navigation collection is published as a navigation pagelet, the description text appears on the navigation pagelet.

Valid from date/Valid to date

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.

Oracle recommends that you leave the Valid to date field clear if you anticipate that the navigation collection should remain active indefinitely.

Owner ID

Use this value to identify navigation collections created by your organization when you are 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 .

If a default value has been set at the system or registry level, it appears, but can be overridden.

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

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 that 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.](#)

Style Sheet Name

Select the style sheet for this navigation collection.

Maximum Child Links

Enter the maximum number of child links to display.

Note. If more links are available for this collection, a More link appears.

Show Images

Select one of these values:

Always Show Images: Select to always display images on pages and pagelets for this navigation collection.

Never Show Images: Select if you don't want images to appear on pages and pagelets for this navigation collection.

Only Show Defined Images: 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-level and registry-level default images are not displayed.

Note. If you select the *Only Show Defined Images* option, small images do not appear on your navigation collection pages or pagelets because small images can be defined only as system-level and registry-level defaults.

Collection Image

Select the large image that you want to display as the top-level image for this navigation collection.

The navigation collection that 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 appears 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 that 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 appears 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 that 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 appears 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 appears when you select a content reference link in the navigation collection hierarchy.

Add Folder

Click to access the Add Folder page. This option appears 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 that 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 appears 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 appears when you select any folder other than the root node folder in the navigation collection hierarchy. This deletes the folder only from the navigation collection.



Designates a folder for which content is user-defined. These folders are added on the Add Folder page by means of 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 for which content is menu-based. These folders are added on the Add Folder page by means of 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:

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)

[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 date and to date for the folder.

Source Portal

This field appears 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 in which a navigation collection will be copied between portals, and the folder reference that you are defining should automatically refer to the current portal.

See [Chapter 5, "Administering Portals," page 97](#).

Source Folder

This field appears 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 that you want to reference.

Label

If you are adding a menu-based folder, this field is populated with the label on the source, which 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, this field is populated with the description on the source, which can be overridden in the Override Description field.

This text appears when the mouse cursor is moved over a folder link and 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 appears as the large image for the folder on the navigation collection publication.

When defined for folders nested within folders, the image appears 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 appears for users who can access the Structure and Content pages 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 source reference security.

See [Chapter 5, "Administering Portals," Administering Folders, page 107](#).

Selecting a Source Folder

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

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 in which a navigation collection will be moved between portals, and the folder reference that you are defining should automatically refer to the current portal.

See [Chapter 5, "Administering Portals," page 97](#).

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* results, but may return results with *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* results, but may return results with *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: User Defined Folder

Label: Register Data Sources

Description: (254 Characters) Register existing processes that will supply XML data for reports.

Override Options

Override Image: PT_FN_REPORTING_TOOLS
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.

Move to New Parent Folder

Parent Folder: BI Publisher

Placement in Folder: 01 - First

OK Cancel

Edit Folder page

With the exception of the Placement Properties group box, which is discussed subsequently, 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

Move to New Parent Folder	Click to move the folder that you are editing to a folder other than the folder designated in the Parent Folder field. You can change the placement of a folder reference only after you have initially saved the navigation collection.
Parent Folder	Displays the parent folder of the folder that you are editing.
Placement in Folder	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:

Description:
(254 Characters)

☐ Open in a new window ☐ Replace Window

Additional Parameters:

Example: name1=value1&name2=value2

Override Options

Override Image:

Override Label:

Override Description:
(254 Characters)

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 in which a navigation collection will be copied between portals, and the content reference that you are defining should automatically refer to the current portal.

See [Chapter 5, "Administering Portals," page 97](#).

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 that 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 appears when the mouse cursor is moved over the content reference link.

Open in a new window

Select to open the page in a new browser window. This option and Replace Window are mutually exclusive.

See [Chapter 17, "Creating and Configuring WorkCenter Pages," Managing WorkCenter Page Configurations, page 633](#).

Replace Window

Select to replace the current browser window, including navigation and target area, with the new content. Replace window updates the template and navigation to display the template and navigation of the target content reference. This option and Open in new window are mutually exclusive.

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 eliminates the need 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 appears 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 appears for users who can access the Structure and Content pages 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 date and from date, and content provider for the selected source content reference.

See [Chapter 5, "Administering Portals," Administering Content References, page 143.](#)

Selecting a Source Link

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

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 in which a navigation collection will be moved between portals, and the content reference that you are defining should automatically refer to the current portal.

See [Chapter 5, "Administering Portals," page 97](#).

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* results, but may return results with *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: PT_PTPP_WIZARD_GBL

Label: Pagelet Wizard

Description: Define Pagelet Wizard pagelets.
(254 Characters)

☒ Open in a new window ☐ Replace 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: Manage Template Pagelets

Placement in Folder: 02 - After Navigation Collections

OK Cancel [Find Source](#)

Edit Link page

With the exception of the Placement Properties group box, which is discussed subsequently, 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 75.](#)

Placement Properties

Move to New Parent Folder Click to move the content reference that 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 that you are editing.

Placement in Folder

Select a value to change the placement of the content reference within its selected navigation collection folder.

Available placement values reflect saved content reference sequencing.

Publishing Navigation Collections

This section discusses how to publish navigation collections.

Pages Used to Publish Navigation Collections

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Publish Collection	PTPP_SCPUBLISHCOLL	PeopleTools, Portal, Portal Utilities, Navigation Collections, Publish Collection	Publish the selected navigation collection to a navigation pagelet, page, or both.
Select Source Folder	PTPP_SCBROWSEFLDR	Click the Move button on the Publish Collection page.	Select the parent folder in which you want the navigation page to appear in the left navigation menu.

Publishing a Navigation Collection

Access the Publish Collection page. (Select PeopleTools, Portal, Portal Utilities, Navigation Collections, Publish Collection.)

Maintain Collection
Publish Collection

Name: Manage Template Pagelets

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:** PeopleTools Documentation

[Structure and Content](#)

☒ **Navigation Page**

Security Access

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

☒ **Enable Caching**

Move EMPLOYEE > Root

[Return to Search](#)

Publish Collection page

Name Displays the name of the navigation collection defined on the navigation 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.

Security Access

Select one of these values:

Public Access: Select to give all users access to the pagelet. This option is selected by default.

Allow Collection Sync (allow collection synchronization): Select to make the pagelet eligible for processing by the Sync Collection Security process.

Do Not Allow Collection Sync (do not allow collection synchronization): Select if you do not want the pagelet security to be updated by the Sync Collection Security process. Select this option to ensure that security that you manually define for the pagelet cannot be overridden by the process.

See [Chapter 4, "Working with Navigation Pages," Understanding Portal Utility Processes, page 89.](#)

Enable Caching

Select to have role-based caching enabled for this published pagelet. Selecting this option adds portal caching attributes to the published navigation pagelet.

See [Chapter 8, "Using Portal Caching Features," page 221.](#)

Pagelet Category

Select the category in which you want the navigation pagelet to appear for selection on the Personalize Content page.

Available pagelet categories are defined in the Structure and Content component. Select the Portal Objects folder link, then select the Pagelets folder link. Select the Add Folder link to add a pagelet category folder.

See [Chapter 5, "Administering Portals," Managing Portal Objects, page 176.](#)

Structure and Content

Select to access the Content Ref Administration page, where you can access details about the content reference created for the navigation pagelet publication. For example, you can check the results of the Sync Collection Security (synchronize collection security) Application Engine process (PTPP_SCSYNC) and change content reference label text or manually update the security.

See [Chapter 4, "Working with Navigation Pages," Understanding Portal Utility Processes, page 89](#) and [Chapter 5, "Administering Portals," Administering Content References, page 143.](#)

Navigation Page

Select to publish the navigation collection as a navigation page, which can be accessed from the menu.

Security Access

Select one of these values:

Public Access: Select to give all users access to the page. This option is selected by default.

Allow Collection Sync (allow collection synchronization): Select to make the page eligible for processing by the Sync Collection Security process.

Do Not Allow Collection Sync (do not allow collection synchronization): Select if you do not want the page security updated by the Sync Collection Security process. Select this option to ensure that any security that you manually define for the page is not overridden by the process.

See [Chapter 4, "Working with Navigation Pages," Understanding Portal Utility Processes, page 89.](#)

Enable Caching

Select to have role-based caching enabled for this published page. Selecting this option adds portal caching attributes to the published navigation page.

See [Chapter 8, "Using Portal Caching Features," page 221.](#)

Move

Click to access the Select Source Folder page, where you can select the parent folder in which you want the navigation page to appear in the left navigation menu.

See [Chapter 4, "Working with Navigation Pages," Selecting a Source Folder, page 71.](#)

Structure and Content

Select to access the Content Ref Administration page, where you can access details about the content reference created for the navigation page publication. For example, you can check the results of the Sync Collection Security process and change navigation page label text.

See [Chapter 4, "Working with Navigation Pages," Understanding Portal Utility Processes, page 89](#) and [Chapter 5, "Administering Portals," Administering Content References, page 143.](#)

Publishing Pagelets

This section discusses how to:

- Publish a pagelet.
- Publish pagelets to multiple portals.

Note. The Publish Pagelets component can be used with Pagelet Wizard and Portal Pack pagelets.

Pages Used to Publish Pagelets to Portals

Page Name	Definition Name	Navigation	Usage
Publish to Multiple Portals	PTPP_PMPUBPRTL	PeopleTools, Portal, Portal Utilities, Publish Pagelets, Publish to Multiple Portals	Publish pagelets to multiple portals.
Publish Multiple Pagelets	PTPP_PMPUBPGLT	PeopleTools, Portal, Portal Utilities, Publish Pagelets, Publish Multiple Pagelets	Publish multiple pagelets to a portal.

Publishing Pagelets to Multiple Portals

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

Publish to Multiple Portals
Publish Multiple Pagelets

Publishing a pagelet entails copying it from the Source Portal to the Target Portal. Selecting a Target Portal where the pagelet already exists will overwrite the Target Portal's pagelet.

Pagelet

*Source Portal: EMPLOYEE Employee-facing registry content
Pagelet Type: ☒ Homepage Pagelet ☐ Template Pagelet
*Pagelet Name: PT_PSRF_REPRT_PAGELET_GBL My Reports

Target Portals

Customize | Find | View All | First 1-5 of 5 Last

Select	Portal Name	Pagelet Exists As	
<input checked="" type="checkbox"/>	CUSTOMER	My Reports	Go to Tab Content / Layout
<input checked="" type="checkbox"/>	EMPLOYEE	My Reports	Go to Tab Content / Layout
<input type="checkbox"/>	PARTNER	My Reports	Go to Tab Content / Layout
<input type="checkbox"/>	PS_SITETEMPLATE	My Reports	Go to Tab Content / Layout
<input type="checkbox"/>	SUPPLIER	My Reports	Go to Tab Content / Layout

☒ Select All
☐ Clear All

Publish Pagelet

Publish to Multiple Portals page

Pagelet

Source Portal	Select the portal from which to copy a pagelet. You can select from portals that reside in the same database where you are currently signed in.
Pagelet Type	<p>Select the type of pagelet that you want to copy. Options are:</p> <p><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.</p> <p><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 Context Manager or WorkCenter pages are used to display the template pagelet.</p>
Pagelet Name	Select a pagelet that exists in the selected Source Portal registry. This is the pagelet content reference that you want to publish to one or more other portal registries that reside in the database where you are currently signed in.

Target Portals

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

Select	Select to publish the selected pagelet to the selected target portal. Use the Portal Name and Title fields to identify the available target portals.
Pagelet Exists As	<p>If the pagelet that you 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>
Go to Tab Content/Layout	Click to display the Tab Content and Tab Layout pages where you can specify display settings for the pagelet.

Publish Pagelet

Select to copy the pagelet definition from the source portal and publish it in the selected target portals.

Content reference are created in the registries of all selected target portals by means of the same values that exist in the original pagelet registration in the source portal.

If the folder in which the original pagelet is registered does not exist for a selected target portal registry, it is generated dynamically.

Publishing Multiple Pagelets to a Portal

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

Publish to Multiple Portals

Publish Multiple Pagelets

Publishing a pagelet entails copying it from the Source Portal to the Target Portal. To overwrite pagelets that already exist in the Target Portal, use the Publish to Multiple Portals page.

Pagelet Criteria

*Source Portal:

EMPLOYEE

Employee-facing registry content

*Target Portal:

CUSTOMER

Customer-facing registry content

Pagelet Type:

☒ Homepage Pagelet
 ☐ Template Pagelet

Pagelet Keyword:

Search

Reset

Pagelets

Customize | Find | View 100 | First 1-10 of 130 Last

Select	Pagelet ID	Title
<input type="checkbox"/>	ADMN_USERROLES_HMPG	User Roles
<input type="checkbox"/>	ADMN_WC_QUERY_ROLES_HMPG	My Roles
<input type="checkbox"/>	BART_SCHEDULE	Bart Schedule
<input type="checkbox"/>	CALCULATOR	Calculator
<input type="checkbox"/>	CALENDAR	Calendar
<input type="checkbox"/>	CURRENCY_CONVERTER	Currency Converter
<input type="checkbox"/>	DICTIONARY	Dictionary
<input type="checkbox"/>	DISPLAYCOOKIES	Display Cookies
<input type="checkbox"/>	EPSPF_E_WOAPR_GBL	Work Order Approval (D)
<input type="checkbox"/>	EP_AP_CFOP_AVGMAIN_GBL	Average Days in Payables

☒ Select All
 ☐ Clear All

Publish Multiple Pagelets page

Pagelet Criteria

Source Portal	Select the portal from which to copy a pagelet. You can select from portals that reside in the same database where you are currently signed in.
Target Portal	<p>Select the portal to which you want to copy pagelets. You can select from portals that reside in the same database where you are currently signed in.</p> <p>After you select the source and target portals, the Pagelet group box displays a list of pagelets that are registered in the source portal but not the target portal.</p>
Pagelet Type	<p>Select the type of pagelet that you want to copy. Options are:</p> <p><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.</p> <p><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 Context Manager or WorkCenter pages are used to display the template pagelet.</p>
Pagelet Keyword	Enter keywords to narrow the scope of pagelets displayed in the Pagelet group box.

Pagelets

Select	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.
Publish Selected Pagelets	<p>Select to copy the pagelet definition from the source portal and publish it to any selected target portals.</p> <p>The portal registry entries for the content references are created in the target portal registry by means of 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 generated dynamically.</p>

Running Portal Utility Processes

This section provides an overview 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, roles, or both 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 that are 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 81.](#)

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 or 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 *deleted* 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 Application 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 Application Portal.

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

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

Warning! Navigation collections cannot be rendered in the PeopleSoft Application Portal if the content provider registry is hosted by a nonlocal 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 nonlocal node. If you want to render a content provider navigation collection page or pagelet in the PeopleSoft Application Portal, the content provider's registry must be hosted by a local node in the content provider database.

See [Chapter 5, "Administering Portals," page 97](#).

Pages Used to Run Portal Utility Processes

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
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. (Select 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](#)

Request Parameters

*Portal Name: Customer-facing registry content

Collection Name:

Synchronize Collection Security page

Request Parameters

Portal Name	Select the portal for which you want to synchronize security for published navigation collections.
Collection Name	<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 nonpublic navigation collection publications that exist in the selected portal.</p> <hr/> <p>Note. 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 81

Running the Delete Empty Folders Process


Access the Delete Empty Folders page. (Select 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: 

Delete Empty Folders page

Request Parameters**Portal Name**

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. (Select 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:  Composite PeopleTools Project

*Portal Name:  Employee-facing registry content

*Content Provider Name:  [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="text"/>

☒ Include Collection Pages

Select Collection Pages

☐ All Collection Pages
☒ Selected Collection Pages

	*Collection Page Name	Collection Page Label
1	<input type="text"/>	<input type="text"/>

☒ Include Folders

Select Folders

☐ All Folders
☒ Selected Folders

	*Folder Name	Folder Label
1	<input type="text"/>	<input type="text"/>

Clean Portal Project page (2 of 2)

Project Name

Select the portal project that you want to clean for export into the PeopleSoft Application 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 that you want to copy. Available portal names are derived from the portal project that you selected, as well as from 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 that you want to include in your portal project. Available node names are derived from the portal project that you selected.

Node Definition

Select to access the Node Definitions - Portal Content page. If you select the Include Pagelets option, you must have a URL value in the node definition to enable the Clean Portal Project process to correctly include personalization pages for the pagelets that you copy over to the PeopleSoft Application Portal. Access the Node Definitions - Portal Content page to enter the content URI text for the content provider node name selected on this page.

When you click this link, you are prompted to save or cancel any changes that you made in the component, and you are then transferred to the Node Definitions - Portal Content page.

See [Chapter 9, "Configuring the Portal Environment," Defining Portal Nodes, page 279.](#)

Project Options**Full Navigation**

Select to run the Clean Portal Project process on all objects in the portal project.

Select Objects

Select to run the Clean Portal Project process on select objects in the portal project. When you select this option, the Select Objects group box appears, enabling you to select the type of navigation that you want to include in your portal project.

Use this option if you don't want to copy and maintain the full content provider navigation structure.

Select Objects**Include Pagelets**

Select to include pagelets (including navigation collection pagelets) in the portal project in the run of the Clean Portal Project process. When you select this option, the Select Pagelets group box appears, enabling you to specify which pagelets you want to be processed.

The actual published navigation collection pagelet content references are included for copying to the PeopleSoft Application Portal. The navigation collection definitions are not included.

Pagelet personalization pages are included in the portal project if the content provider node name contains the correct URL. Click the Node Definition link to access the Node Definitions - Portal Content page, where you can enter this URL.

Include Collection Pages

Select to include navigation collection pages in the portal project in the run of the Clean Portal Project process. When you select this option, the Select Collection Pages group box appears, enabling you to specify which navigation collection pages you want to be processed. The actual published navigation collection pages are included for copying to the PeopleSoft Application Portal. The navigation collection definitions are not included.

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 appears, 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 Application 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.

Chapter 5

Administering Portals

This chapter provides overviews of portal administration, folder navigation, and discusses how to:

- Administer portal definitions.
- Administer folders.
- Administer SmartNavigation 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

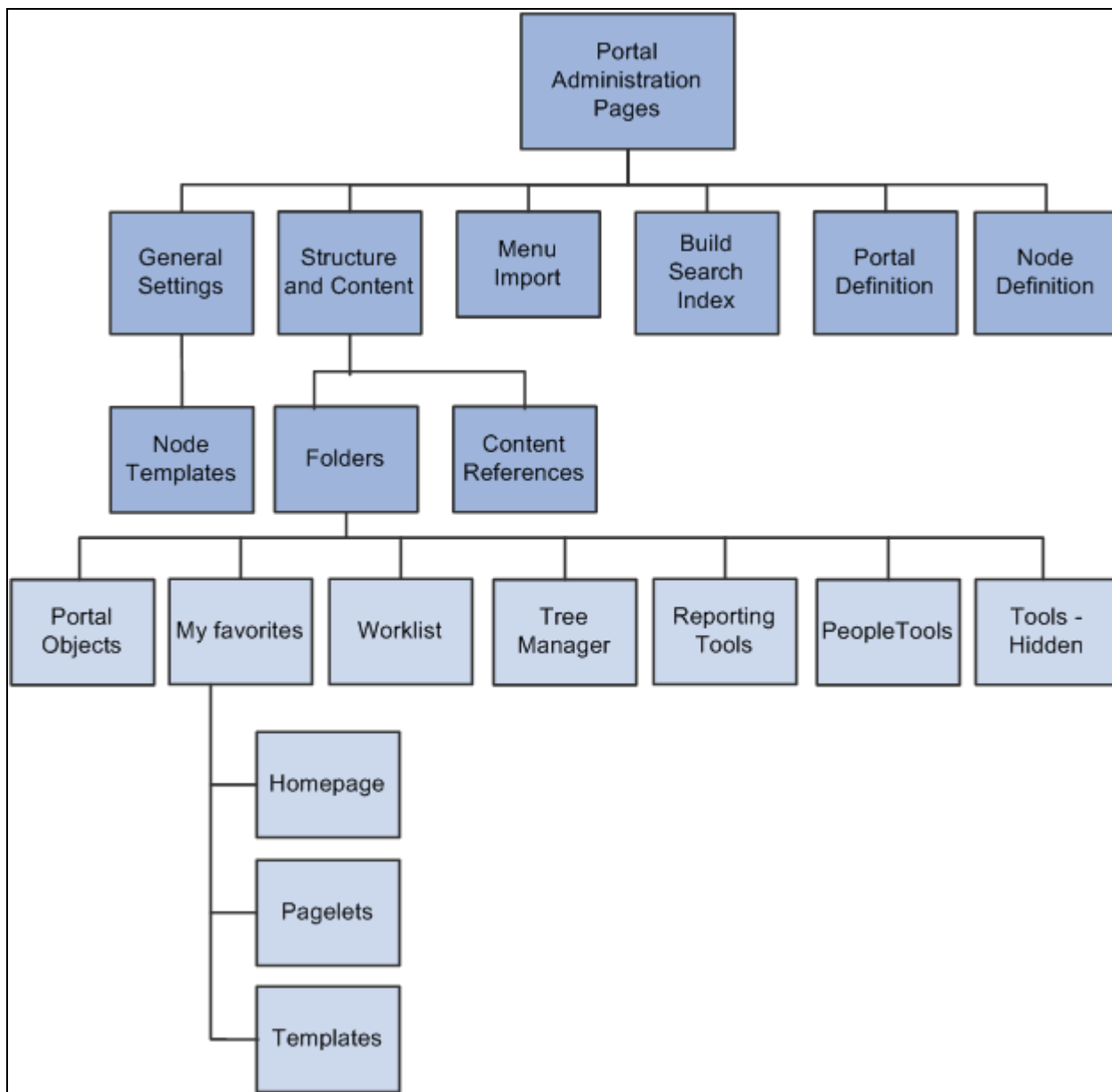
Every component delivered with PeopleSoft applications is registered as a content reference in the portal registry . Common administrative tasks include adding, deleting, and renaming portal definitions. If you have portal administrator access, you can use the administration pages to manage the registry,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

Additionally, to provide programmatic access to the portal registry, PeopleTools 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. Developers can also register components by using the Registration Wizard, which is part of PeopleSoft Application Designer.

You use the administration pages to:

- Manage the portal registry, or the structure and content of a portal, which includes:
 - Templates.
 - Navigation collections.
 - Folders.
 - Content references.
 - Pagelets.
 - Homepages.
 - Dashboard pages.
- Manage general portal settings, which include node templates, the portal template, navigation options, portal search options, and so on.
- Add, edit, and delete local and remote portal definitions.
- Add, edit, and delete local and remote node definitions.
- Populate a portal registry with custom menus by importing menu definitions.
- Generate a Verity search index or collection that reflects the content of the portal registry.

The following diagram shows the organizational structure of the portal administration component and folders. Note the location of the special Portal Objects folder. This folder contains folders and content references for all templates and pagelets associated with a given portal.



Portal administration pages

Note. To view the portal administration pages, you must have the appropriate security permissions.

See Also

PeopleTools 8.52: PeopleSoft Integration Broker Administration, "Adding and Configuring Nodes"

Understanding Folder Navigation

PeopleSoft applications integrate folders into the menu, or navigational, structure. While most forms of navigation in PeopleSoft applications use folders, the term *folder navigation* has a specific meaning for PeopleTools portals. Folder navigation is the default method of navigation delivered in PeopleSoft applications and refers to a specific response by the portal when the user clicks a folder, which is rather than expanding and collapsing menu items, the portal generates navigation pages in the target content area.

Navigation Pages

Note these points about the characteristics of navigation pages:

- Navigation pages provide users with an organized, graphical approach to traversing the menu structure.
- Navigation pages, like menus, reflect the structure of the portal registry. Each navigation page corresponds to a folder in the portal registry and provides access to any subordinate folders and content references.
- Users click folders to navigate through the menu structure to other navigation pages.
- Breadcrumb links appear in the header of the navigation page to indicate position in the hierarchy of folders. Click a breadcrumb link to return a higher-level navigation page.
- Navigation pages provide access to access transaction pages when users click a content reference link.
- The character and appearance of navigation pages is controlled by a default folder navigation object (page) that is standard with every PeopleTools implementation.
- You can modify the appearance of any navigation page by changing its default folder navigation object.
- The folders and content references that appear on navigation pages vary by user according to his or her assigned permissions.

This sample navigation page, which uses the default folder navigation object, shows the Security folder when folder navigation is enabled:



Navigation page for the Security folder

Traversing Menus Without Engaging Navigation Pages

If you use one of these methods to navigate the menu structure, the portal will not generate navigation pages.

- *For drop-down navigation:* Click the folder name. This action causes the menus to cascade sideward revealing subfolders and content references.
- *For left navigation:* Click the triangle icon to the left of the menu name. This action causes the menus to expand downward revealing submenus and content references.

Important! You enable folder navigation at the portal definition and general portal settings levels; folder navigation must be enabled at both levels for the feature to work. When you enable folder navigation at the site level, you can navigate all folders by using this method; however, you can disable or override folder navigation at the individual folder level.

Note. If folder navigation is *disabled*, then navigation pages do not appear when the user clicks a folder. Instead, the item expands to display any subordinate menu items. The user continues to expand menu items by clicking them until he or she clicks a content reference. No content appears in the target area until the user clicks the content reference.

See Also

[Chapter 5, "Administering Portals," Adding and Editing Portal Definitions, page 106](#)

[Chapter 5, "Administering Portals," Managing General Settings for Portals, page 172](#)

[Chapter 4, "Working with Navigation Pages," Understanding Navigation Pages, page 43](#)

Working with Permission Lists, *PeopleTools 8.52 PeopleBook: Security Administration*

Working with Roles, *PeopleTools 8.52 PeopleBook: Security Administration*

Common Elements Used in This Chapter

Chart Navigation Object Name

At the portal definition and general settings levels, specify the portal object that creates the general appearance and character of SmartNavigation folders. The system default value is

PT_PORTAL_DEFAULT_CHART_NAV, which is a hidden iScript-based content reference that renders the basic SmartNavigation page.

At the individual folder level, specify an object that overrides the system-level chart navigation object.

If you don't specify a value at the folder level, the portal uses the system-level chart navigation object.

Copy object

Click to access the Copy Object to a Different Portal Registry page, which enables you to copy an existing folder to a different portal registry. This button appears only when you're editing a folder.

Default Chart Navigation Page

Select to replace the standard navigation page with chart-based SmartNavigation page. This field is disabled by default. To enable this field, select a value in the Data Source field.

If you select this check box, then the application uses *PT_ABN_ORGCHART* component as the default PeopleTools chart navigation page. If you deselect this check box, then you must enter your own value in the Folder Navigation Object Name field.

Folder Navigation Object Name

At the portal definition and general settings levels, specify the content reference that the system launches when the user clicks a folder icon. The system default value is *PT_PTPP_SCFNAV_BASEPAGE_SCR*, which is a hidden iScript-based content reference that renders the standard navigation page.

At the individual folder level, specify an object that overrides the system-level default.

If you don't specify a value at the folder level, the portal uses the system-level folder navigation object.

Important! You must use the definitions associated with the *PT_ABN_ORGCHART* component as the basis of any custom SmartNavigation folder object.

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Creating Custom SmartNavigation Pages."

Note. If you select a content reference that's hidden from portal navigation, the folder navigation occurs, but the menu name that you click remains highlighted.

Hide From Portal Navigation Select to make this folder invisible to users.

Is Folder Navigation Disabled	<p>Select to override the system-level default setting and disable folder navigation for individual folders.</p> <hr/> <p>Note. If you disable and then re-enable folder navigation at the system level, this field is automatically cleared.</p> <hr/>
Is Folder Navigation Enabled	<p>Select to enable folder navigation for all folders in the current portal registry or selected portal definition.</p> <p>When folder navigation is enabled, if the user clicks the folder name in the menu structure, the navigation page or a content reference that you specify in the Folder Navigation Object Name field appears in the target area.</p>
Label	Enter the text to appear as the link for this folder.
Long Description	Enter text to appear when the cursor rests over the link for this folder. If you leave this field blank, the label appears.
Name	<p>Enter a name for the folder.</p> <hr/> <p>Note. The name can consist of letters, digits, and underscores (_). It cannot contain any spaces and cannot begin with a digit.</p> <hr/>
(Object) Owner ID	Select the owner of the portal object such as portal, folder, or content reference. The owner ID helps determine the team that created or modified the object. You can define unique owner IDs for your organization by entering field translate values for the OBJECTOWNERID field using PeopleSoft Application Designer. Do not use PeopleSoft owner IDs.
Parent Folder	The name of the current parent folder. This field is display-only.
Product	(Optional) Portal-aware applications can use this field for group processing. All PeopleTools objects have the product name <i>PT</i> .
Select New Parent Folder	Click to copy the folder to a new location within the current portal registry. This button appears only when you're editing a folder.
Sequence number	Enter a sequence number to define where this folder appears in its parent folder, and therefore where its associated menu entry appears within its parent menu. If you leave the sequence number blank, the folder is added to the top of the list of folders and is displayed in alphabetical order. You can change the sequence only one item at a time, and each time you do so, you must save your changes to see the result.
Valid from date and Valid to date	Enter a valid from date. The valid to date is optional and is omitted by default. These dates can be used for application-specific processing by portal-aware applications and are used by the portal navigation to determine visibility.

Administering Portal Definitions

This section lists common elements used in this section and discusses how to:

- Define portals.
- Add and edit portal definitions.
- Change default portals.

Common Elements Used in This Section

Hosted by this node	The node where the portal is defined. This is either a remote or local node. The node must have been defined previously on the Maintain Node Definitions page.
Portal Name	The name to appear as the portal label. This value is used for the browser window title when a long description has not been entered.

Pages Used to Administer Portal Definitions


<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Portal Definitions	PORTAL_DEFINITIONS	PeopleTools, Portal, Portal Definitions	Define portals from other databases and associate them with a node.
Add Portal	PORTAL_DEFN_ADD	Click the Add button on the Portal Definitions page.	Create a new portal definition.
Edit Portal	PORTAL_DEFN_ADD	Click the Edit button on the Portal Definitions page.	Update existing portal definitions.
Removal Confirmation	PORTAL_DEFN_DEL	Click the Delete button on the Portal Definitions page.	Delete portal definitions.

Defining Portals

Access the Portal Definitions page (PeopleTools, Portal, Portal Definitions).

Portal Definitions

Current Portal: EMPLOYEE

Portal Definitions				Customize	Find		First	1-7 of 7	Last
Portal Type	Portal Name	Hosted by this node	Default	Edit	Add	Delete			
Local	CUSTOMER	ERP	<input type="checkbox"/>	<button>Edit</button>	<button>Add</button>	<button>Delete</button>			
Local	EMPLOYEE	LOCAL_NODE	<input checked="" type="checkbox"/>		<button>Add</button>				
Local	MOBILE	LOCAL_NODE	<input type="checkbox"/>	<button>Edit</button>	<button>Add</button>	<button>Delete</button>			
Local	PARTNER	LOCAL_NODE	<input type="checkbox"/>	<button>Edit</button>	<button>Add</button>	<button>Delete</button>			
Local	PORTAL	LOCAL_NODE	<input type="checkbox"/>	<button>Edit</button>	<button>Add</button>	<button>Delete</button>			
Local	PS_SITETEMPLATE	LOCAL_NODE	<input type="checkbox"/>		<button>Add</button>				
Local	SUPPLIER	ERP	<input type="checkbox"/>	<button>Edit</button>	<button>Add</button>	<button>Delete</button>			

Portal Definitions page

You can have multiple portals on a database or create one portal on multiple databases. You can define portals from other databases and associate them with a node. This supports transfers between portals and redirected requests for a portal not supported by the local web server.

Portal Type	Displays the attribute of the selected node. Possible types are <i>Local</i> and <i>Remote</i> .
Default	Select to use this as the default node. Only local nodes can be the default. If you change the default, be sure to click the Save button.
Edit	Click to update the portal definition. <hr/> Note. This button does not appear for the site template or the local node that is currently in use. To modify the current portal, use the General Settings page. <hr/> See Chapter 5, "Administering Portals," Managing General Settings for Portals, page 172.
Add	Click to add a new portal definition. The new portal is listed in alphabetical order.
Delete	Click to delete a portal definition. You cannot delete the site template or node that is currently in use.

See Also

[Chapter 5, "Administering Portals," Managing General Portal Settings, page 171](#)

Adding and Editing Portal Definitions

Access the Portal Definitions - Add Portal page (Click the Add button on the Portal Definitions page).

Add Portal page

Note. The Edit Portal page contains the same fields as the Add Portal page (Click the Edit button on the Portal Definitions page).

Long Description

Enter a description to be used by screen readers if accessibility mode is enabled.

This description is also used as the browser window title. If you don't enter a description in this field, the portal name is used as the browser window title.

Default Template Name

Select the template to use for this new portal.

Folder Navigation

Use these fields to enable folder navigation and define the objects associated with folder navigation for portals other than the one you are currently signed in to. To enable folder navigation for the current portal, use the General Settings page.

Important! Folder navigation is not enabled unless you select the Is Folder Navigation Enabled check box on both the Portal Definition and General Settings pages. Is Folder Navigation Enabled is selected by default in the standard PeopleTools portals.

See [Chapter 5, "Administering Portals," Managing General Settings for Portals, page 172.](#)

Changing Default Portals

You can change the default portal in two ways:

- Change the default portal in the node definition.
- Insert the new default portal name into the URL.

Changing the Default Portal in the Portal Definition

To change the default portal:

1. In the current default portal, select PeopleTools, Portal, Node Definitions.
2. Select the default local node and then click the Portal tab.
3. Select the Default check box for the portal that you want to open.
4. Save.
5. Sign out.
6. Shut down and restart the portal web server.
7. Sign in.

When you sign in, the new default portal is invoked.

Changing the Default Portal by Inserting the New Portal Name in the URL

You can also change the default portal by changing the URL. To change the URL, insert the new portal name instead of the existing portal name. For example, to change to the supplier portal from the employee portal, make the following substitution:

URL before the change: `http://localhost/ps/ps/EMPLOYEE/`
`QE_LOCAL/c/UTILITIES.PORTAL_DEFINITIONS.GBL`

URL after the change: `http://localhost/ps/ps/SUPPLIER/`
`QE_LOCAL/c/UTILITIES.PORTAL_DEFINITIONS.GBL`

Note. This type of portal change is valid until the browser session is closed.

Administering Folders

This section provides an overview of folder administration and discusses how to:

- Define folders.
- Set folder permissions.

Understanding Folder Administration

In the portal registry, folders organize content references into a multilevel hierarchy. A folder can contain content references and other folders. When you create folder definitions, you explicitly configure the properties, attributes, and position of the folder in the hierarchy. The portal uses your configurations to create the menu structure, which remains static unless you modify structure in the portal registry.

Organization of the Portal Registry

You use the Structure and Content component to access the portal registry, where you create and manage folders and content references. When you click the Structure and Content link, the application opens the portal registry at the root level. The initial set of folders, content references, and links that you see correspond to the high level folders and links that appear directly under Main Menu or in the Menu pagelet.

The Structure and Content component enables you to organize and manage your folders and content references; all folders and content references that share the same Structure and Content page are at the same level hierarchically and share the same parent folder. You click the Edit link to access the properties of a folder. You click the folder label link to navigate one level lower in the hierarchy where any child folders and content references that exist appear.

As you navigate through the hierarchy of folders, breadcrumbs appear at the top of the page directly beneath the Main Menu navigation bar. The breadcrumbs appear to visually indicate your current position in hierarchy of the portal registry. Click a link in the breadcrumbs to return to the folder. In this sample Structure and Content page, the breadcrumbs indicate that you are viewing elements that appear in the PeopleTools, Security folder:

[Root](#) > [PeopleTools](#) > [Security](#)

Structure and Content

* Click the folder label to view the child folders and content references for that folder

* Click the "Edit" link to edit the folder definition

▼ Folders			
Personalize Find View All First 1-8 of 8 Last			
Label	Edit	Sequence number	
User Profiles	Edit	100	Delete
Permissions & Roles	Edit	110	Delete
Password Configuration	Edit	115	Delete
Directory	Edit	120	Delete
Security Objects	Edit	130	Delete
Query Security	Edit	140	Delete
Encryption	Edit	150	Delete
SAML Administration Setup	Edit	160	Delete

[Add Folder](#)

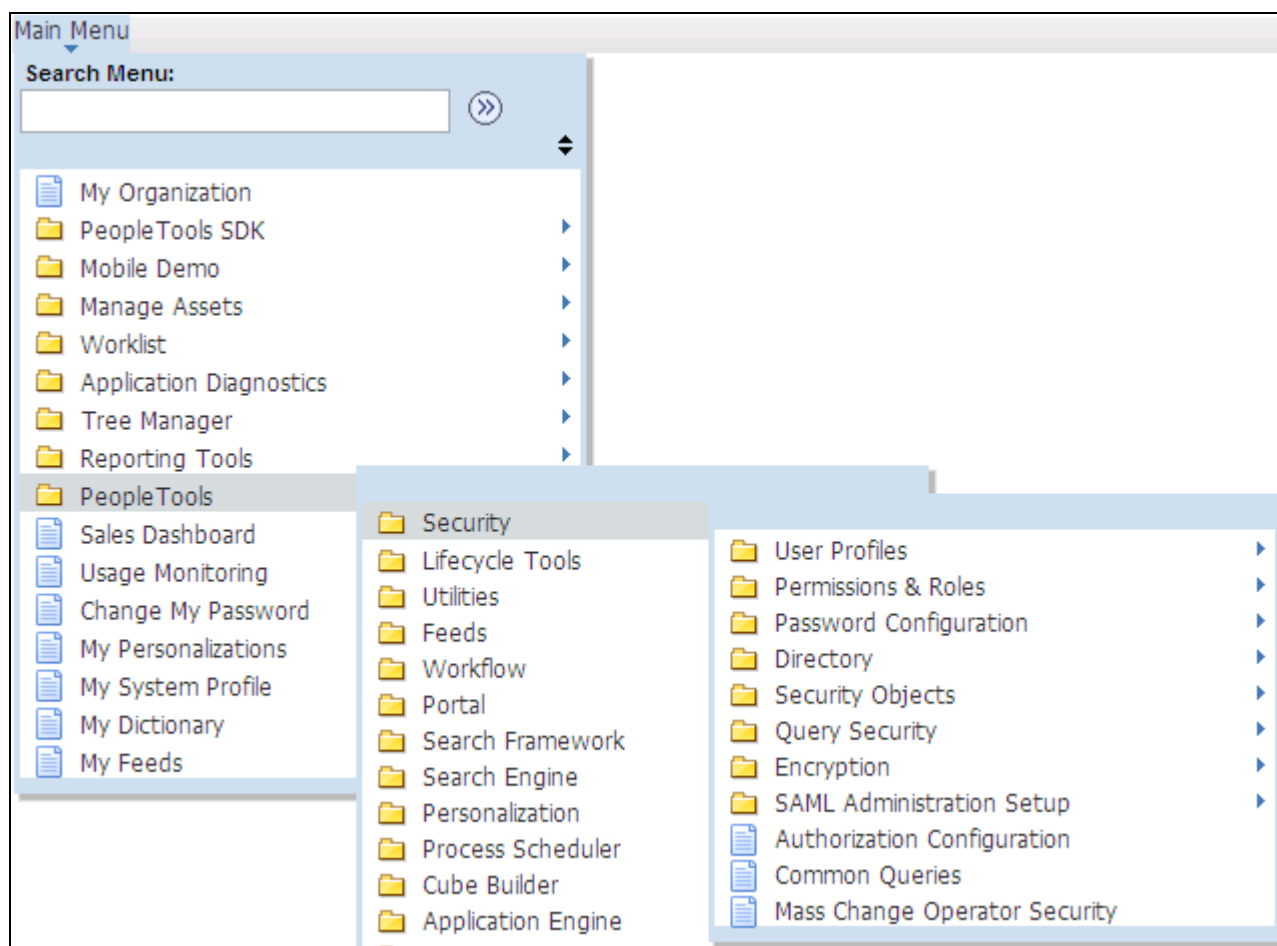
* Click the "Edit" link to edit the content reference definition

▼ Content References						
Personalize Find View All First 1-3 of 3 Last						
Link	Label	Edit	Sequence number	Create Link	Number of links	
<input type="checkbox"/>	Authorization Configuration	Edit	0	Create Link	0	Delete
<input type="checkbox"/>	Common Queries	Edit	200	Create Link	0	Delete
<input type="checkbox"/>	Mass Change Operator Security	Edit	201	Create Link	0	Delete

[Add Content Reference](#) [Add Content Reference Link](#)

Structure and Content page showing the folders and content references that appear in the Security folder

The portal uses the hierarchy of folders in the portal registry to create the menu structure of the PeopleSoft application. Menus open to reveal the folders and content references in the same general sequence and location as they appear in the portal registry. In this sample drop-down menu, notice that the organization of folders and content references in the PeopleTools, Security folder correspond to the folders and links located in the Security folder in the portal registry:



Folder and content reference organization in the PeopleTools, Security folder

Folders can appear at any level in the menu hierarchy in both the left navigation Menu pagelet and the Main Menu drop-down navigation. Folders cascade open when users click the folder label and generate navigation pages when you click the folder icon. Most PeopleSoft application folders are standard navigation folders; however, you can configure certain folder properties to create SmartNavigation folders.

See Also

[Chapter 5, "Administering Portals," Administering Folders, page 107](#)

Pages Used to Administer Folders

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Folder Administration	PORTAL_FLDR_ADM	PeopleTools, Portal, Structure and Content Click Add Folder link at the bottom of the Folders group box to add a new folder. Alternatively, click the Edit link to edit an existing folder.	Define folders.
Folder Security	PORTAL_FLDR_SEC	PeopleTools, Portal, Structure and Content. Access the Folder Administration page. Select the Security tab.	Set folder permissions.

Defining Folders

Access the Folder Administration page (Select PeopleTools, Portal, Structure and Content Click the Add Folder link).

Folder Administration

Folder Security

[Root](#) >

Folder Administration

*Name: Parent Folder: Root

*Label:

Long Description:
(254 Characters)

Product: *Valid from date: 06/21/2010 Creation Date: 06/21/2010

Sequence number: Valid to date: Author: VP1

Object Owner ID

☐ Hide from portal navigation

Folder Navigation

☐ Is Folder Navigation Disabled ☐ Default Chart Navigation Page

Folder Navigation Object Name:

Application Navigation

Data Source: ☐ Display as Content Reference

Folder Attributes

Name: ☒ Translate

Label:

Attribute value:

Folder Administration page

1. Enter values in the Folder Administration fields.
2. (Optional) Enter values in the Folder Attributes fields.
3. If this folder is a SmartNavigation folder, enter values in the Application Navigation fields.
See [Chapter 5, "Administering Portals," Defining SmartNavigation Folders, page 138.](#)
4. Set the folder security.
See [Chapter 5, "Administering Portals," Setting Folder Permissions, page 114.](#)
5. Save the folder.

Folder Administration

This table describes the primary fields that appear on the Folder Administration page:

Name	Enter a unique identifier for the folder. <hr/> Note. The name can consist of letters, digits, and underscores (_). It cannot contain any spaces and cannot begin with a digit. <hr/>
Label	Enter the text to appear as the link for this folder.
Copy object	Click to access the Copy Object to a Different Portal Registry page, which enables you to copy an existing folder to a different portal registry. This button appears only when you're editing a folder.
Parent Folder	The name of the current parent folder. This field is display-only.
Select New Parent Folder	Click to copy the folder to a new location within the current portal registry. This button appears only when you're editing a folder.
Long Description	Enter text to appear when the cursor moves over the link for this folder. If you leave this field blank, the label appears.
Product	(Optional) Portal-aware applications can use this field for group processing. All PeopleTools objects have the product name <i>PT</i> .
Sequence number	Enter a sequence number to define where this folder appears in its parent folder, and therefore where its associated menu entry appears within its parent menu. If you omit the Sequence number, the folder is added to the top of the list of folders and is displayed in alphabetical order. You can change the sequence only one item at a time, and each time you do so, you must save your changes to see the result.
Valid from date and Valid to date	Enter a valid from date. The valid to date is optional and is omitted by default. These dates can be used for application-specific processing by portal-aware applications and are used by the portal navigation to determine visibility.

Application Navigation

Use these fields to create SmartNavigation folders.

See [Chapter 5, "Administering Portals," Defining SmartNavigation Folders, page 138.](#)

Folder Attributes

Folder attributes can be used by portal-aware applications for application-specific processing that adds functionality to folders.

1. Specify the attribute name.
2. (Optional) Specify the attribute label.
3. Enter the attribute value.
4. Select the Translate check box to make an attribute label and attribute value translatable.

This table describes the fields that you use to define folder attributes.

Name	Enter the attribute name. The attribute must be an attribute that is recognized by the portal. If you enter an attribute, then you must also enter an attribute value.
Label	Enter text to describe the attribute. This field is required only if the Translate check box is selected.
Attribute Value	Enter the value that the portal should use in conjunction with the attribute.
Translate	Select to translate the label.

See Also

PeopleTools 8.52: PeopleSoft Applications User's Guide, "Working With PeopleSoft Applications," Using Menu Pagelets

Chapter 5, "Administering Portals," Content Reference Attributes, page 150

Setting Folder Permissions

Access the Folder Security page (Select PeopleTools, Portal, Structure and Content. Navigate the portal registry to the appropriate folder, click the Edit link, and then select the Folder Security tab.)

Folder Security page

Public	Select to make the folder available to all users. When this check box is selected, all other fields except Label are hidden.
---------------	--

Author Access Select to restrict folder access to the creator of the folder, regardless of how other permissions are set. This setting is useful for troubleshooting and testing purposes.

Security Authorizations

You can base noninherited folder security on roles or permission lists. You can specify any combination of roles and permission lists to provide folder security.

Type Select from these types:

Permission List: Enables access for users based on permission list membership. You then specify the permission lists that should have access to this folder.

Role: Enables access for users based on role membership. You then specify the roles that should have access to this folder.

Name Select the permission list or role that provides access to this folder.

Cascade Select to indicate that the permission list or role of the current row should also apply to any folder or content reference that is subordinate to this folder.

View Definition Click to access the security administration component for the permission list or role specified on the current row.

Inherited Security Authorizations

Displays authorizations that have been inherited from parent folders. The fields in this grid have the same purpose as the corresponding fields in the Security Authorizations grid. You can change inherited authorizations by modifying the Cascade setting in the Security Authorizations grid of the parent folder.

See Also

PeopleTools 8.52: Security Administration, "Setting Up Permission Lists"

PeopleTools 8.52: Security Administration, "Setting Up Roles"

Administering SmartNavigation Folders

This section provides overviews of SmartNavigation folders; rowset, tree, and dynamic hierarchy data sources; and discusses how to:

- Define SmartNavigation folders.
- Define rowset-based SmartNavigation folders.

- Define tree-based SmartNavigation folders
- Define dynamic hierarchy folders.

Important! This topic contains code samples to guide you in using the built-in PeopleCode functions that generate SmartNavigation folders and content references. The code samples exist for illustrative purposes only; never use code samples in a production environment.

Understanding SmartNavigation Folders

SmartNavigation is a form of folder navigation, but unlike standard navigation, with SmartNavigation, you define only the root folder in the portal registry. All subordinate SmartNavigation folders and content references are generated dynamically when the user begins to navigate a SmartNavigation menu. The character and appearance of SmartNavigation is also quite different from standard folder navigation because of additional folder properties that enable the portal to access and use your application data.

Note these points about SmartNavigation folders:

- When you configure SmartNavigation folders, you explicitly define the root folder only; all subordinate menu items, which reflect the data values from the data source, are generated dynamically when the user clicks SmartNavigation menu.
- The menu items in SmartNavigation menus reflect a hierarchical set of application data, which you specify as the data source for the SmartNavigation folder.
- Two data source types exist for SmartNavigation folders: rowsets and trees.
- Each SmartNavigation menu item and content reference corresponds to a data element: a row in a rowset or node in a tree.
- SmartNavigation menu hierarchy reflects the parent-child relationships that exists between rows.
- You differentiate SmartNavigation folders and content references from standard folders and content references by the images that appear beside the link. The default SmartNavigation folder and content reference images look like stylized representations of multiple entities or a single entity, respectively. You can replace these images with your own.

See [Appendix D, "Replacing SmartNavigation Images," page 717](#).

- SmartNavigation folders can appear as content references instead of folders.
- You can configure SmartNavigation to take advantage of related content services, which appear in an Actions folder in the drop-down menu hierarchy.

Important! SmartNavigation folders appear in the drop-down navigation only; they do not appear in left navigation (the Menu pagelet).

See [Chapter 5, "Administering Portals," Navigation Options, page 175](#).

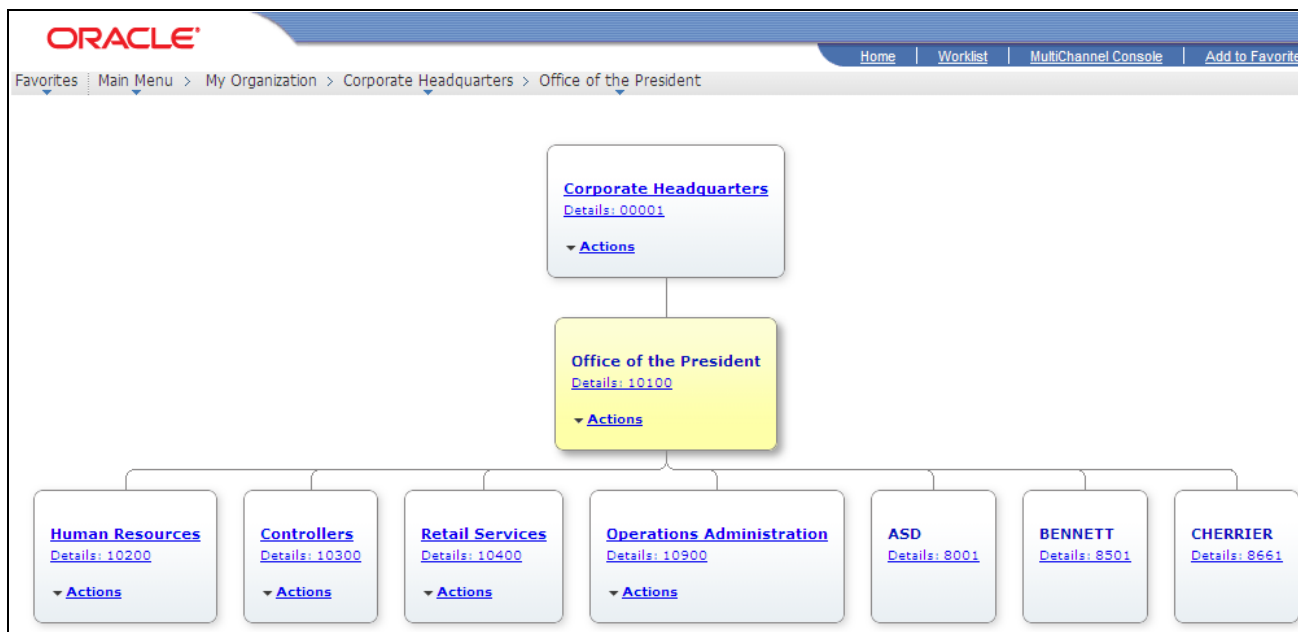
SmartNavigation Pages

When you click a SmartNavigation folder, the portal generates a navigation page; this response is comparable to standard folder navigation. The SmartNavigation page provides a characteristically different navigation experience than the standard navigation page.

Note these points about SmartNavigation pages:

- SmartNavigation pages display an organizational chart rather than folders and links.
- The character and appearance of SmartNavigation pages are controlled by PT_ABN_ORGCHART, the default chart navigation component that is standard with every PeopleTools implementation. You can clone this component and use the copy to create a different character and appearance to your SmartNavigation page or to extend and enhance the functionality of the chart.
- The chart on SmartNavigation pages reflects both the content and the structure of your data. Each chart node corresponds to a data element: a row in a rowset or a node in a tree. Chart hierarchy reflects the parent-child relationships that exists between rows.
- The nodes that appear on navigation pages vary by user based on the data permission security model that is specific to your application and your implementation.
- You associate nodes with details components so that when users click the Details link, the transaction page appears.
- As is the case with standard navigation pages, breadcrumb links appear at the top of the navigation page to indicate position in the hierarchy. Click a breadcrumb link to access a higher-level node.
- You can configure SmartNavigation pages to take advantage of related content services so that related actions are accessible from each chart node. Users do not have to navigate back through a menu; the links are available in the Actions drop-down list box on the node.

Notice the organization chart and the navigation path (breadcrumbs) on this sample SmartNavigation page, which clearly indicates the hierarchical nature of this type of navigation. :



Sample SmartNavigation page showing organization chart structure

See [Chapter 15, "Developing and Configuring Related Content Services," Understanding Related Actions Assignment, page 570.](#)

Understanding Data Sources: Rowsets

A rowset is a data structure that represents hierarchically organized records. A scroll area and all of its subordinate scroll areas are an example of a rowset. You create PeopleCode application packages or iScripts to programmatically populate rowset objects with your application data, which the portal uses when it builds SmartNavigation folders.

Rowset Characteristics

Note these points about the rowsets that you use as SmartNavigation folder data sources:

- The rowset must be a standalone rowset because the component buffers do not contain this data.
- The rowset structure must include these subrecords in this order:
 1. PTORGNODE_SBR
 2. PT_ABNNDURL_SBR
 3. PT_ABNNDCTL_SBR
 4. PTORGBOXFLD_SBR

- Because SmartNavigation pages display up to three levels of the organizational chart, the rowset should contain the relevant data required to display:
 - The requested node, which is determined by a user click or mouse over event.
 - The parent of the requested node.
 - The children of the requested node.
- The rowset should contain the data that the portal requires to build the hierarchy of folders in the breadcrumb trail. The breadcrumb trail should begin with the defined SmartNavigation folder and be followed by the rowset root node. Next, each child node should appear in the breadcrumb trail in hierarchical order until the trail ends with either the focus node or content reference.
- The rowset should contain data that is relevant to the current user only.
- For each row in the rowset, set a details component. The details component information must include these elements:
 - *MenuName*- the menu definition that contains the component.
 - *BarName*- the menu bar beneath which the component appears.
 - *ItemName* - the component name.
 - *PageName*- the page that appears to the user when the component is accessed.
 - *ActionSpecifier*- the action mode of the component. Use the single character identifier, such as U, A, and so on.
 - *SearchRecordName* - the component search record.
 - *NumberOfKeys* - the number of keys. (Use a three-digit format; for example, use 001 for one key, 010 for ten keys, and 100 for one hundred keys.)
 - *KeyFieldName* - the key field name.
 - *KeyFieldValue* - the key value.
- The details component information must be assigned to the PT_ABN_CHART_DFLD2 field value as a string that takes the following format:

Note. A comma followed by a space delimits the elements of the string.

MenuName.MenuName, BarName.BarName, ItemName.ItemName, Page.PageName, Action.ActionSpecifier, Record.SearchRecordName, KeyFieldNumber.NumberOfKeys, FieldName.KeyFieldName, FieldValue.'KeyFieldValue', FieldName.KeyFieldName, FieldValue.'KeyFieldValue'

Important! If you deviate from the format then SmartNavigation cannot parse the details component information string.

Application Package PeopleCode Sample: Rowset

This item in this list step through a sample PeopleCode program for use as the data source of a rowset-based SmartNavigation folder. The next topic explains the PeopleCode that you see in this list.

See [Chapter 5, "Administering Portals," Application Package PeopleCode Sample Explanation: Rowset, page 121.](#)

Important! This code sample exists for illustrative purposes only. Use this sample to guide your programming; never use code samples in a production environment.

1. Declare the method and variables.

```
method QE_ABNRS_M

    Local Request &req = %Request;
    Local Rowset &rs_DataSource, &rs_DatabaseData;
    Local Rowset &rs_ChartRowset, &rs_DbReqNode;

    rem create related actions rowset for SmartNavigation;
    Local Rowset &rs_RelatedActions;

    rem added for app pkg;
    Local array of array of string &reqParams;
    Local string &dataKey;
    Local string &szDtlCompData;
    Local integer &i;
```

2. Get the SmartNavigation request parameters and specify the initial node of the rowset.

```
    rem Get request parameters and set initial node
    rem Binu's Org;
    &reqParams = GetABNReqParameters("BINU01");
```

3. Create the stand alone rowset data source.

```
    rem begin: create the data source rowset;

    rem retrieve complete data source;
    rem &rs_DatabaseData = CreateRowset(Record.MY_ORG_NODES);
    rem &dataKey = "PT_ORG_001";
    rem &rs_DatabaseData.Fill("Where MY_ORG_KYFLD=:1", &dataKey);
    rem &rs_DataSource = CreateRowset(Record.MY_ORG_NODE_DR);
    rem &rs_DatabaseData.CopyTo(&rs_DataSource, Record.MY_ORG_NODES, =>
Record.MY_ORG_NODE_DR);

    rem retrieve subset of data source
    &rs_DbReqNode = CreateRowset(Record.MY_ORG_NODES);
    &rs_DatabaseData = CreateRowset(Record.MY_ORG_NODES);
    &dataKey = "PT_ORG_001";
    &rs_DbReqNode.Fill("Where MY_ORG_KYFLD=:1 AND PTCHART_NODE=:2", =>
&dataKey, GetABNNode(&reqParams));
    &rs_DatabaseData.Fill("Where MY_ORG_KYFLD=:1 AND (PTCHART_NODE=:2 =>
OR PTPARENT_CHART_ND=:2 OR PTCHART_NODE=:3)", &dataKey, =>
GetABNNode(&reqParams), =>
&rs_DbReqNode(1).MY_ORG_NODES.PTPARENT_CHART_ND.Value);
    &rs_DataSource = CreateRowset(Record.MY_ORG_NODE_DR);
    &rs_DatabaseData.CopyTo(&rs_DataSource, Record.MY_ORG_NODES, =>
Record.MY_ORG_NODE_DR);
```


4. Set the details component

```

rem set details component for all nodes;
&szDtlCompData = "MenuName.MY_MENU, BarName.USE, ItemName.EMPLOYEE, =>
Page.EMPLOYEE, Action.U, Record.EMPLOYEE, KeyFieldNumber.001, =>
FieldName.EMPLID, FieldValue.";

For &i = 1 To &rs_DatabaseData.ActiveRowCount

    &rs_DataSource(&i).QE_ABNRS_DS.PT_ABN_CHART_DFLD2.Value = =>
&szDtlCompData | &rs_DataSource(&i).QE_ABNRS_DS.PTCHART_NODE.Value;

End-For;

rem end: create the data source;

```

5. Get the chart rowset.

```

rem get the chart rowset returns the chart rowset;
&rs_ChartRowset = GetABNChartRowSet();

```

6. Get the SmartNavigation related actions rowset.

```

rem gets related action rowset from the chart component;
&rs_RelatedActions = GetABNRelActnRowSet();

```

7. Generate the HTML for the SmartNavigation menu.

```

rem generate the html fragment for the menu;
rem GenHTMLMenu(&rs_DataSource, =>
GetABNNode(&reqParams), =>
GetABNInitialNode(&reqParams));

```

8. Load the chart rowset.

```

rem load the chart rowset from the data source rowset;
LoadABN(&rs_DataSource, &rs_ChartRowset, =>
&rs_RelatedActions, GetABNNode(&reqParams), =>
GetABNInitialNode(&reqParams));

end-method;

```

Application Package PeopleCode Sample Explanation: Rowset

Each item in this list explains the corresponding list item in the application package PeopleCode sample program.

1. Declare the function and variables.
2. Get the SmartNavigation request parameters and specify the initial node of the rowset.
 - Retrieves a two dimensional array, or array of arrays, of name/value pairs.
 - The name/value pairs are request parameters needed by SmartNavigation functionality.
 - The input parameter is the identifier of the initial node .

3. Create the stand alone rowset data source.

- The rowset structure is based on the record MY_ORG_NODES, which includes the necessary subrecords.
- This table describes the variables that are used to store data in the functions:

<i>Parameter</i>	<i>Description</i>
&rs_dbReqNode	Rowset that contains the data for the requested node.
&rs_DatabaseData	Rowset that contains the data for the requested node, parent node, child nodes.
&rs_dataKey	Variable that contains the high level key.
&rs_DataSource	Rowset that contains the entire data set that is used by SmartNavigation chart.

- Retrieve the data for the requested node (&rs_DbReqNode.Fill).
- Retrieve the data for the parent and child nodes of the requested node (&rs_DatabaseData.Fill).
- Copy the data from all nodes to the chart rowset (&rs_DatabaseData.CopyTo)

4. Set the details component.

- Set the name of the details component.
- Loop through all of the nodes and insert the component string into the PT_ABN_CHART_DFLD2 field.

5. Get the chart rowset.

- This function returns Null if triggered by a mouse over event; returns the rowset if triggered by an on click event at a menu element or the first link of a chart node.
- Upon retrieval of the related actions rowset, the chart rowset is flushed. So, this function be called one time only in the iScript program.
- Retrieval of the related action rowset allows developers to access and modify the chart rowset beyond the PeopleTools' functionality.

6. Get the SmartNavigation related actions rowset.

- This function returns Null if triggered by an on click event of the menu item (folder) label; returns the rowset if triggered by an on click event of the menu item (folder) image or the first link of a chart node.
- Upon retrieval of the related actions rowset, the chart rowset is flushed. So, this function be called one time only in the iScript program.
- Retrieval of the related action rowset allows developers to access and modify the chart rowset beyond the PeopleTools' functionality.

7. Generate the HTML for the SmartNavigation menu.

8. Load the chart rowset from the data source rowset.

- This table describes the parameters for the function:

Parameter	Description
&rs_DataSource	Data source rowset retrieved in item 4 in this list.
&rs_ChartRowset	Chart rowset retrieved in item 6 in this list.
&rs_RelatedActions	Related actions rowset retrieved in item 7 in this list.
&reqParams	The requested parameters retrieved in item 2 in this list.
GetABNNode(&reqParams)	The requested node based on the user on click event of a menu item (folder) image or the on click event of the first link of a chart node.
GetABNInitialNode(&reqParams)	The initial node set in item 2 in this list

- End the method.

See Also

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNChartRowSet

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNInitialNode

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNNode

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNRelActnRowSet

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNReqParameters

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GenHTMLMenu

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," LoadABN

Understanding Data Sources: Trees

Trees are hierarchically-organized structures that represent a group of summarization rules for a particular database field. For example, a tree can specify how your manufacturing locations should be summarized for reporting purposes. Or a tree can show the reporting relationships within an organization by specifying how the individual department should be summarized into territories, territories into regions, and regions into countries. Similarly, a tree can categorize items in a catalog.

Tree Characteristics

Note these points about trees that you use as SmartNavigation folder data sources:

- Trees must be valid and active. Inactive, draft, and unaudited trees render unexpected results.
- SmartNavigation passes the values of several tree-specific fields to the application via URL. Certain characters are inappropriate for use in a URL and should be avoided. When you set up trees, do not use any of the following characters in the Tree Name, SetID, Set Control Value, and Tree Branch fields: pound (#), percent (%), dollar (\$), ampersand (&), plus (+), comma (,), forward slash/virgule (/), colon (:), semi-colon (;), equals (=), question mark (?), at symbol (@), space (), quotation marks("), less than symbol (<), greater than symbol (>), left curly brace ({), right curly brace (}), vertical bar/pipe (|), backslash (\), caret (^), tilde (~), left square bracket ([), right square bracket (]), and grave accent (`).
- The root node of the tree appears as the initial node of the SmartNavigation page unless you use PeopleCode to specify another initial node.

iScript PeopleCode Sample: Tree

This items in this list step through a sample PeopleCode program for use as the data source of a tree-based SmartNavigation folder. The next topic explains the PeopleCode that you see in this list.

See [Chapter 5, "Administering Portals," iScript PeopleCode Sample Explanation: Tree, page 126.](#)

Important! This code sample exists for illustrative purposes only. Use this sample to guide your programming; never use code samples in a production environment.

1. Declare the function and variables.

```
Function IScript_PT_ABN()

    Local Request &req = %Request;
    Local string &szLI;
    Local ApiObject &Session = %Session;
    Local ApiObject &MyTree, &MyNode, &ChildNode, &ChildLeaf;
    Local Rowset &rs, &rars;
```

2. Retrieve the chart rowset.

```
    rem get the chart rowset;
    &rs = GetABNChartRowSet();
```

3. Retrieve the related actions rowset from the chart component.

```
    rem retrieve the related action rowset from the chart component;
    &rars = GetABNRelActnRowSet();
```

4. Get the SmartNavigation request parameters and specify the initial node of the tree.

```
    rem get the request parameters and specify the initial node or not;
    &reqParams = GetABNReqParameters("00001");
```

5. Create an empty tree from session object and open the tree based on the request parameters.

```
/* get an empty tree from the session object */
&MyTree = &Session.GetTree();

/* open the tree based on the request parameters */
&TreeReturn = &MyTree.Open(GetABNTreeSetid(&reqParams), ' '~>
GetABNTreeUserKey(&reqParams), =>
GetABNTreeName(&reqParams), =>
GetABNTreeEffdt(&reqParams), "", True);
```

6. Retrieve the requested tree node.

```
/* get the requested tree node */
&MyNode = &MyTree.FindNode(GetABNNode(&reqParams), "");
```

7. Load the requested node into the chart rowset.

```
rem load the requested node into the chart and track the starting node;
&MyNode.LoadABNChart(&rs, &rars, True, =>
GetABNInitialNode(&reqParams));
```

8. Generate the related actions of the requested tree node.

```
rem generate the requested node's related actions;
&szLI = &szLI | &MyNode.GenRelatedActions();
```

9. Generate the SmartNavigation folder elements to display in the menu (based on some optional criteria).

```
/* begin-filter
based on some criteria, display only the child nodes and leaves of the
requested tree node that meet the criteria */

&nNodeOrder = &MyNode.allChildCount;
If &MyNode.HasChildNodes Then
/* get the first child node */
&ChildNode = &MyNode.FirstChildNode;

/* generate the <li> tag that the portal consumes for this node */
&szLI = &szLI | &ChildNode.GenABNMenuElement=>
(GetABNInitialNode(&reqParams));
```

10. Load the child node into the chart rowset buffer.

```

    &ChildNode.LoadABNChart(&rs, &rars, False, =>
GetABNInitialNode(&reqParams));
End-If;
If &MyNode.HasChildLeaves Then
    If &MyNode.ChildLeafCount >= 3 Then
        &MaxCount = 3;
    Else
        &MaxCount = &MyNode.ChildLeafCount;
    End-If;

    For &i = 1 To &MaxCount
        If &i = 1 Then
            /* get the first child leaf */
            &ChildLeaf = &MyNode.FirstChildLeaf;
        Else
            /* get the next child leaf */
            &ChildLeaf = &ChildLeaf.NextSib;
        End-If;
    
```

11. Generate the SmartNavigation content reference elements to display in the menu.

```

    /* generate the <li> tag that is consumed by the portal for this leaf */
    &szLI = &szLI | &ChildLeaf.GenABNMenuElement();

```

12. Load a child node of the requested tree node and its related actions into the chart buffer.

```

    rem load the child leaf of requested node into the chart;
    &ChildLeaf.LoadABNChart(&rs, &rars);
End-For;

End-If;

/* end-filter */

```

13. Generate the entire HTML that the portal consumes to create the SmartNavigation menu.

```

    /* generate the html fragment required by the portal to build menu structure */
    GenHTMLMenu(&szLI);

```

14. Close the tree and end the function.

```

    &MyTree.Close();

End-Function;

```

iScript PeopleCode Sample Explanation: Tree

Each item in this list explains the corresponding list item in the sample Tree PeopleCode program.

See [Chapter 5, "Administering Portals," iScript PeopleCode Sample: Tree, page 124.](#)

1. Declare Function and variables.

2. Retrieve the chart rowset.

- This function returns Null if triggered by an on click event of the menu item label; returns the rowset if triggered by an on click event of the menu item image or the first link of a chart node.
- The rowset must be retrieved because the tree API doesn't have access, or know of the existence of the panel processor.
- The rowset must be passed to the tree api so that the tree API can properly load the chart rowset.
- Upon retrieval of the chart rowset, the chart rowset is flushed. So, function should be called one time only in the iScript program.
- Retrieval of the chart rowset enables developers to access and modify the chart rowset beyond the PeopleTools' functionality.
- If you use a chart component other than the PeopleTools-delivered chart component, the component buffer structure must match the PeopleTools-delivered template so that this function can find the correct rowset. The chart rowset must be at level 1 and the first scroll at level 1.

3. Retrieve the related actions rowset from the chart component buffer.

- This function returns Null if triggered by a mouse over event; returns the rowset if triggered by an on click event at a menu element or the first link of a chart node.
- Upon retrieval of the related actions rowset, the chart rowset is flushed. So, this function be called one time only in the iScript program.
- Retrieval of the related action rowset allows developers to access and modify the chart rowset beyond the PeopleTools' functionality.

4. Get the SmartNavigation request parameters and specify the initial node of the tree.

- Retrieves a two dimensional array, or array of arrays, of name/value pairs.
- The name/value pairs are request parameters needed by SmartNavigation functionality.
- The input parameter is the initial node identifier.
- If no input parameter is passed, the initial node is the root node of the tree.

5. Create an empty tree from session object and open the tree based on the request parameters.

- Create an empty tree session object.
- Open the tree using the request parameters. This table describes the tree parameters:

<i>Tree Parameter</i>	<i>Description</i>
GetABNTreeSetid(&reqParams)	Retrieves the SetId key value from the input parameter
GetABNTreeUserKey(&reqParams)	Retrieves the tree user key value from the input parameter
GetABNTreeUserKey(&reqParams)	Retrieves the SetId key value from the input parameter
GetABNTreeEffdt(&reqParams)	Retrieves the SetId key value from the input parameter

Note. For the preceding table, &reqParams is the output of item 4 in this list.

6. Retrieve the requested tree node.

- The requested node is the menu element upon which the user is focused during the mouse over or click event.
- The SmartNavigation function retrieves the node identifier from the input parameter, which is the output of item two in this list.

7. Load the requested node into the chart rowset.

- Load the requested node from item six in this list into the chart rowset buffer.
- If necessary, load the parent node into the chart buffer.
- Determine the initial node by retrieving its identifier from request parameters set in item three in this list.
- This table describes the function parameters:

<i>Parameter</i>	<i>Description</i>
&rs	Chart component rowset retrieved from item 2 in this list.
&rars	Related action rowset retrieved by item 3 in this list.
Boolean	Boolean which, if true, directs the function to load the node's parent so that the correct number of chart levels are displayed.
GetABNInitialNode(&reqParams)	Retrieves the identifier tree initial node from the input parameter. The value of the input parameter is the output of item three in this list. You must keep track of the starting node so that PeopleTools can navigate correctly.

8. Generate the related actions elements to display in the menu.

- Generate the element for related actions. In the SmartNavigation menu, these elements appear as the folder and subordinate actions for the requested tree node.
- The output of the function is concatenated to the current string of elements, &szLI, to create a list of elements.
- The list of elements is used to generate an HTML fragment that the portal consumes.

9. Generate the SmartNavigation folder elements to display in the menu (based on some optional criteria).

- Filter the nodes that you want to display and order them
- Generate an element for a tree node. In the SmartNavigation menu, this element appears as a folder.
- The output of the function is concatenated to the current string of elements, &szLI, to create a list of elements.
- The list of elements is used to generate an HTML fragment that the portal consumes.
- This table describes the function parameters:

Parameter	Description
&rs	Chart component rowset retrieved from item 2 in this list.
&nNodeOrder	The display order of the leaf in the chart.
&szLI	The list of items that appear in the menu.
GetABNInitialNode(&reqParams)	Retrieves the identifier tree initial node from the input parameter. The value of the input parameter is the output of item three in this list.

10. Load the child node into the chart rowset buffer.

- Load the child node from item 2 in this list into the chart buffer.
- Load the related actions rowset from item 3 in this list into the chart buffer.
- Determine the initial node by retrieving its identifier from request parameters set in item 4 in this list.
- This table describes the function parameters:

Parameter	Description
&rs	Chart component rowset retrieved from item 2 in this list.
&rars	Related actions rowset retrieved from item 3 in this list.
Boolean	Boolean which, if false, directs the function to load the calling node into the chart buffer. Because the requested node is loaded in the buffer already, select false.
GetABNInitialNode(&reqParams)	Retrieves the identifier of the initial node from the input parameter. The value of the input parameter is the output of item 3 in this list.

11. Generate the SmartNavigation content reference elements to display in the menu.

- Generate an element for a tree leaf. In the SmartNavigation menu, this element appears as a content reference.
- The output of the function is concatenated to the current string of elements, &szLI, to create a list of elements.
- The list of elements is used to generate an HTML fragment that the portal consumes.

12. Load a child leaf into the chart rowset buffer.

- Load the child leaf from item 2 in this list into the chart buffer.
- Load the related actions of the child leaf from item 3 in this list into the chart buffer.

13. Generate the HTML that creates the SmartNavigation menu.

Generate the HTML fragment using the list of elements from items 9, 10, and 11 in this list.

14. Close the tree and end the function.

See Also

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions,"
GetABNChartRowSet

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNTreeEffdt

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNTreeName

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNTreeSetid

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNTreeUserKey

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNInitialNode

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNNode

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions,"
GetABNRelActnRowSet

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions,"
GetABNReqParameters

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GenHTMLMenu

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," LoadABN

Understanding Data Sources: Dynamic Hierarchy

A dynamic hierarchy is a data source that enables you to group SmartNavigation folders within a folder. You define the dynamic hierarchy by creating an application package PeopleCode program that you reference from the defined dynamic hierarchy folder. All of the properties of the subordinate SmartNavigation folders, such as data source, must be defined programmatically in an application package PeopleCode program. Then, the PeopleCode generates the enclosing SmartNavigation folder elements of each tree or rowset so that groups of SmartNavigation folders appear under one menu.

With other SmartNavigation folders, permission lists and roles determine whether the SmartNavigation folder and the underlying details component are visible and accessible to users. Dynamic hierarchy folders, through the use of application package PeopleCode, enable you to add another element to the security mechanism for the programmatically defined SmartNavigation folders. For example, you could use a dynamic hierarchy folder to display project data for projects that are assigned to the current user only.

Dynamic Hierarchy Characteristics

Note these points about the dynamic hierarchy data source:

- The dynamic hierarchy folder must specify an application package PeopleCode program.
- The application package PeopleCode must generate a string of elements, each of which a programmatically-defined SmartNavigation folder that will appear as a subordinate menu item in the dynamic hierarchy folder.
- These subordinate menu items are logical SmartNavigation folders; they are not defined in the portal registry.
- The application package PeopleCode then generates an HTML fragment that the portal consumes; this HTML creates the SmartNavigation menu structures..

Application Package PeopleCode Sample: Dynamic Hierarchy

This items in this list step through a sample PeopleCode program for use as the data source of a dynamic hierarchy folder. The next topic explains the PeopleCode that you see in this list.

See [Chapter 5, "Administering Portals," Application Package PeopleCode Sample Explanation: Dynamic Hierarchy, page 133.](#)

Important! This code sample exists for illustrative purposes only. Use this sample to guide your programming; never use code samples in a production environment.

1. Declare the method within the appropriate class.

```
class TEST_ABN_DYN
  method GenerateDynFlyout( );
end-class;
```

```
method GenerateDynFlyout
```

2. Declare the variables.

```

Local string &flderList;
Local string &ds_t, &cref_t, &id_t, &label_t, &portal, &node, &chart, =>
&pcode_t, &tree_name, &tree_setid, &tree_userkey, &tree_effdt, &tree_branch;
Local string &ds_r, &cref_r, &id_r, &label_r, &pcode_r;

```

3. Provide the values for the tree-based SmartNavigation folder as variable assignments.

```

rem tree information;
&ds_t = "t";
&cref_t = "f";
&label_t = "New Dynamic ABN Tree";
&chart = "PT_ABN_CHART.PT_ABN_ORGCHART.GBL";
&pcode_t = "QE_ABNTREE.qe_abntree.QE_ABN_TREE_AP";
&tree_name = "QE_PERS_DATA";
&tree_setid = "QEDM1";
&tree_userkey = "";
&tree_effdt = "1997/05/05";
&tree_branch = "";

```

4. Create the list of tree-based SmartNavigation folders.

```

rem create the <li> element for the tree-based SmartNav folder;
&flderList = GenDynABNElement(&ds_t, &cref_t, &fldr_id, =>
&label_t, &chart, &pcode_t, &tree_name, &tree_setid, &tree_userkey, &tree_effdt, &tree

```

5. Provide the values for the rowset-based SmartNavigation folder.

```

rem rowset information;
&ds_r = "r";
&cref_r = "f";
&label_r = "Dynamic ABN Rowset";
&chart = "PT_ABN_CHART.PT_ABN_ORGCHART.GBL";
&pcode_r = "QE_ABN_RS_APKG.QE_ABN_RS_1:QE_ABNRS_C1.QE_ABNRS_M";

```

6. Create the list of rowset-based SmartNavigation folders.

```

rem create the <li> element for the smartnav rowset folder and append;
&flderList = &flderList | GenDynABNElement(&ds_r, &cref_r, &label_r, =>
&chart, &pcode_r);

```

7. Generate the HTML that the portal uses to construct the menu navigation.

```

rem this function already exists;
GenHTMLMenu(&flderList);

```

```

end-method;

```

Application Package PeopleCode Sample Explanation: Dynamic Hierarchy

Each item in this list explains the corresponding list item in the Application Package PeopleCode Sample: Dynamic Hierarchy:

See [Chapter 5, "Administering Portals," Application Package PeopleCode Sample: Dynamic Hierarchy, page 132.](#)

1. Declare the method in the appropriate class.
2. Declare the variables.
 - Declare the necessary local variables.
 - All variables in the sample program are strings.
3. Provide the values for the tree-based SmartNavigation folder as variable assignments.

This table describes the tree parameters; you use these parameters in the next step.

Parameter	Description
<code>&ds_t = "data source type"</code>	Data source type: t, tree; r, rowset
<code>&ceref_t = "flag value"</code>	Display as content reference flag: true, content reference; false, folder
<code>&fldr_id = "folder ID"</code>	Folder ID
<code>&label_t = "menu folder label"</code>	Folder label
<code>&chart = "chart component"</code>	Chart component
<code>&pcode_t = "PeopleCode location"</code>	PeopleCode ID: (<i>package:path:method</i>) or (<i>weblib_record.event.function_name</i>)
<code>&tree_name = "tree name"</code>	Tree name
<code>&tree_setid = "setID value"</code>	Tree setID
<code>&tree_userkey = "key value"</code>	Tree user key
<code>&tree_effdt = "EFFDT value"</code>	Tree effective date
<code>&tree_branch = "branch value"</code>	Tree branch

4. Create the list of tree-based SmartNavigation folders.
 - This function creates the tree-based SmartNavigation folders based on the parameter values you provide.
 - The parameters are the same as the Application Navigation properties that you supply on the Folder Administration page to create a SmartNavigation folder.

See [Chapter 5, "Administering Portals," Defining SmartNavigation Folders, page 138.](#)

- The output is a string composed of `` elements for the tree-based SmartNavigation folders.

5. Provide the values for the rowset-based SmartNavigation folder as variable assignments.

This table describes the rowset parameters; you use these parameters in the next step.

Parameter	Description
<code>&ds_r = "data source type"</code>	Data source type: t, tree; r, rowset
<code>&cref_r = "flag value"</code>	Display as content reference flag: true, content reference; false, folder
<code>&fldr_id = " folder ID"</code>	Folder ID
<code>&label_r = "menu folder label"</code>	Folder label
<code>&chart = "chart component"</code>	Chart component
<code>&pcode_r = "PeopleCode location"</code>	PeopleCode ID: (package:path:method) or (weblib_record.event.function_name)

6. Create the list of rowset-based SmartNavigation folders.

- This function creates the rowset-based SmartNavigation folders based on the parameter values you provide.
- The parameters are the same as the Application Navigation properties that you supply on the Folder Administration page to create a SmartNavigation folder.

See [Chapter 5, "Administering Portals," Defining SmartNavigation Folders, page 138.](#)

- The output is a string composed of elements of the tree-based SmartNavigation elements from item 4 in this list concatenated with the string composed of elements for the rowset-based SmartNavigation folders.
7. Generate the HTML that the portal uses to construct the menu navigation.
- The dynamic hierarchy folder becomes the top level menu and each programmatically-defined SmartNavigation folder appears as a subordinate menu item in the list of folders.
 - The input parameter is the list of elements.
 - The output is an HTML fragment.
8. Load the chart rowset.
- Load the chart rowset from the data source rowset.
 - Load the related actions rowset.
 - Set the initial chart node.
 - End the method.

See Also

Appendix D, "Replacing SmartNavigation Images," page 717

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GenDynABNElement

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GenHTMLMenu

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions,"
GetABNChartRowSet

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNTreeEffdt

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNTreeName

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNTreeSetid

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNTreeUserKey

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNInitialNode

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNNode

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions,"
GetABNRelActnRowSet

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions,"
GetABNReqParameters

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," LoadABN

Common Elements Used In This Section

This section discusses the fields that identify the different data sources that you use with SmartNavigation folders.

Application Navigation

This table describes the fields that appear in the Application Navigation group box:

Data Source	Specify the type of data source that the system uses to generate the SmartNavigation folder and page elements. Select <i>Dynamic Hierarchy</i> , <i>Rowset</i> , or <i>Tree</i> . When you select a data source, the system enables and selects Default Chart Navigation Pagecheck box, and displays the PeopleCode Type field.
--------------------	---

Display as Content Reference	<p>Select to replace the SmartNavigation folder with a content reference link. If you select this check box, no subordinate menu items appear when the user clicks the link. The system immediately opens the SmartNavigation page to the initial node in the target area.</p> <p>This field is enabled for rowset and tree-based folders. This field is hidden for dynamic hierarchy folders.</p> <hr/> <p>Note. This property applies to the root-level SmartNavigation folder only. System-generated drop-down navigation subfolders are not affected by this setting.</p> <hr/>
Node Name	Select the node where the content resides.
PeopleCode Type	<p>This field appears when you specify a data source.</p> <p>Select <i>Application Package</i> to enable the Application Package Parameters group box.</p> <p>Select <i>iScript</i> to enable the iScript Parameters group box</p> <hr/> <p>Note. Use the group box that corresponds to the PeopleCode type to identify the PeopleCode program.</p> <hr/>
Style Sheet	Enter the style sheet definition that replaces the standard folder and content reference icons in the SmartNavigation menus.

Application Package Parameters

The Application Package Parameters group box appears when any of these conditions are met:

- *Dynamic Hierarchy* is selected as a data source.
- *Rowset* is selected as the data source and *Application Package* is selected as the PeopleCode type.
- *Tree* is selected as the data source and *Application Package* is selected as the PeopleCode type.

This table describes the fields that appear in the Application Package Parameters group box

Application Package	Enter the PeopleCode application class name where the application class method is defined.
Application Package Path	Enter the names—delimited by colons—of each subpackage in the hierarchy that drills down to the class that contains the method. Beginning and ending colons are <i>not</i> required. For example, the path could be <i>sub_package:sub_package:app_class</i> or simply <i>app_class</i> if no sub application packages exist.
Application Package Method	Enter the name of the specific method that builds the rowset that populates the navigation page in the target area.

iScript Parameters

The iScript Parameters group box appears when any one of these conditions are met:

- *Tree* as a data source and *iScript* as the PeopleCode type.
- *Rowset* as the data source and *iScript* as the PeopleCode type.

This table describes the fields appear in the iScript Parameters group box:

Record (Table) Name	Specify the record definition that contains the iScript that defines the rowset.
Field Name	Specify the record field that contains the iScript that defines the rowset.
PeopleCode Event Name	Specify the event that triggers the iScript.
PeopleCode Function Name	Specify the function in the iScript that contains the business logic.

Tree Properties

The Tree Properties group box appears when you select *Tree* as the data source.

This table describes the fields that appear in the Tree Properties group box:

Tree Name	Select the name of the tree that the portal is to use to render the menu.
SetID	Enter the SetID for the tree.
Set Control Value	Enter the set control value for the tree.
Effective Date	(Optional) Enter the effective date for the tree.
Tree Branch	(Optional) Enter the tree branch for the tree.

Defining SmartNavigation Folders

Access the Folder Administration page (Select PeopleTools, Portal, Structure and Content. Click the Add Folder link).

Folder Navigation	
<input type="checkbox"/> Is Folder Navigation Disabled	<input checked="" type="checkbox"/> Default Chart Navigation Page
Folder Navigation Object Name:	<input type="text"/>
Application Navigation	
Data Source:	Tree <input type="button" value="v"/>
*Node Name:	EMPLOYEE <input type="button" value="m"/>
<input checked="" type="checkbox"/> Display as Content Reference	
Tree Properties	
*Tree Name:	QE_PERS_DATA
SetID:	QEDM1
Set Control Value:	<input type="text"/>
Effective Date:	05/05/1997 <input type="button" value="31"/>
Tree Branch:	<input type="text"/>
PeopleCode Type:	<input type="button" value="v"/>
Style Sheet:	<input type="text"/>

Folder Administration page showing SmartNavigation folder options for the tree data source

To configure SmartNavigation folders:

1. Enter a name, label and other identifying information.
2. In the Application Navigation group box, select a data source and define the data source parameters.
 See [Chapter 5, "Administering Portals," Defining Dynamic Hierarchy Folders, page 140.](#)
 See [Chapter 5, "Administering Portals," Defining Rowset-Based SmartNavigation Folders, page 141.](#)
 See [Chapter 5, "Administering Portals," Defining Tree-Based SmartNavigation Folders, page 142.](#)
3. (Optional) To use a nondefault chart navigation page, deselect the Default Chart Navigation Page check box and enter a value in the Folder Navigation Object Name field.

Important! If the node is a *remote* content site, then you must clear the Default Chart Navigation Page field and then enter a *remote* content reference in the Folder Navigation Object field. Otherwise, the portal cannot generate the correct URL for the SmartNavigation folder. You also must enable single signon.

4. Set folder permissions.
 See [Chapter 5, "Administering Portals," Setting Folder Permissions, page 114.](#)
5. Save the folder.

See Also

PeopleTools 8.52: Security Administration, "Implementing Single Signon"

Defining Dynamic Hierarchy Folders

Access the Folder Administration page (Select PeopleTools, Portal, Structure and Content Click the Add Folder link).

The screenshot shows the 'Folder Administration' page with the following sections and fields:

- Folder Navigation**: Contains a checkbox labeled 'Is Folder Navigation Disabled'.
- Application Navigation**: Contains a 'Data Source' dropdown menu set to 'Dynamic Hierarchy', a '*Node Name' text input field with a search icon, and a 'Style Sheet' text input field.
- Application Package Parameters**: Contains three text input fields labeled 'Application Package', 'Application Package Path', and 'Application Package Method'.

Folder Administration page showing fields necessary to create dynamic hierarchy folders

To define dynamic hierarchy folders:

1. Enter a name, label, and any other identifying information for the folder.
2. Select *Dynamic Hierarchy* as the data source.
3. Specify the content site node.
4. Enter the parameters of the application package PeopleCode program that serves as the data source for this folder.

See [Chapter 5, "Administering Portals," Understanding Data Sources: Dynamic Hierarchy, page 132.](#)

5. (Optional) Enter the style sheet that provides alternate images for SmartNavigation menu items.

See [Appendix D, "Replacing SmartNavigation Images," page 717.](#)

6. (Optional) Specify any necessary folder attributes.
7. Set folder permissions.

See [Chapter 5, "Administering Portals," Setting Folder Permissions, page 114.](#)

8. Save the folder definition.

Defining Rowset-Based SmartNavigation Folders

Access the Folder Administration page (Select PeopleTools, Portal, Structure and Content Click the Add Folder link).

The screenshot shows the 'Folder Administration' page. It has two main sections: 'Folder Navigation' and 'Application Navigation'. In the 'Folder Navigation' section, there is a checkbox 'Is Folder Navigation Disabled' (unchecked) and a checkbox 'Default Chart Navigation Page' (checked). Below these is a text field 'Folder Navigation Object Name:' with a search icon. The 'Application Navigation' section contains a 'Data Source:' dropdown menu set to 'Rowset', a checkbox 'Display as Content Reference' (unchecked), a '*Node Name:' text field with a search icon, a 'PeopleCode Type:' dropdown menu, and a 'Style Sheet:' text field.

Folder Administration page showing fields necessary to create rowset-based SmartNavigation folders

To define rowset-based SmartNavigation folders:

1. Enter a name, label, and any other identifying information for the folder.
2. Select *Rowset* as the data source.
3. Specify the content site node.
4. In the PeopleCode Type field, select which type of PeopleCode program defines the data source. Select *Application Package* or *iScript*.
5. Enter either the iScript or application package parameters that define the rowset.

See [Chapter 5, "Administering Portals," iScript Parameters, page 138](#).

See [Chapter 5, "Administering Portals," Application Package Parameters, page 137](#).

Note. If you use an iScript, you must grant permissions to the web library and iScript for users who will access the SmartNavigation folder.

See *PeopleTools 8.52: Security Administration*, "Setting Up Permission Lists," Setting Web Library Permissions.

6. (Optional) Select *Display as Content Reference* so that the SmartNavigation folder appears and functions as a content reference instead of a folder.
7. (Optional) Deselect the *Default Chart Navigation Page* check box and select a custom folder navigation object.

Complete this step only if you want to replace the SmartNavigation chart page with a custom navigation page.

8. (Optional) Specify any necessary folder attributes.

9. Set folder permissions.

See [Chapter 5, "Administering Portals," Setting Folder Permissions, page 114.](#)

10. Save the folder definition.

Defining Tree-Based SmartNavigation Folders

Access the Folder Administration page (Select PeopleTools, Portal, Structure and Content Click the Add Folder link).

The screenshot shows the 'Folder Navigation' and 'Application Navigation' sections of the Folder Administration page. The 'Folder Navigation' section includes a checkbox for 'Is Folder Navigation Disabled' (unchecked) and a checked checkbox for 'Default Chart Navigation Page'. Below these is a text field for 'Folder Navigation Object Name' with a search icon. The 'Application Navigation' section includes a 'Data Source' dropdown menu set to 'Tree', a checked checkbox for 'Display as Content Reference', and a '*Node Name' text field containing 'EMPLOYEE' with a search icon. Below this is a 'Tree Properties' section with fields for '*Tree Name' (containing 'QE_PERS_DATA'), 'SetID' (containing 'QEDM1'), 'Set Control Value' (empty), 'Effective Date' (containing '05/05/1997' with a calendar icon), and 'Tree Branch' (empty). At the bottom, there is a 'PeopleCode Type' dropdown menu and a 'Style Sheet' text field.

Folder Administration page showing fields necessary to create tree-based SmartNavigation folders

1. Enter a name, label, and any other identifying information for the folder.
2. Select *Tree* as the data source.
3. Specify the content site node.
4. In the PeopleCode Type field, select which type of PeopleCode program defines the data source. Select *Application Package* or *iScript*.
5. Enter either the iScript or application package parameters that define the rowset.

See [Chapter 5, "Administering Portals," iScript Parameters, page 138.](#)

See [Chapter 5, "Administering Portals," Application Package Parameters, page 137.](#)

Note. If you use an iScript, you must grant permissions to the web library and iScript for users who will access the SmartNavigation folder.

See *PeopleTools 8.52: Security Administration*, "Setting Up Permission Lists," Setting Web Library Permissions.

6. (Optional) Select **Display as Content Reference** so that the SmartNavigation folder appears and functions as a content reference instead of a folder.
7. (Optional) Deselect the **Default Chart Navigation Page** check box and select a custom folder navigation object.

Complete this step only if you want to replace the SmartNavigation chart page with a custom navigation page.

8. (Optional) Specify any necessary folder attributes.
9. Set folder permissions.

See [Chapter 5, "Administering Portals," Setting Folder Permissions, page 114.](#)

10. Save the folder definition.

Administering Content References

This section discusses how to:

- Define content references.
- Create related links.
- Set content reference permissions.
- Review content reference permissions.
- Test content references.
- Define content reference links.
- Set content reference link security.

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. Scroll to the Content References list and click Edit. Alternatively, click Add Content Reference.	Create new content references or edit existing content references.
Content Reference Security	PORTAL_CREF_SEC	On the Content Ref Administration page, select the Security tab.	Set content reference security.

Page Name	Definition Name	Navigation	Usage
User ID Queries	USER_QUERY	PeopleTools, Security, User Profiles, User Profiles. Select the User ID Queries tab.	Review content reference security by user.
Permission List Queries	PLIST_QUERIES	PeopleTools, Security, Permissions & Roles, Permission Lists. Select the Permission List Queries tab.	Review content reference security by permission list.
Role Queries	ROLE_QUERY	PeopleTools, Security, Permissions & Roles, Roles. Select the Role Queries tab.	Review content reference security by role.
Select a Content Reference or Content Reference Link	PORTAL_CREF_SELECT	On the Structure and Content page, scroll to the Content References list and click Add Content Reference Link.	Select a content reference or content reference link.
Content Reference Link Administration	PORTAL_CREF_LNK	<p>These options are available to navigate to the Content Reference Link Administration page:</p> <ul style="list-style-type: none"> On the Structure and Content page, scroll to the Content Reference group box. Click Edit for any content reference for which the Link check box is selected. Select PeopleTools, Portal, Structure and Content <p>Scroll to the Content References list and click Create Link.</p> <ul style="list-style-type: none"> Select PeopleTools, Portal, Structure and Content. Click Portal Objects. Click Homepage. Click Tabs. <p>Scroll to the Content References list and click Edit for any content reference.</p> <p>Click Create Content Reference Link.</p>	Add new or edit existing content reference links.

Page Name	Definition Name	Navigation	Usage
Related Links Group	PORTAL_RLNK_ADM	<ul style="list-style-type: none"> Select PeopleTools, Portal, Structure and Content. Click Portal Objects. <p>Scroll to the Content References list and click the Edit link.</p> <ul style="list-style-type: none"> Select the template name <i>RELATEDLINKS_TEMPLATE</i>. <p>Enter <i>RELLINK</i> in the Name field of the Content Reference Attributes group box.</p> <p>Enter a label for the attribute.</p> <p>Click the Edit Related Links link.</p>	Create related link groups.
Security	PORTAL_CLINK_SEC	Select PeopleTools, Portal, Structure and Content. Navigate the registry to the content reference link and click Edit. On the Content Reference Link Administration page, select the Security tab.	Set content reference link security.

Defining Content References

Access the Content Ref Administration page (PeopleTools, Portal, Structure and Content. Navigate the portal registry hierarchy until you are in the parent folder for the content reference; then click the Add Content Reference link.)

GeneralSecurity

Root >

Content Ref Administration

*Name

Author

QEDMO

*Label

Parent Folder

Root

Long Description

(254 Characters)

Product

*Valid from date

08/12/2011

Sequence number

Valid to date

Owner ID

Usage Type

Target

Creation Date

08/12/2011

Storage Type

Remote by URL

Template Name

WSRP Producible

No Template

URL Information

*Node Name

URL Type

PeopleSoft Component

Component Parameters

*Menu Name

*Market

GBL

*Component

Additional Parameters

Example: name1=value1&name2=value2

Hide from portal navigation

Hide from MSF navigation

IWC Message Events

Content Reference Attributes

Name

Delete

Label

Translate

Attribute Information

Attribute value

Add

Content Ref Administration page

Note. Many of the fields described in this section appear only when you enter certain field values.

Many of these fields are similar to fields on the Folder Administration page.

When you save a content reference that points to a PeopleSoft component or script on the default local node, the security settings for the content reference are inherited automatically from the menu or script security settings. Any permission lists or roles that you enter on the Content Reference Security page before saving the content reference are deleted and replaced by those of the component or script. These new security settings are also propagated up the folder hierarchy. This ensures that the content reference is visible and all folders are viewable, enabling users to navigate to the component.

Warning! Do not add internal component-based PeopleSoft content references using the Content Ref Administration page. Use the registration wizard instead so that information is consistent between PeopleSoft Application Designer and the portal. Any changes made to the menu definition in PeopleSoft Application Designer and not made to the portal content reference definition break the content reference. The portal navigation system uses the portal registry, not the internal location of menu definitions in PeopleSoft Application Designer.

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Using the Registration Wizard."

This table describes the fields that appear on the Content Ref Administration page:

Usage Type	<div>Select from these usage types:</div> <div><ul style="list-style-type: none">Frame template.Homepage tab.HTML template.Inline frame template.Pagelet.</div> <div><div>Important! The portal recognizes pagelets that are registered as portal objects in the Pagelets folder only. This restriction is not enforced by the portal registry; you can create a pagelet that is not recognized.</div><div>The name that you specify for a new pagelet can consist of letters, digits, and underscores (_). It cannot contain any spaces and cannot begin with a digit.</div></div> <div><ul style="list-style-type: none">Target.<div>This is the default value.</div><div>Based on the value that you select, the application shows, hides, disables, and enables the other fields on the page so that only the fields that are relevant to the usage type are available. For example, if you select a usage type of <i>Pagelet</i>, then Pagelet Attributes group box appears in which you can specify the default column, the refresh time, and other properties that apply specifically to pagelets.</div><div>For a typical PeopleSoft application page, set the usage type to <i>Target</i> (the default value) and the URL type to <i>PeopleSoft Component</i>.</div></div>
Storage Type	<div>This field is enabled when you selected a usage type of <i>Frame template</i> or <i>HTML template</i>.</div> <div>Select <i>Local</i> (in <i>HTML Catalog</i>) or the default value, <i>Remote by URL</i>.</div>

HTML Area	<p>This text area is enabled when you select a usage type of <i>HTML template</i> or <i>Frame Template</i> and <i>Local (in HTML Catalog)</i> as the storage type.</p> <p>An HTML template uses the HTML that is entered into the HTML area of a content reference that is located in the Templates folder, which you find within the Portal Objects folder. Bind variables cannot be used within the HTML area. The HTML area includes any HTML that is required for the template itself, plus tags that specify the URLs for template pagelets.</p>
Template Name	Select the template for this page. If you leave this field blank, the portal servlet automatically uses the default template of the node at runtime. The node default template is set up on the General Settings page. If a node default template has not been specified, it uses the portal default template.
WSRP Producible	This option appears only for usage types of <i>Target</i> and <i>Pagelet</i> . Select this option for PeopleSoft content that you intend to expose for remote WSRP portals to consume.
No Template	Select to have the target content (your application page) not wrapped with a portal template at runtime; by viewing your page, the user is effectively taken outside of the portal environment (this is not usually desirable). When this check box is selected, the Template Name field is hidden.
Create Content Reference Link	Click to create or edit a content reference link.
Add Content Reference	Click to add a content reference.
Test Content Reference	<p>Click to test a content reference.</p> <p>See Chapter 5, "Administering Portals," Creating Related Links, page 153.</p>
Hide from portal navigation	Select to hide the content reference from menu navigation.
Hide from MSF Navigation	MSF is not currently supported.
IWC Message Events	<p>Click to access the Interwindow Communications page.</p> <p>See Chapter 14, "Configuring Interwindow Communication," Configuring Interwindow Communication Publish and Subscribe Options, page 531.</p>

URL Information

The fields that appear in this group box are based on the selection that you make in the URL Type field.

Node Name	Select the node for this page. Select <i>Always use local</i> to associate the page with the default local node. The default local node is designated on the Portal Definitions page.
------------------	---

URL Type

Select from these URL types:

Non-PeopleSoft URL: Enter the URL in the Portal URL field that appears. This usage type can be used in conjunction with a defined node. The result is the concatenation of the selected node's URI text and the portal URL that you entered.

PeopleSoft Component: The Component Parameters region appears. Select the menu name, market, and component name to use for this content reference.

PeopleSoft Generic URL: Enter the URL in the Portal URL field that appears. This usage type can be used in conjunction with a defined node. The result is the concatenation of the selected node's URI text and the portal URL that you entered.

PeopleSoft Homepage Tab: In the Homepage tab attributes group box you can, designate a homepage as a dashboard page, add help functionality to a homepage tab, hide the pagelet action bar, and grant users the privilege to rename the homepage.

PeopleSoft Script: The iScript Parameters group box appears. Select the record (table) and field names, PeopleCode event name, and PeopleCode function name that specifies the iScript to use.

PeopleSoft Worklist URL: Enter the URL in the Portal URL field. Use the following format:

w/WORKLIST?ICAction=ICViewWorklist&Menu=Worklist&Market=GBL&PanelGroupName=WORKLIST

Remote Portlet: Select the producer and the portlet that you want to register manually. Only producers that have been registered are available for selection. Values for the fields Label and Long Description are supplied by default to the portlet's title and description. You can change these values. Click **Producer Details** to display the Display Producers page and view remote producers that have been defined in the system.

See [Chapter 20, "Using WSRP to Consume and Produce Remote Portlets," page 669](#).

Note. The availability of these options depends on the selected usage type.

Additional Parameters

Enter query string parameters to be appended onto a PeopleSoft component or PeopleSoft script. For example, enter *emplid=8001*.

Pagelet Attributes

The Pagelet Attributes group box enables you to manage pagelet behavior and appearance.

See [Chapter 12, "Developing and Registering Pagelets," Managing Pagelet Attributes, page 507](#).

Content Reference Attributes

Content reference attributes provide a framework to specify and store free-form information about a content reference. For example, you can use content reference attributes to specify and store keywords, help information, related content properties, and related links for content references.

In addition, portal-aware applications can use content reference attribute information for application-specific processing. Portal-aware applications are those that:

- Know about the portal registry and API.
- Know how to get information from a content reference.
- Contain code that can access the properties of content references and do something meaningful with the information.

Name	Enter an object name to identify the attribute programmatically. This name is not displayed to the user. <hr/> Note. The name can consist of letters, digits, and underscores (_) only. It cannot contain any spaces and cannot begin with a digit. <hr/>
Translate	Select to specify that the label and attribute values are translated.
Attribute Information	Click to create related links. You can create related links only after entering <i>RELLINK</i> in the Name field.
Label	Enter a descriptive name for the attribute to display to the user. The label is required only if Translate is selected.
Attribute value	Enter information about the attribute that the portal uses to apply the attribute to the content reference.

Links

This grid appears only if content reference links that connect to this content reference exist. Click Link Definition to view the definition of each listed content reference link.

Available Content Reference Attributes

This table describes the content reference attributes that your PeopleTools portal recognizes:

Content Reference Attribute Name	Description
KEYWORD	Use this attribute to specify keywords that you want to associate with this content reference. Any search using these keywords returns this content reference as a search result.

Content Reference Attribute Name	Description
Name of a default homepage object or image	<p>Use this attribute to override a default HTML object or image used for homepage components, such as tabs and pagelets, by providing the name of an alternative HTML object or image.</p> <p>Note. You can override only a specified set of homepage objects and images.</p>
IMAGE_BIND_12 to IMAGE_BIND_26	Use these attributes (numbered from 12 to 26) to insert additional custom images on a homepage tab.
NAVNEWWIN	<p>Use this attribute to specify that the component or page always opens in a new browser window.</p> <p>Note. This attribute applies to links on navigation collection template pagelets that appear in the pagelet area of a WorkCenter page.</p>
NAVTOPWIN	<p>Use this attribute to specify that the component or page opens in the current browser window and replaces the current page completely.</p> <p>Note. This option affects only the link behavior when it is rendered in a navigation collection template pagelet inside the pagelet area of a WorkCenter page.</p>
PTPGLTDEFERLOAD	<p>Use this attribute to disable the parallel loading of a pagelet so that it appears only when the entire homepage or dashboard is loaded.</p> <p>To defer the loading of the pagelet until the entire page loads, enter <i>false</i> as the attribute value.</p> <p>The default value is <i>true</i>.</p>
PSCACHECONTROL	Use this attribute to implement caching for individual pagelets and targets based on user, role, or application.
PORTAL_BANNER_PGLT	Use this attribute to create a banner pagelet, which appears at the top of a homepage or dashboard page and spans the width of the page.
PORTAL_ENCODING_OVERRIDE	Use this attribute to override the character set encoding that is used by the content reference.
PSTIMEOUT	Use this attribute to specify how long the portal should wait for a pagelet to load before it considers the pagelet unavailable.

Content Reference Attribute Name	Description
PT_PORTAL_PGLT_HDR_IMG	Use this attribute to add an icon to the header of the pagelet. The attribute value should be the name of the image definition that is stored in the image catalog using PeopleSoft Application Designer.
PTPPB_PAGELET_HELP_LINK	Use this attribute to specify a URL that you want to associate with the help link for a pagelet. Note. This attribute takes priority over a PeopleSoft Help ID if both properties are set.
RCF_SERVICE_PARAMETER	Use this attribute to provide related content service parameters in name/value pairs for the content reference. This presence of this attribute changes the default behavior of component-based URLs and supplants all parameters with those that you enter in the Value field.
RELLINK	Use this attribute to create related links for the content reference.

Content Reference Type Parameters

This table summarizes the valid combinations of usage type, storage type, and URL type or key HTML content supported for content references:

Usage Type	Storage Type	URL Type or Key HTML Content
Frame template	Remote by URL	PeopleSoft Script
Frame template	Remote by URL	Non-PeopleSoft URL
Frame template	Local (in HTML Catalog)	Target or IClientComponent element
Target	Remote by URL	PeopleSoft Component
HTML template	Local (in HTML Catalog)	Target, Pagelet, or Source element
HTML template	Remote by URL	PeopleSoft Script
HTML template	Remote by URL	Non-PeopleSoft URL
Homepage tab	Local (in HTML Catalog)	PeopleSoft Homepage tab
Inline Frame template	Local (in HTML Catalog)	Target or Pagelet component
Inline Frame template	Remote by URL	PeopleSoft Script
Inline Frame template	Remote by URL	Non-PeopleSoft URL

<i>Usage Type</i>	<i>Storage Type</i>	<i>URL Type or Key HTML Content</i>
Pagelet	Remote by URL	Non-PeopleSoft URL
Pagelet	Remote by URL	PeopleSoft Component
Pagelet	Remote by URL	PeopleSoft Script
Pagelet	Remote by URL	PeopleSoft Generic URL
Pagelet	Remote by URL	PeopleSoft Worklist URL
Pagelet	Remote by URL	Remote Portlet
Target	Remote by URL	PeopleSoft Generic URL
Target	Remote by URL	PeopleSoft Script
Target	Remote by URL	PeopleSoft Worklist URL
Target	Remote by URL	Non-PeopleSoft URL
Target	Remote by URL	Remote Portlet

See Also

[Chapter 5, "Administering Portals," Defining Folders, page 111](#)

[Chapter 7, "Working with Portal Templates," Understanding Portal Templates, page 203](#)

[Chapter 5, "Administering Portals," Managing General Portal Settings, page 171](#)

[Chapter 12, "Developing and Registering Pagelets," Configuring Pagelet Time-out Settings, page 510](#)

[Chapter 6, "Modifying the Portal Interface," Using Predefined Homepage HTML Objects, page 191](#)

[Chapter 8, "Using Portal Caching Features," Administering Server-Based Caching, page 222](#)

[Chapter 9, "Configuring the Portal Environment," Overriding Page Encoding, page 283](#)

Creating Related Links

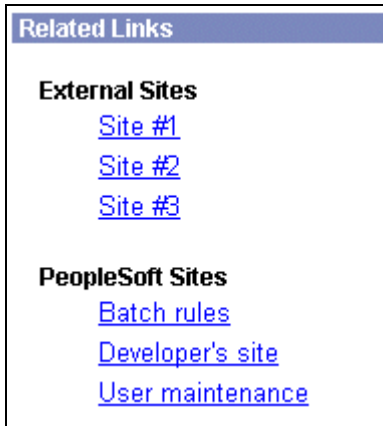
You can create related links for a portal page using the RELINK content reference attribute. You can create related links to external sites, to internal sites, and to other portal pages.

Note. Related links can be implemented with your choice of template on the content reference. Using the PeopleTools Related Links template displays the related links for the content reference, but not the portal menu navigation.

To create related links:

1. Define the RELINK content reference attribute.
2. Create related links groups.
3. Create related links for each group.
4. Test the related links.

In this sample page, you see the results of using the related links template and adding the RELINK attribute on a given content reference:



RELINK attribute results on a content reference

This table describes the page elements shown in the sample page:

<i>Page Element</i>	<i>Description</i>
Related Links	RELINK content reference attribute with the label Related Links.
External Sites	A related link group defined with the label External Sites.
Site #1, Site #2, Site #3	Related links for the External Sites group.
PeopleSoft Sites	A related link group defined with the label PeopleSoft Sites.
Batch rules, Developers' site, and User maintenance	Related links for the PeopleSoft Sites group.

Defining the RELINK Content Reference Attribute

To define a related link content reference attribute:

1. Access the Content Ref Administration page (PeopleTools, Portal, Structure and Content. Navigate to the appropriate content reference; then click the Edit link.)

2. Enter *RELATEDLINKS_TEMPLATE*. in the Template Name field.
3. In the Name field of the Content Reference Attributes region, enter *RELLINK*.

After you enter this value and move out of the field, the Attribute Information link changes to Edit Related Links.

4. Verify that the Translate check box is selected.
5. Enter a label for the related links attribute.

Note. You do not need to enter any information in the Attribute Value field. After you create the related link group and the related links, the system populates this field with all defined attributes as a string.

Creating Related Link Groups

Access the Related Links Group (PORTAL_RLNK_ADM) page.

To create a related link group:

1. In the Related Link Groups Label field, enter a name.
This name is the heading under which you place related links.
2. To create additional related link groups, click Add and repeat step 1.

Creating Related Links

To create a link:

1. In the Related Links region on the Related Links Group (PORTAL_RLNK_ADM) page, enter a name in the Related Link URL Label field.
This name appears as a link under the group name.
2. If you're linking to an internal site, select a node name from the list.
If you're linking to an external site, leave this field blank.
3. In the Related Link URL field, enter URL information for the link.
If the link is to an internal site, enter the URL or URI. The URL or URI can be to an internal site, to a portal, and so on. If you specified a value for the Portal URI Text field on the node definition, you can enter a specific document name as the target. The name is appended to the URI text for a full URL.
If the link is to an external site, enter the URL.
4. To define additional links for the group, click Add and repeat steps 2 and 3.

5. Click OK to save the entry and return to the Content Ref Administration page.

The Attribute Value field is populated with an XML-type formatted string. This string contains all the related link data and is stored in the PORTAL_ATTR_VAL field of the PSPRSMATTRVAL record for the content reference specified in the PORTAL_OBJNAME field. The following example shows a generated XML-type formatted string:

```
<GRPLBL>Worklist Pages
  <URLLBL>Worklist Details</URLLBL>
  <CP>Portal</CP>=>
  <URL>ICType=Panel&Menu=WORKLIST&Market=GBL&PanelGroupName=>
WORKLIST_DETAILS
  </URL>
  <URLLBL>Worklist Monitor</URLLBL>
  <CP>Portal</CP>=>
  <URL>ICType=Panel&Menu=WORKFLOW_ADMINISTRATOR&Market=>
GBL&PanelGroupName=WF_MONITOR_01
  </URL>
</GRPLBL>

<GRPLBL>Tasks
  <URLLBL>Task List Summary</URLLBL>
  <CP>Portal</CP>=>
  <URL>ICType=Panel&Menu=PORTAL_COMPONENTS&Market=>
=GBL&PanelGroupName=EO_PE_TASK_LIST=>
  </URL>
</GRPLBL>
```

Testing Related Links

You should test related links to view how the page appears to users, ensure that the labels are correct, and verify that the links function properly. To test related links, click Test Content Reference on the Content Ref Administration page just above the URL Information region.

Note that:

- If the portal cache is enabled on the web server, the related link template does not appear for a content reference page accessed from the menu until the cache expires.
- Related link URLs are displayed only when a user has security access to the URLs.

To grant security access to a related link URL, register the related link URL as a separate content reference and associate permission lists with it.

If the related link is not separately registered, then it is treated as having public access.

If a related link group contains no URLs to which the user has security access, the related link group is hidden. If the user does not have access to any of the related links associated with the target page, then the related links section of the template is hidden.

If the target page does not display any related links, ensure that:

- The application and portal databases are registered as content providers.

Also check for data-entry errors.

- The templates are registered in PeopleSoft Application Designer with the correct names.

If the templates are registered, check for data-entry errors.

- The correct related link template is specified for the target page.

Also ensure that the value of the PORTAL_TEMPL_NAME field in the PSPRSMDEFN record displays a correct related link template

- The RELINK string was saved on the target page in the content registry.

Check whether an XML string appears in the Attribute Value field on the Content Ref Administration page. If not, then delete and reenter the RELINK attribute for all the related link information.

- You assigned security in the registry to the user for the content references listed as related link URLs.

If not, assign security using the Content Reference Security page for registered related link URLs.

Setting Content Reference Permissions

Access the Content Reference Security page (Select PeopleTools, Portal, Structure and Content. Click the Edit link for a content reference; then select the Security tab.)

General Security

Root >

Content Reference Security

Label:

☐ Public

☒ Author Access

Security Authorizations					Personalize	Find	First	1-2 of 2	Last
	Type	Name	Description	View Definition					
1	Permission List			View Definition					
2	Role			View Definition					

Inherited Security Authorizations				Personalize	Find	First	1 of 1	Last
	Type	Name	Description	View Definition				
				View Definition				

Save Notify

General Security

Content Reference Security page

This page is similar to the Folder Security page. You can specify the permission lists and roles that grant security access to the content reference.

Certain URL types inherit the security permission list and role authorizations from the referenced object. For these URL types, you can view, but not modify the security by using the Content Reference Security page. All fields except Public and Author Access are display-only. This is the list of URL types that inherit security authorization:

- *PeopleSoft Component.*
- *Worklist.*
- *PeopleSoft Script.*

When you save a content reference that points to a PeopleSoft component or script on the default local node, the security settings for the content reference are inherited automatically from the menu or script security settings. Any permission lists or roles that you entered on the Content Reference Security page before saving the content reference are deleted and replaced by those of the component or script. These new security settings are also propagated up the folder hierarchy. This ensures that the content reference is visible and all folders are viewable, enabling users to navigate to the component.

Note. If you copy, move, or add a content reference to a folder, the security settings of the content reference are propagated up the folder hierarchy. This ensures that the content reference is visible and all folders are viewable, and it enables users to navigate to the content reference. If you remove the content reference from that folder, the security settings that were applied when it was added to the folder are also removed from the folder hierarchy, without disturbing any settings still required for other content references remaining in that folder.

Note. Content reference links do not inherit cascaded security from folders.

See Also

Chapter 5, "Administering Portals," Setting Folder Permissions, page 114

PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide, "Using the Registration Wizard"

PeopleTools 8.52: Security Administration, "Setting Up Permission Lists"

PeopleTools 8.52: Security Administration, "Setting Up Roles"

Reviewing Content Reference Permissions

You can use automated queries to discover the content references to which a given user, permission list, or role has access. Each query provides a different degree of detail about the properties of the content references.

Reviewing Content Reference Security by User

Access the User ID Queries page (PeopleTools, Security, User Profiles, User Profiles, User ID Queries).

This page contains several general-purpose security queries, along with the following links to content reference access queries for the selected user, which are described on the page:

- User ID's Content Reference Access.
- User ID's Content Reference (includes Portal) Access.

- User ID's Content Reference (includes Menu, Component and Market) Access.
- User ID's Content Reference (includes Portal, Menu, Component and Market) Access.

When you click one of the query links, a page appears in a new window, containing a grid with the information described for each content reference.

Reviewing Content Reference Security by Permission List

Access the Permission List Queries page (PeopleTools, Security, Permissions & Roles, Permission Lists, Permission List Queries).

This page is similar to the User ID Queries page, with the following equivalent links to content reference access queries for the selected permission list:

- Permission List's Content Reference Access.
- Permission List's Content Reference (includes Portal) Access.
- Permission List's Content Reference (includes Menu, Component and Market) Access.
- Permission List's Content Reference (includes Portal, Menu, Component and Market) Access.

When you click one of the query links, a page appears in a new window, containing a grid with the information described for each content reference.

Reviewing Content Reference Security by Role

Access the Role Queries page (PeopleTools, Security, Permissions & Roles, Roles, Role Queries).

This page is similar to the User ID Queries page, with the following equivalent links to content reference access queries for the selected role:

- Role's Content Reference Access.
- Role's Content Reference (includes Portal) Access.
- Role's Content Reference (includes Menu, Component and Market) Access.
- Role's Content Reference (includes Portal, Menu, Component and Market) Access.

When you click one of the query links, a page appears in a new window containing a grid with the information described for each content reference.

Testing Content References

After creating or modifying a content reference and setting up permissions, test the content reference.

1. Click Home in the portal navigation header
2. Use the Menu pagelet or the Main Menu to navigate to the content reference.
3. Click the content reference.

You can also click the Test Content Reference on the Content Ref Administration page.

Note. The Test Content Reference link appears on the Content Ref Administration page when the usage type of the content reference type is set to *Target* only.

Defining Content Reference Links

Access the Select a Content Reference or Content Reference Link page (Select PeopleTools, Portal, Structure and Content. Navigate the portal registry hierarchy into the parent folder in which you want the link to appear. Click the Add Content Reference Link link that appears after the Content References group box.)

Root

Select a Content Reference or Content Reference Link

Click the registry name and pick a content reference or content reference link.
Click "Cancel" to exit and not create a link.

Left | Right

- + SUPPLIER
- + PS_SITETEMPLATE
- + PORTAL
- + PARTNER
- + MOBILE
- + EMPLOYEE
- + CUSTOMER

Cancel

Select a Content Reference or Content Reference Link page

Click the folder icons to navigate the tree structure until you find the target of the link. Click the link and the Content Reference Link Administration page appears.

GeneralSecurity

Root >

Content Reference Link Administration

Target Information

Label: Sales Dashboard
Description:
*Name: SALES_ANALYSIS_LINK
*Portal: EMPLOYEE

Link Information

*Name:
Label:
Long Description:
(254 Characters)
Product:
Sequence number:
Owner ID: PPT PeopleTools

User ID: QEDMO
Parent Folder: Root
*Valid from date: 08/12/2011
Valid to date:
Creation Date: 08/12/2011

☒ No Template

URL Information

Hosted by this node: Always use local
URL Type: PeopleSoft Homepage Tab
Portal URL: h/?tab=SALES_ANALYSIS&psInkid=SALES_ANALYSIS_LINK

☐ Hide from portal navigation
☐ Hide from MSF navigation

Content Reference Link Attributes

Name:
Label:
Attribute value:

☒ Translate
[Attribute Information](#)

Delete
Add

Select All
Deselect All
Copy

Content Reference Attributes					
	Select	Name	Label	Attribute value	Translate
1	<input type="checkbox"/>	NAVTOPWIN		TRUE	<input type="checkbox"/>
2	<input type="checkbox"/>	PORTAL_OP_DASHBOARD		TRUE	<input type="checkbox"/>
3	<input type="checkbox"/>				<input checked="" type="checkbox"/>

Content Reference Link Administration page

PeopleSoft portals store content references based on their URLs, so you can't register two different content references that specify the same URL. Content reference links overcome this limitation by enabling you to define more than one menu item that invokes a given URL. You define a content reference link like any other content reference, but instead of specifying a URL, you specify an existing content reference as its destination. At runtime, clicking the content reference link on the portal menu connects to the destination content reference by URL.

Note. To the user, a content reference link looks and works the same as a content reference on the portal menu.

A content reference link can point to:

- A content reference of usage type *Target*.
- Another content reference link.

You can define a content reference link in the same locations that you can define a content reference of usage type *Target*. Multiple content reference links can connect to a single destination content reference.

By connecting one content reference link to another content reference link, you can create a chain of redirection, which always ends in a content reference of usage type *Target*. You don't need to know what the ultimate destination is when you define your new content reference link, and you can duplicate the property settings of the content reference link to which it connects rather than those of the ultimate destination. However, Oracle recommends that you limit your use of chains of content reference links because they impose processing overhead on your portal when selected.

Warning! If you delete a destination content reference or content reference link, all content reference links that connect to it are also deleted. This behavior propagates back through any content reference link chain.

Target Information

Name	Select the destination content reference to which you want this content reference link to connect.
Portal	Select the portal that contains the destination content reference to which you want this content reference link to connect. The <i>Always use local</i> value refers to the portal where the content reference link is defined, so moving or copying it to a different portal changes the resolution of this value.
Important! If you change this value, you must also select a new link content reference.	

Link Information

Name	Enter a unique object name for this content reference link. This field is required.
Test Content Reference Link	Click to test this content reference link. Clicking this button should produce the same result as clicking the content reference link from the portal menu.

If you omit values in following fields, their values are inherited from the destination content reference:

- Label.
- Long Description.
- Product.
- Sequence number.
- Object owner identifier.
- No Template.
- Template Name.
- Valid from date.
- Valid to date.

URL Information

These fields are display-only and reflect the URL information of the content reference that you are linking to.

See [Chapter 5, "Administering Portals," URL Information, page 148.](#)

Content Reference Attributes

If the destination content reference has defined content reference attributes, they appear in this grid. You can copy them from the grid to the Content Reference Link Attributes group box using the following page elements:

Select	Select to designate a content reference attribute for copying to the Content Reference Link Attributes group box.
Select All	Click to designate all content reference attributes for copying to the Content Reference Link Attributes group box.
Copy	Click to copy the selected content reference attributes to the Content Reference Link Attributes group box.

Content Reference Link Attributes

Content reference link attributes and content reference attributes share the same page fields.

You can define content reference link attributes directly in this group box, or you can copy an attribute from the destination content reference using the Content Reference Attributes grid, and modify it to suit your purposes.

At runtime, the content reference link attributes defined here are combined with the content reference attributes defined at the link destination, and they are applied together.

Important! If a content reference link attribute defined here has the same name as a content reference attribute defined at the link destination, but a different attribute value, the value defined here takes precedence and is applied at runtime.

Content reference link attributes are independent and separate from the content reference attributes at the link destination. Modifying one has no effect on the other.

Setting Content Reference Link Permissions

Access the Content Reference Link Security page (Click the Security tab on the Content Reference Link Administration page.)

The fields on this page are disabled because content reference links inherit the security settings of the destination object. The inherited security settings propagate upward through the folder hierarchy from the content reference link.

See Also

[Chapter 5, "Administering Portals," Setting Content Reference Permissions, page 157](#)

Managing Folders and Content References

This section discusses how to:

- Copy folders and content references.
- Move folders and content references.
- Delete folders and content references.
- Synchronize portal object security.
- Register components in multiple portals.
- Register components multiple times.

Pages Used to Manage Folders and Content References

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Copy Object to a Different Portal Registry	PORTAL_COPY_OBJ	<ul style="list-style-type: none"> PeopleTools, Portal, Structure and Content. Access the Content Ref Administration page. Click the Copy Object button. Select PeopleTools, Portal, Structure and Content. Access the Folder Administration page. Click the Copy Object button. 	Copy folders and content references.
Select New Parent Folder	PORTAL_CREF_MOVE PORTAL_FLDR_MOVE	<ul style="list-style-type: none"> Select PeopleTools, Portal, Structure and Content. Access the Content Ref Administration page. Click the New Parent Folder button. Select PeopleTools, Portal, Structure and Content. Access the Folder Administration page. Click the New Parent Folder button. 	Move folders and content references.
Portal Security Synchronization	PORTAL_SEC_SYNC	PeopleTools, Portal, Portal Security Sync	Synchronize portal object security.
General Settings	PORTAL_REG_ADM	PeopleTools, Portal, General Settings	<p>Manage the settings for the portal that you're signed in to.</p> <p>Note. To modify other portals, use the Portal Definitions page.</p>

Copying Folders and Content References

Access the Copy Object to a Different Portal Registry page (Select PeopleTools, Portal, Structure and Content, Portal Objects, Content Ref Administration. Click the Copy Object button).

To copy an object:

1. Navigate to the portal registry to which you want to copy the object.

Click the portal name to copy the object to root level. To expand the folders and navigate further into the registry structure, click the folder icon (not the folder name).

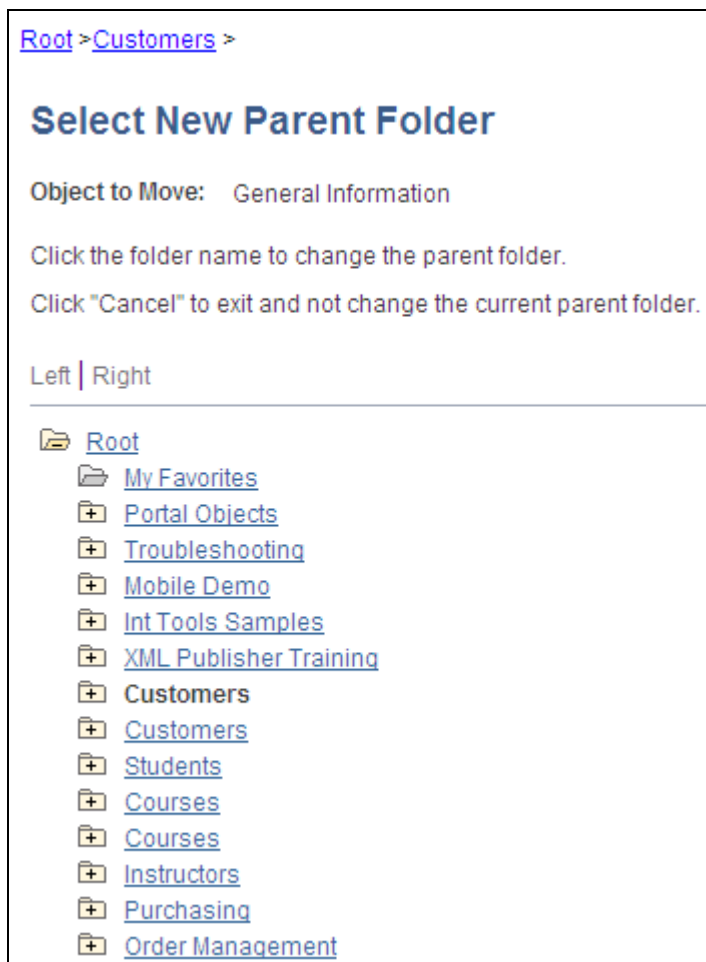
Important! Clicking the folder name starts the copying process.

2. (Optional) If you are copying a folder, the Recurse subfolders? check box appears. Select this check box to copy the folder and all the subfolders and content references within it.
3. Click the folder name to copy the object into that folder.
4. Click the Yes - Copy button to proceed.

Warning! If you attempt to copy an object to a location that already contains an object of the same name and type, the newly copied object replaces the existing object. For example, if you copy a folder and subfolders to another portal where a subfolder by that name already exists, the existing subfolder is deleted and replaced with the subfolder from the source portal. No warning message appears.

Moving Folders and Content References

Access the Select New Parent Folder page (PeopleTools, Portal, Structure and Content, Portal Objects, Content Ref Administration Click the New Parent Folder button).



Select New Parent Folder page

The current parent folder is shown in bold type, and the folder tree is expanded just enough to show the location of the current parent folder. All other folders are shown collapsed. You can expand the tree to show content references, which are displayed as tree leaves and appear in normal text.

To move a folder or content reference to a new parent folder:

1. Navigate to the new parent folder by clicking the appropriate folder icons with plus signs.

Note. When navigating, do not click the folder name unless it's the name of the new parent folder that you want to select. Once you click a folder name, as opposed to the icon, the folder name that you selected becomes the new parent folder.

2. Select the new parent folder by clicking the folder name.
3. Click Save.

Note. When you move a content reference that points to a component or script at the default local node, its security settings are cascaded up to its new parents all the way up to the root folder. In addition, when you move a folder, its security settings (not including security inherited from a subordinate object) are cascaded up to its new parent, all the way up to the root folder. This ensures that the folder or content reference is accessible in its new location.

Deleting Folders and Content References

To delete a folder or content reference:

1. Select PeopleTools, Portal, Structure and Content.
2. In the portal registry, navigate to the folder or content reference that you want to delete.
3. Click the Delete button next to the object that you want to delete.

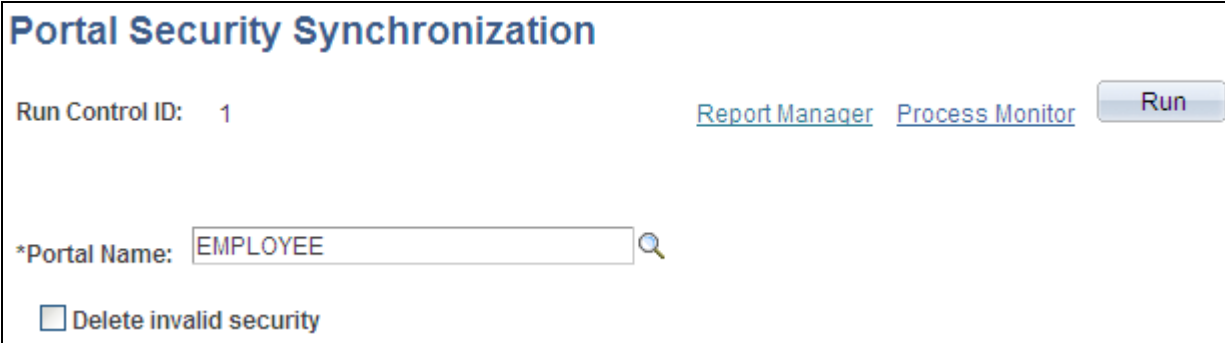
All child folders and child content references are deleted when you delete the selected folder. Make sure that no critical dependencies exist for these objects before deleting the parent folder.

Warning! Do not delete folders or content references that are delivered with PeopleTools. These default objects have special significance to your portal. Changing them could make your portal unusable.

When you delete a content reference or a content reference link, all content reference links that connect to it are also deleted. This behavior propagates back through any content reference link chain.


Synchronizing Portal Object Security

Access the Portal Security Synchronization page (PeopleTools, Portal, Portal Security Sync).



Portal Security Synchronization

Run Control ID: 1 [Report Manager](#) [Process Monitor](#) [Run](#)

*Portal Name: 

☐ Delete invalid security

Portal Security Synchronization page

The hierarchical relationships and dependencies between objects in the portal registry determine what security settings each object must have. The portal won't work correctly if these security relationships aren't maintained. Here are some examples of these relationships:

- A folder that is not public or hidden must have at least the level of access that its immediate child objects (folders, content references, and content reference links) have.
- A content reference link must have exactly the same level of access as the object (content reference or content reference link) to which it links.
- A content reference that represents a PeopleSoft component or iScript must have exactly the same level of access as the object that it represents.

Portal object security settings can become unsynchronized when you move portal objects from one database to another using the Project Copy feature in PeopleSoft Application Designer. When you merge projects this way, if the projects contain any portal objects with identical names, the security settings of the portal objects in the last project copied overwrite the security settings of portal objects copied earlier. Also, when a copied portal object doesn't overwrite an existing object, it changes the structure of the resulting portal registry hierarchy.

Use the Portal Security Synchronization page to reinstate the correct security relationships between objects in the portal registry after you copy a project that contains portal objects.

This page enables you to run the Application Engine program `PORTAL_CSS` from the standard menu navigation. You can use this feature on local portals only, not on portals on remote databases. In addition, from any local portal, you can run this program against another local portal.

To synchronize portal object security, specify a local portal in the Portal Name field and click Run. The portal objects are synchronized in the following way:

1. The security settings of each content reference are compared to the component or iScript that it represents, and updated to match.
2. The security settings of each content reference link are compared to the content reference or content reference link to which it connects, and updated to match.
3. The security settings of each content reference and content reference link are propagated to its parent folder, in addition to the parent folder's existing settings.

None of the parent folder's existing security access is reduced.

Note. The settings aren't propagated if the parent folder is public or hidden.

4. The security settings of each folder are propagated to its parent folder, in addition to the parent folder's existing settings.

None of the parent folder's existing security access is reduced.

Note. The settings aren't propagated if the parent folder is public or hidden.

The following warning messages may appear when security synchronization cannot occur:

- No permissions in PSAUTHITEM for object %1
- Cref %1 points to Menu: %2, Component: %3 which doesn't exist.

Note. When this message appears, you need to delete the invalid content reference .

The security administrator and portal administrator are responsible for determining why an object has no security or why an object does not exist.

Delete Invalid Security Option

When you move portal objects from one database to another, roles and permission lists assigned to folders and content references on the source database may not exist on the target database and therefore become invalid. Select the Delete Invalid Security check box on the Portal Security Synchronization page to remove nonexisting roles and permission lists from folders and content references.

Note. When the Delete Invalid Security option is selected, the PORTAL_CSS process runs more slowly because it checks every role and permission list on every portal registry structure. When a nonexistent role or permission list is found on the portal registry structure, it is removed from that portal registry structure.

Registering Components in Multiple Portals

After you use Registration Wizard to register a new component in a portal, you may need to register the component in other portals. You can register components in multiple portals in two ways:

- Use Registration Wizard again.

Select the second option: Add this component to a portal registry. Then reenter the content reference label, long description, and sequence number (if not 1) to match the other portal's entry.

- Copy the component to other portals.

When you have copied your component to another portal, open that portal to verify that it was copied as anticipated.

See [Chapter 5, "Administering Portals," Copying Folders and Content References, page 166.](#)

See Also

[Chapter 5, "Administering Portals," Changing Default Portals, page 107](#)

Registering Components Multiple Times Within a Portal

The portal registry does not allow a URL to be registered more than once. When you define a content reference, the component name is part of the URL. If you must register the same component URL in more than one location, add a unique character to the parameter field. (Be sure that you really must register a URL twice because this creates upgrade and maintenance concerns.) See the following example, in which the additional numeric character *1* is added to the Additional Parameter field:

GeneralSecurity

Root > Customers >

Content Ref Administration

Name:

T1_CUST_GBL

Author:

PTOOLS1

*Label:

General Information

Copy object

Select New Parent Folder

Long Description:

(254 Characters)

Enter, view, and update customer details (0400, 0410)

Product:

PPT

*Valid from date:

10/22/2003

Sequence number:

10

Valid to date:

Object Owner ID

PPT

PeopleTools

Creation Date:

10/22/2003

Usage Type:

Target

WSRP Producible

Storage Type:

Remote by URL

No Template

Template Name:

(Invalid Value)

8.50 default template

Create Content Reference Link

Add Content Reference

Test Content Reference

URL Information

*Node Name

LOCAL_NODE

URL Type:

PeopleSoft Component

Component Parameters

*Menu Name:

0400_PTOOLS1

*Market:

GBL

*Component:

T1_CUST

Additional Parameters:

1

Example: name1=value1&name2=value2

Registering a component the second time

See Also

Chapter 5, "Administering Portals," Registering Components in Multiple Portals, page 170

Managing General Portal Settings

This section discusses how to manage general portal settings.

Page Used to Manage General Portal Settings

Page Name	Definition Name	Navigation	Usage
General Settings	PORTAL_REG_ADM	PeopleTools, Portal, General Settings	Manage general settings for portals.

Managing General Settings for Portals

Access the General Settings page (PeopleTools, Portal, General Settings).

General Settings

Portal Name: EMPLOYEE
Long Description: (254 Characters) Employee-facing registry content
Default Template Name: DEFAULT_TEMPLATE 8.50 default template
Object Owner ID: PPT PeopleTools

Folder Navigation

☒ Is Folder Navigation Enabled
Folder Navigation Object Name: PT_PTPP_SCFNAV_BASEPAGE_SCR Base Navigation Page
Chart Navigation Object Name: PT_PORTAL_DEFAULT_CHART_NAV Default Chart Navigation Page

Portal Search Options

☒ Display search in navigation ☐ Display search in header ☒ Display global search in header
Auto Suggest Limit: 50

Search Results Page

***Menu Name:** PORTAL_ADMIN Portal Administration
***Component Name:** PTSF_GLOBAL_SEARCH
***Market:** GBL

Navigation Options

***Navigation Type:** Drop-down

Favorites Options

☒ Display My Favorites folder ☒ Display Recently Used

Menu Text Options

☒ Display menu text in header Menu title position 47
☒ Display menu in nav pagelet

Disable Drop-down Menu Sort : ?

JPN,KOR,THA,ZHS,ZHT

Header Frame Options

Header frame rows: 65

Node Templates

*Node Name	*Default Template Name		
PT_LOCAL	8.50 default template	Add	Delete

General Settings page

Use this page to manage the settings for the portal that you're signed in to. To modify other portals, use the Portal Definitions page.

Note. You must sign out of the portal and sign in again before your changes take effect.

Default Template Name	Select the template to use to wrap all registered content that has not been associated with another template in this portal, as well as all unregistered content.
------------------------------	---

Warning! If you change the default template, make sure that the new default is able to handle any arbitrary page.

See [Chapter 5, "Administering Portals," Administering Portal Definitions, page 104.](#)

Folder Navigation

You enter the default folder navigation values for the current portal on this page. Enabling folder navigation at the portal level enables folder navigation for all standard folders and SmartNavigation in the portal. You can subsequently disable folder navigation at the individual folder level

Folder navigation is not enabled unless you select the Is Folder Navigation Enabled check box on both the Portal Definition and General Settings pages. The Is Folder Navigation Enabled check box is selected by default in the standard PeopleTools portals.

See [Chapter 5, "Administering Portals," Adding and Editing Portal Definitions, page 106.](#)

Note. If you disable and then re-enable folder navigation at this level, folder navigation is re-enabled for *all* folders. All overrides at the individual folder level are cleared.

Portal Search Options

Use the fields in this group box to enable system-level search options.

Display search in navigation	Select to display the search at the top of the menu pagelet or main menu.
-------------------------------------	---

Display search in header	Select to display the search field in the universal navigation header. This search is limited to the current application. You can display either the standard search or the global search in the universal navigation header. The two options are mutually exclusive.
---------------------------------	---

Display global search in header	Select to display the portal-wide search field in the universal navigation header. This search spans all application portal sites. You can display either the standard search or the global search in the universal navigation header. The two options are mutually exclusive.
--	--

Auto Suggest Limit	Enter the maximum number of rows to appear in the drop-down autocomplete list for edit and prompt fields that have autocomplete enabled at the record definition level.
---------------------------	---

The default value is 50.

Search Results Page

Enter the names of the menu definition, component definition, and market of the Search Results page. This group box is disabled when the `Display global search in header` check box is selected.

Navigation Options

Navigation Type Select *Left* to enable menu navigation that uses the Menu pagelet, which appears at the side of the page.

Select *Drop-down* to enable menu navigation that uses the navigation header, which appears at the top of the page.

Important! Many PeopleTools portal-related features, such as SmartNavigation, persistent search, and the My Favorites and Recently Used menus are available only when you use drop-down navigation type.

Favorites Options

Display My Favorites folder Select to enable the application user to save favorite components and pages in the My Favorites folder, which appears in the Favorites drop-down menu. This option is available only when you select *Drop-down* as your navigation type.

Display Recently Used Select to enable the portal to store and display the five most recently visited components as a folder under the Favorites menu. This option is available only when you select *Drop-down* as your navigation type.

Menu Text Options

Display menu text in header Select to show the menu title in the header rather than in the left frame above the menu.

Display menu in nav pagelet Select to show the menu title in the pagelet header.

Note. In PeopleTools, the menu title is displayed in the header. If both `Display menu text in header` and `Display menu in nav pagelet` are selected, the menu title appears twice.

Menu title position The number of pixels from the top of the frame that the menu expand/collapse bar appears in a frame template. The default is 47.

Disable Drop-down Menu Sort

Enter the PeopleSoft language codes of the languages for which you want to remove the drop-down menu sort functionality at the system level. The drop-down menu sort is disabled for these languages by default: Japanese, Korean, Thai, Simplified Chinese, and Traditional Chinese.

Enter PeopleSoft language codes in a comma-separated list. Select PeopleTools, Utilities, International, Languages to see all PeopleSoft language codes.

Header Frame Options

This group box sets the number of header frame rows if you are using the PeopleTools default template. The value entered for this field determines the size of the header frame. The default value for the PeopleTools portal is 65 and the default value for the Application Portal is 100.

Node Templates

Configure node-specific template information.

Node Name Select from the list of nodes defined for this portal. If you add a new node name, you must also specify the default template for that node.

Default Template Name Select the default template for all content references using this node. The selected template is applied only to content references for which the No Template check box is deselected but no template is selected.

See Also

PeopleTools 8.52: PeopleSoft Applications User's Guide, "Working With PeopleSoft Applications," Using Menu Pagelets

Managing Portal Objects

The Portal Objects folder is of special significance to the portal. This section discusses how to:

- View Portal Objects folder settings.
- Manage templates.
- Add and edit templates.
- Delete templates.

Pages Used to Manage Portal Objects

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Structure and Content - Portal Objects	PORTAL_OBJ_LIST	PeopleTools, Portal, Structure and Content, Portal Objects	View portal objects folder settings.

Page Name	Definition Name	Navigation	Usage
Content Ref Administration	PORTAL_CREF_ADM	PeopleTools, Portal, Structure and Content, Portal Objects, Templates Click Edit next to the name of the template that you want to access.	Manage templates. Add and edit templates. Delete templates.

Viewing Portal Objects Folder Settings

Access the Structure and Content page (Select PeopleTools, Portal, Structure and Content. In the Folders group box, click the Edit link for Portal Objects).

[Root](#) > Portal Objects

Structure and Content

* Click the folder label to view the child folders and content references for that folder
 * Click the "Edit" link to edit the folder definition

Folders			
Label	Edit	Sequence number	
Homepage	Edit	0	Delete
Navigation Collections	Edit	0	Delete
Pagelets	Edit	0	Delete
Template Pagelets	Edit	0	Delete
Templates	Edit	0	Delete
WSRP Produced Portlets	Edit	0	Delete

[Add Folder](#)

* Click the "Edit" link to edit the content reference definition

Content References						
Link	Label	Edit	Sequence number	Create Link	Number of links	
<input type="checkbox"/>	Portal Expire	Edit	0	Create Link	0	Delete

Structure and Content page

Understanding Templates

Templates are stored in the HTML catalog and are edited through the portal administration pages. For the portal to process templates properly, they must be referenced in the Templates folder within the Portal Objects folder. Storing templates in the HTML catalog enables them to be upgraded like other objects.

Templates are stored in the HTML catalog according to the following naming convention: Template names always begin with the literal *PR_*. Here's an example name: *PR_PORTALNAME_CONTENT REFERENCE ID*

The following example shows a sample template:

The screenshot displays the 'Content Ref Administration' web interface. It features a navigation bar with 'General' and 'Security' tabs. Below the navigation bar, the breadcrumb path is 'Root > Portal Objects > Templates >'. The main title is 'Content Ref Administration'. The form includes the following fields and sections:

- Name:** DEFAULT_TEMPLATE
- Author:** PTDMO
- Parent Folder:** Templates
- *Label:** 8.50 default template (with a 'Copy object' button and a 'Select New Parent Folder' button)
- Long Description:** (254 Characters) 8.50 default template
- Product:** PT
- *Valid from date:** 05/18/2000
- Sequence number:** (empty field)
- Valid to date:** (empty field)
- Object Owner ID:** PPT (with a search icon and 'PeopleTools' text)
- Usage Type:** (Invalid Value) (dropdown menu)
- Creation Date:** 05/18/2000
- Storage Type:** Remote by URL (dropdown menu)
- URL Information:**
 - *Node Name:** LOCAL_NODE (with a search icon)
 - URL Type:** PeopleSoft Script (dropdown menu)
 - iScript Parameters:**
 - *Record (Table) Name:** WEBLIB_PT_NAV
 - *Field Name:** ISCRIPT1
 - *PeopleCode Event Name:** FieldFormula
 - *PeopleCode Function Name:** IScript_PT_NAV_TPL_FRAME
 - Additional Parameters:** (empty field)
 - Example:** name1=value1&name2=value2
- Producer:** (empty field with a search icon)
- Portlet:** (empty field with a search icon)
- Buttons:** 'Add Content Reference' (link), 'Copy object' (button), 'Select New Parent Folder' (button), 'Producer Details' (link)

Example of portal template

Templates can be larger than the maximum size permitted by the HTML catalog. Such templates are automatically divided into several files in the HTML catalog. In this case, a letter is appended to each HTML object that makes up the template, as shown here:

PR_EMPLOYEE_123456789_A

PR_EMPLOYEE_123456789_B

See Also

Chapter 7, "Working with Portal Templates," page 203

Adding and Editing Templates

Access the Content Ref Administration page (Select PeopleTools, Portal, Structure and Content. Then click these links in succession: Portal Objects, Templates).

To add a new template, click Add Content Reference in the Content References region at the bottom of the Content Ref Administration page.

To edit an existing template, click the Edit link next to the template that you want to edit in the Content References region at the bottom of the Content Ref Administration page.

See [Chapter 5, "Administering Portals," Defining Content References, page 145](#).

The fields for adding a dynamic template are the same as for any content reference, except for additional iScript parameter fields. These fields are enabled only if the storage type is set to *Remote by URL*.

If the URL Type field is set to *PeopleSoft Script*, the iScript Parameters group box appears. Select the record name, PeopleCode event name, field name, and PeopleCode function name that stores the iScript that the content reference accesses to build the template.

See Also

[Chapter 5, "Administering Portals," Defining Folders, page 111](#)

[Chapter 5, "Administering Portals," Defining Content References, page 145](#)

Deleting Templates

You delete templates the same way that you delete folders and content references.

See [Chapter 5, "Administering Portals," Deleting Folders and Content References, page 168](#).

Note. Do not delete the DEFAULT_TEMPLATE portal template. This template has special significance to your portal and must always be available. Deleting this template could make your portal inoperative. Oracle recommends that you not delete any template unless you are fully aware of how it's used within the portal.

Reviewing Menu Item Information

This section lists common elements used in the section and discusses how to:

- View menu item details.
- View menu folder structure information.
- View menu security details.

Common Elements Used in This Section

Menu Item	Displays the menu item ID. The menu item label is displayed to the right.
Content Provider Name	Displays the database node that contains the PeopleSoft page.
Product	(Optional) Displays text that is used to tag menu items for inquiry. <hr/> Note. Each PeopleSoft product line has a naming standard. For example, PAPP is always used for PeopleSoft Application Portal. <hr/>
Menu Path	Displays the navigation path for this menu item.
Hidden	If this check box is selected, then the menu item is hidden from portal menu navigation. However, you can still access the content by using a link that is incorporated on a target page or pagelet.
Is Public	If this check box is selected, the menu item is available for all users. No data appears in the Accessible Permissions, Accessible Roles, and Accessible Users scroll areas.
Auth Access (author access)	If this check box is selected, the author can access the menu item regardless of how other permissions are set.

Pages Used to Review Menu Item Information


<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Menu Item Detail	PT_IQREGITEM	PeopleTools, Portal, View Menu Item Detail	View menu item details.
Menu Folder Structure	PT_IQREGFOLDERS	PeopleTools, Portal, View Menu Folder Structure	View menu folder structure information.
Menu Security	PT_IQREGSECURTY	PeopleTools, Portal, View Menu Security	View menu security details.


Viewing Menu Item Details

Access the Menu Item Detail page (PeopleTools, Portal, View Menu Item Detail).

Menu Item Detail

Portal Object Name: PERSONAL_DATA_GBL Personal Data
Content Provider Name: LOCAL_NODE Use Local Node
Product: PSU
Usage Type: Target
Template name: DEFAULT_TEMPLATE 8.50 default template
Component: PERSONAL_DATA
Long Description: View/edit details about company employees
Menu Path: Root > Employees >
Hidden: ☐ **Is Public:** ☐ **Auth Access:** ☒ PTRN

Accessible Permissions		Customize Find View All  First 1-2 of 2 Last
Permission Name	Description	
PSU1020	PSU HR Administration	
ALLPAGES	All pages and weblibs	

Accessible Roles		Customize Find View All  First 1-4 of 4 Last
Role Name	Description	
Developer	Application Developer	
PeopleTools I Student	Student in course 0400	
HR Specialist	HR Specialist	
All Pages	All Pages	

Accessible Users		Customize Find View All  First 1-10 of 18 Last
User ID	Description	
PTMBL	PeopleTools Training - Mobile	
PTDMO	Developer	
TC1	Training Coordinator 1	
PTTRA	Application Engine student	
PTCODE	PeopleCode Student	
MGR2	Manager2	
PTWF	PeopleTools Training Workflow	
MGR1	Manager1	
PTXMLP	XMLP Training user	
PTOOLS1	Course 0400 User	

Menu Item Detail page

Use this page to review the portal registry information for a menu item. The details that appear include the exact users and the roles and permission lists that allow access to a given menu item.

Usage Type	Displays the type of object that is registered in the portal as a content reference. Possible values are <i>Target</i> , <i>Pagelet</i> , <i>Frame Template</i> , <i>HTML Template</i> , and <i>Homepage Tab</i> .
Template Name	Displays the template that controls how content appears on the page. The default template is used except for special circumstances.
Component, iScript, and URL Type	Information that appears depends on the URL type that is entered on the Content Ref Administration page. Possible URL type values are <i>Non-PeopleSoft URL</i> , <i>PeopleSoft Component</i> , <i>PeopleSoft Generic URL</i> , <i>PeopleSoft Script</i> , <i>PeopleSoft Worklist URL</i> , and <i>Remote Portlet</i> . The availability of these values depends on the selected usage type.
Long Description	Displays the text that appears when you move the cursor over the menu navigation link.

Note. The Accessible Permissions, Accessible Roles, and Accessible Users scroll areas are populated only if the access type is set to *permission list*.

See [Chapter 5, "Administering Portals," Defining Content References, page 145](#).

Accessible Permissions

This section displays the permission list name and description for all permission lists that can access this menu item.

Note. If the component or iScript is using a local node with no extra parameters, the permission lists are automatically populated based on the menu object component security.

Accessible Roles

Displays the role name and description for all roles that can access this menu item.

Accessible Users

Displays the user ID and description for all users that can access this menu item.

Viewing Menu Folder Structure Information

Access the Menu Folder Structure page (PeopleTools, Portal, View Menu Folder Structure).

Menu Folder Structure

Select the Node Name and Product for Menu information that you would like to see. If left blank, all possible information will be shown.

Portal Name:

EMPLOYEE

Content Provider

Name:

Product:

Menu Folder:

PT_SECURITY

Security

Fetch

Menu Folder Structure

Customize | Find | View All | First | 1-42 of 42 | Last

Objects

Details

Menu Item	Details	Menu Path	Seq Nbr.	Portal Object Name
Mass Change Operator Security		Root > PeopleTools > Security >	201	PT_MC_OPR_SECURITY_GBL
Common Queries		Root > PeopleTools > Security >	200	PT_MAINT_SEC_QUERIES_GBL
Cache Directory Schema		Root > PeopleTools > Security > Directory >	20	PT_RUN_DSSHEMA_GBL
Delete Directory Configuration		Root > PeopleTools > Security > Directory >	60	PT_DSPURGEDIRID_GBL
Workflow Address Book		Root > PeopleTools > Security > Directory >	60	PT_WF_NOT_LDAP
User Profile Map		Root > PeopleTools > Security > Directory >	40	PT_DSUSRPROF2_GBL
Configure Directory		Root > PeopleTools > Security > Directory >	10	PT_PSDSETUP_GBL
Authentication Map		Root > PeopleTools > Security > Directory >	30	PT_DSSECMAP_GBL
Role Membership Rules		Root > PeopleTools > Security > Directory >	50	PT_DSROLERULE_GBL
Algorithm Chain		Root > PeopleTools > Security > Encryption >	110	ALGORITHM_CHAIN
Algorithm Keyset		Root > PeopleTools > Security > Encryption >	120	ALGORITHM_KEYSET
Delete Encryption Profile		Root > PeopleTools > Security > Encryption >	135	CRYPT_PRFL_DELETE
Load Encryption Libraries		Root > PeopleTools > Security > Encryption >	100	ALGORITHM_PROFILE
Test Encryption Profile		Root > PeopleTools > Security > Encryption >	140	CRYPT_DEMO
Encryption Profile		Root > PeopleTools > Security > Encryption >	130	ENCRYPTION_PROFILE


Menu Folder Structure page: Objects tab

Use this page to view a list of content references for a selected portal registry folder. You see the contents of one or several menu folders.

Portal Name	Displays the name of the portal that you are currently using.
Content Provider, Name, Product, and Menu Folder	Select a combination of criteria for filtering the results. If any of the fields are left blank, all possible information for those fields are shown.

Objects Tab

Select the Objects tab.

Menu Item	Displays the menu item description.
	Click to view more detail information on the Menu Item Detail page.
Seq Nbr (sequence number)	Displays where the menu item is located in its menu folder. If this field is blank, the menu item is added to the top of the list of content references within its folder and displayed in alphabetical order.
Portal Object Name	Displays the menu item ID.

Details Tab

Select the Details tab.

Menu Folder Structure								
Objects		Details		Customize Find View All First 1-42 of 42 Last				
Menu Item	Product	Content Provider Name	Component	iScript	Hidden	Is Public	Auth Access	User ID
Mass Change Operator Security	PT	LOCAL_NODE	MC_OPR_SECURITY		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Common Queries	PT	LOCAL_NODE	MAINT_SEC_QUERIES		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Cache Directory Schema	PT	LOCAL_NODE	RUN_DSSCHEMA		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Delete Directory Configuration	PT	LOCAL_NODE	DSPURGEDIRID		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Workflow Address Book	PT	LOCAL_NODE	PT_WF_NOT_DSCFG		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
User Profile Map	PT	LOCAL_NODE	DSUSRPROF2		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Configure Directory	PT	LOCAL_NODE	PSDSSETUP		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Authentication Map	PT	LOCAL_NODE	DSSECMAP		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Role Membership Rules	PT	LOCAL_NODE	DSROLERULE		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Algorithm Chain	PT	LOCAL_NODE	ALGORITHM_CHAIN		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	QEDMO
Algorithm Keyset	PT	LOCAL_NODE	CRYPT_KEYSET		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	QEDMO
Delete Encryption Profile	pt	LOCAL_NODE	CRYPT_PRFL_DELETE		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	QEDMO
Load Encryption Libraries	PT	LOCAL_NODE	ALGORITHM_PRFL		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	QEDMO
Test Encryption Profile	PT	LOCAL_NODE	CRYPT_DEMO		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	QEDMO
Encryption Profile	PT	LOCAL_NODE	ENCRYPTION_PRFL		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	QEDMO

Menu Folder Structure page: Details tab

User ID

If the Auth Access check box is selected, this field displays the author's user ID.

Viewing Menu Security Details

Access the Menu Security page (PeopleTools, Portal, View Menu Security).

Menu Security

Select the level of Object security and Folder visibility for Menu information that you would like to see. Please note that inquiries which return a large amount of rows may take a minute or two.

Object Security

☒ User User ID:

☐ Role

☐ Permission List

Folder Visibility

☐ Show All

☒ Choose Top Level Menu Item:

☐ Show Top Level Only

Fetch

Menu Security

Customize | Find | View All | | | First 1-39 of 39 Last

Menu Item	Details	Menu Path	Seq Nbr.	Portal Object Name
Portal Expire		Root > Portal Objects >		PORTAL_EXPIRE
My Page		Root > Portal Objects > Homepage > Tabs >		DEFAULT
Design Helper		Root > Portal Objects > Navigation Collections > XML Publisher > Complete Setup >	18	ADMN_S200601131641106358836634
Report Category		Root > Portal Objects > Navigation Collections > XML Publisher > Complete Setup >	17	ADMN_S200601131640216688131351
Content Library		Root > Portal Objects > Navigation Collections > XML Publisher > Define XMLP Reports >	9	ADMN_S200601131637299978942228
Report Definition		Root > Portal Objects > Navigation Collections > XML Publisher > Define XMLP Reports >	5	ADMN_S200601131637094435254982
Translations		Root > Portal Objects > Navigation Collections > XML Publisher > Define XMLP Reports >	10	ADMN_S200601131637448169194616
Data Source		Root > Portal Objects > Navigation Collections > XML Publisher > Register Data Sources >	4	ADMN_S200601131635094344920194

Menu Security page: Objects tab

Use this page to view a list of folders or content references that a user ID, role, or permission list can access.

Note. The Seq Nbr (sequence number) column is hidden in this illustration, and because of the large number of objects, not all are shown.

Object Security

Select the level of object security for the menu information that you want to see.

User, Role, and Permission List

Select a type of object security, and then select an object.

You can view the complete menu folder structure that is available in the navigation (or hidden, but accessible from another page) for the selected security value.

Warning! To prevent unexpected errors and inordinately long retrieval times, do *not* select the *ALLPAGES* permission list in conjunction with the *Show All* folder option. Always narrow your folder options if you select the *ALLPAGES* permission list.

Folder Visibility

Select the level of visibility for the menu information that you want to see.

Show All

Select to view all objects.

Note. To avoid unexpected errors and inordinately long retrieval times, Oracle recommends that you not select this option if you select the *ALLPAGES* permission list in the Object Security section.

Choose Top Level

Select a level-one folder. The Menu Security grid displays all content references and their complete paths below this level that are accessible by the selected object security level.

Show Top Level Only

Select to display only the level-one folders that are accessible by the object security type.

Permission Lists

If either the Role object security option or the Show Top Level Only folder visibility option is selected, then this scroll area appears, displaying the permission lists that are associated with each menu item.

Building Registry Search Indexes

This section discusses how to:

- Associate keywords with content references.
- Build search indexes for PeopleSoft applications.

Note. If you use PeopleSoft Application Portal, registry search implementation is different. Enterprise Search documentation supersedes PeopleTools Search documentation.

See PeopleSoft Application Portal 9 PeopleBook: Application Portal Technology, Configuring the Application Portal Search Infrastructure

Pages Used to Build Registry Search Indexes

<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 Add Content Reference or Edit.	Use the attribute KEYWORD to associate keywords with a content reference. Any search for these keywords returns the content reference as a search result.
Build Search Index	PORTAL_SRCH_IDX	PeopleTools, Portal, Build Registry Search Index	Build search indexes for portal applications.

Associating Keywords with Content References

Access the Content Ref Administration page (PeopleTools, Portal, Structure and Content Click the Edit link for the appropriate content reference.)

The screenshot shows a web form titled "Content Reference Attributes". It includes the following elements:

- Name:** A text input field containing the text "KEYWORD".
- Label:** An empty text input field.
- Attribute value:** A text input field containing the text "skill".
- Translate:** A checkbox that is checked, located to the right of the Name field.
- Delete:** A button located to the right of the Translate checkbox.
- Add:** A button located at the bottom left of the form.

Content Ref Administration page – Content Reference Attributes

The search index derives keywords from the content reference labels and long descriptions. You can add additional keywords to a content reference by adding portal KEYWORD attributes. This is very useful if synonym-type keywords exist that are not mentioned in the title or description, for example, *P.O.* versus *Purchase Order*.

Note. Attribute values must be separated with a comma and a space.

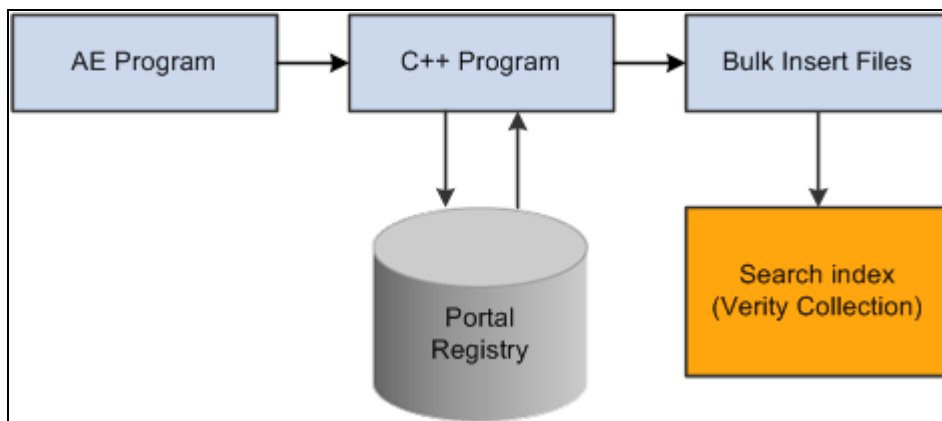
Building Search Indexes for PeopleSoft Applications

Access the Build Search Index page (PeopleTools, Portal, Build Registry Search Index).

You can build a search index from the portal administration pages and schedule the build process using PeopleSoft Process Scheduler. Each time that you build the search index, the index is entirely rebuilt, and any previous entries are overwritten.

Note. Only content references can be indexed.

The process for building a search index for a portal application is fairly easy because data in the portal registry is already in a known format and this data is used to build the search index. When you build a search index from the portal administration pages, an Application Engine program calls a C++ program, PORTAL_INDEX, which reads from the portal registry. The C++ program then generates the bulk insert files to create the search index. The following illustration represents this process:



Building a search index process

To build a search index:

1. Select or add a run control ID.

You can use any run control ID; however, you may want to use a portal name to associate the run control ID with the registry for which you are building the search index.

2. Select the All Installed Languages check box to build the index for all installed languages.
3. Select the language code for which you want to build the search index.

The default language code is the base language.

4. Click Run to launch the build process.

PeopleSoft Process Scheduler initiates an Application Engine program named PORTAL_INDEX. Click the Process Monitor link to view the status of the index build. Click Report Monitor to review output.

See Also

PeopleTools 8.52: System and Server Administration, "Configuring Verity Search and Building Verity Search Indexes"

Chapter 6

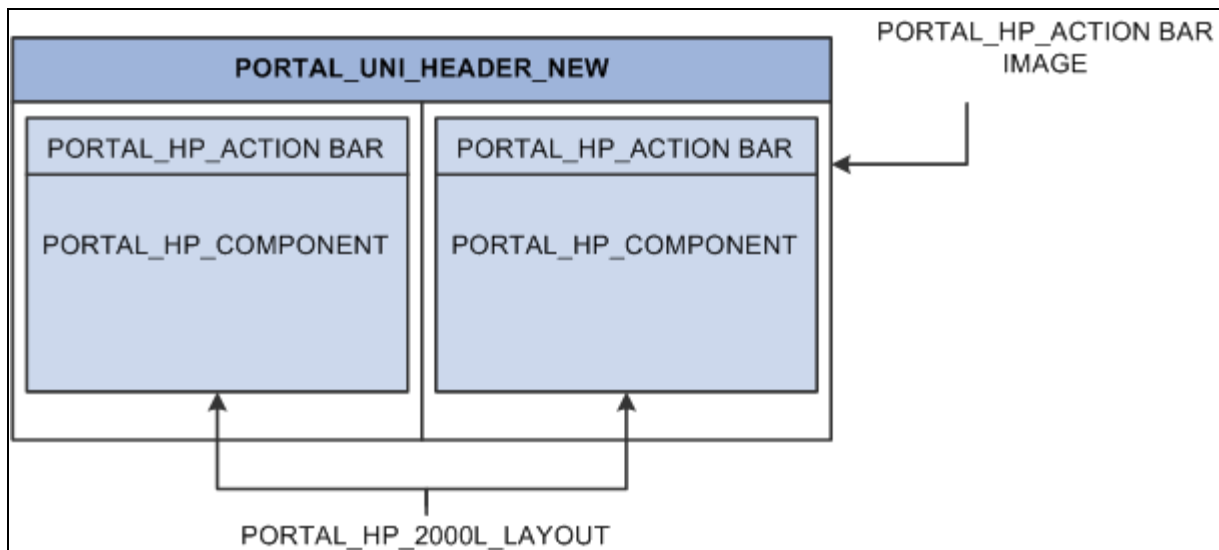
Modifying the Portal Interface

You can edit HTML objects to control the appearance of your portal. This chapter discusses how to:

- Use predefined template HTML objects.
- Use predefined homepage HTML objects.
- Use custom portal headers.
- Change link colors.
- Apply changes.

Using Predefined Template HTML Objects

The following example shows a two-column layout that is used to display two pagelets side by side. You can see which part of the page each HTML object controls. The `PORTAL_HP_USER_TEMPLATE` object controls the overall page, and the `PORTAL_HP_COMPONENT` object controls the appearance of each pagelet. Similarly, the `PORTAL_UNI_HEADER_NEW` object controls the navigation header.



HTML objects that make up a template

This table describes the HTML objects:

HTML Object	Description
PORTAL_HP_USER_TEMPLATE	Each user homepage is generated from this HTML object. Modify this object to add default components that are displayed to all users.
PORTAL_HP_3COL_LAYOUT	This is the HTML object for a three-column user homepage. The default column width is set to 33 percent for each column.
PORTAL_HP_2COL_LAYOUT	This is the HTML object for a two-column user homepage. The default column widths are set to 33 percent for column one and 66 percent for column two.
PORTAL_HP_COMPONENT	<p>Use this HTML object to create all homepage pagelets. Modify it to change a pagelet's appearance.</p> <p>Do not move, delete, or change the first or last line of this HTML definition:</p> <pre><!-- Begin Pagelet=%BIND(:6) -->. <!-- End Pagelet=%BIND(:6) --></pre> <p>These special lines must exist exactly as shown. You must also ensure that all bind variables exactly match the bind variables that are used in the associated PeopleCode.</p>
PORTAL_HP_ACTIONBAR	Use this HTML object to create the pagelet action bar. The default action bar buttons are Remove and Minimize/Restore.
PORTAL_HP_ACTIONBARIMAGE	Use this HTML object to create the Customize image button for the pagelet action bar. The image button is added only if you've defined an advanced personalization page.
PORTAL_UNI_HEADER_NEW	This HTML object controls the universal header for the portal.

With the exception of the PORTAL_UNI_HEADER_NEW HTML object, all homepage HTML objects are built in the BuildStaticHPTemplate function.

This function is located in FUNCLIB_PORTAL.TEMPLATE_FUNC.FieldFormula.

The homepage modification functionality (minimize, maximize, and remove) is located in WEBLIB_PORTAL.PORTAL_HOMEPAGE.FieldFormula.

The iScripts are IScript_HPCompRemove, IScript_HPCompMinimize, and IScript_HPCompExpand.

Using Predefined Homepage HTML Objects

This section discusses how to:

- Construct homepages.
- Construct headers.
- Construct pagelets.
- Adapt homepage and pagelet objects.

Constructing Homepages

This example shows the layout that is used for displaying a homepage with a header, pagelets, and a menu:



HTML objects that make up a homepage

This table describes the HTML objects that the system uses to construct homepages:

HTML Object	Description
PORTAL_HP_USER_TEMPLATE	This is the HTML object for the homepage tab. This object contains the iScript reference to the universal header, a bind variable for the two-column or three-column layout HTML, and binds for the Powered by PeopleSoft image.

HTML Object	Description
PORTAL_UNI_HEADER_NEW	This HTML object contains all the HTML for the header portion of a homepage tab. The HTML object is set up in <code>WEBLIB_PORTAL.PORTAL_HEADER.FieldFormula.GetUniHeaderHTML()</code> .
PT_IFRAME_HDR	This HTML object contains all of the HTML for the iFrame header.
PT_HNAV	This HTML object contains the HTML for the drop-down navigation header.
PORTAL_HP_2COL_LAYOUT	This HTML object contains all the pagelet HTML for the two-column layout (columns 1 and 2).
PORTAL_HP_2COL_LAYOUT_ACCESS	This HTML object contains all the pagelet HTML for the two-column layout (columns 1 and 2) when the homepage is in Accessibility mode.
PORTAL_HP_3COL_LAYOUT	This HTML object contains all the pagelet HTML for the three-column layout (columns 1, 2 , and 3).
PORTAL_HP_3COL_LAYOUT_ACCESS	This HTML object contains all the pagelet HTML for the three-column layout (columns 1, 2 , and 3 when the homepage is in Accessibility mode.
PORTAL_HP_COMPONENT	This is the HTML object for pagelets.

Constructing Headers

This table describes the HTML objects that the system uses to construct headers:

HTML Object	Description
PORTAL_HP_TABS	This is the HTML object for all tabs. Use this HTML object when more than one system tab exists for a user. This HTML object is set up in <code>WEBLIB_PORTAL.PORTAL_HOMEPAGE.FieldFormula.GetHPTabHTML()</code> .

HTML Object	Description
PORTAL_HP_INACTIVE_TAB	This is the HTML object for the inactive tab. The inactive tab name is linked. This HTML object is set up in WEBLIB_PORTAL.PORTAL_HOMEPAGE.FieldFormula.GetHPTabHTML().
PORTAL_HP_ACTIVE_TAB	This is the HTML object for the active tab. Only one active tab can exist. The tab name is static text. This HTML object is set up in WEBLIB_PORTAL.PORTAL_HOMEPAGE.FieldFormula.GetHPTabHTML().
PORTAL_HP_PERSONALIZE_LINKS_TD	This is the HTML object for the Personalize Content and Personalize Layout links. This HTML object is used only when more than one system tab exists for a user. This HTML object is set up in WEBLIB_PORTAL.PORTAL_HOMEPAGE.FieldFormula.GetHPTabHTML().
PORTAL_HP_PERSONALIZE_LINKS	This is the HTML object for the Personalize Content and Personalize Layout links. This HTML object is used only when one system tab exists for a user. This HTML object is set up in WEBLIB_PORTAL.PORTAL_HOMEPAGE.FieldFormula.GetHPTabHTML().
PORTAL_HEADER_ADDTOFAVORITES	This is the HTML object for the Add To Favorites link. This HTML object is set up in WEBLIB_PORTAL.PORTAL_HEADER.FieldFormula.GetUniHeaderHTML().
PORTAL_HEADER__ADDTOFAVSFORM	This is the HTML object for the Add to Favorites form. This form contains the PortalActualURL and PortalContentURL hidden fields. The Add to Favorites page uses these hidden fields to construct the content reference to the page to which you are adding a favorite (PortalContentURL) and a link back to the actual page (PortalActualURL). This HTML object is set up in WEBLIB_PORTAL.PORTAL_HEADER.FieldFormula.GetUniHeaderHTML().
PORTAL_HEADER_WL	This is the HTML object for the View Worklist link. This HTML object is set up in WEBLIB_PORTAL.PORTAL_HEADER.FieldFormula.GetUniHeaderHTML().

Constructing Pagelets

This table describes the HTML objects that the system uses to construct pagelets:

<i>HTML Object</i>	<i>Description</i>
PORTAL_HP_ACTIONBAR	This is the HTML object for the pagelet action bar. For a personalized tab, the pagelet action bar always includes a Maximize image button. It may also contain up to three more image buttons (Remove, Edit, and Help). This HTML object is not populated for a pagelet on a system tab.
PORTAL_HP_ACTIONBAR_REFRESH	This is the HTML object for the Refresh image button on the pagelet action bar.
PORTAL_HP_ACTIONBAR_REMOVE	This is the HTML object for the Remove image button on the pagelet action bar.
PORTAL_HP_ACTIONBAR_EDIT	This is the HTML object for the Edit image button on the pagelet action bar.
PORTAL_HP_ACTIONBAR_HELP	This is the HTML object for the Help image button on the pagelet action bar.
PORTAL_HP_COMPONENT	This is the HTML object for pagelets.
PT_HP2_PGLTHNDL	This is the HTML object for the pagelet label when the pagelet cannot be dragged.
PT_HP2_PGLTHNDL_DRAG	This is the HTML object for the pagelet label when the pagelet can be dragged.
PT_HP2_BANNER	This is the HTML object for the banner pagelet.
PT_HP2_PGLT_HDR_IMG	This is the HTML object for the pagelet header image.

Adapting Homepage and Pagelet Objects

This section discusses how to:

- Build portal homepage tabs by using fixed-name HTML and image objects.
- Add images to homepage tabs.

Building Portal Homepage Tabs by Using Fixed-Name HTML and Image Objects

PeopleSoft applications use fixed-name HTML and image objects to build portal homepage tabs. You can override default HTML objects and images that are used for homepage elements, such as tabs and pagelets, by adding attributes on the Content Ref Administration page.

Access the Content Ref Administration page (Select PeopleTools, Portal, Structure and Content. Click the Portal Objects link.)

To work with homepages, click the Homepages link, then the Tabs link. Click a homepage tab label.

To work with pagelets, click the Pagelets link, then a pagelet group name link. Click a pagelet label.

Content Reference Attributes region of the Content Ref Administration page

To override a default fixed-name HTML or image object:

1. In the Content Reference group box at the bottom of the page, click the Edit link for a content reference.
2. Scroll to the Content Reference Attributes group box for a content reference.
3. In the Name field, enter the name of the default object that you want to change.
4. Deselect the Translate check box.

Note. You must deselect the Translate check box for the override to be successful.

5. In the Attribute Value field, specify the new object value.

Note. The value that you specify can consist of letters, digits, and underscores (_). It cannot contain any spaces and cannot begin with a digit.

6. Click Save.

You can omit the Label field value.

As shown in the example, to replace the Powered by PeopleSoft image on the homepage tab, add a nontranslatable attribute named PT_PORTAL_POWEREDBY with a new image object name in the database.

This table shows a list of default objects that you can change for a tab:

HTML or Image Object	Value
PT_PORTAL_POWEREDBY	<i>Image name in database.</i>
PT_PORTAL_IC_REFRESH	<i>Image name in database.</i>
PT_PORTAL_IC_EDIT	<i>Image name in database.</i>
PT_PORTAL_IC_CLOSE	<i>Image name in database.</i>
PT_PORTAL_IC_COLLAPSE	<i>Image name in database.</i>
PT_PORTAL_IC_EXPAND	<i>Image name in database.</i>
PT_PGLT_HELP	<i>Image name in database.</i>
PORTAL_HP_COMPONENT	<i>HTML object name in database.</i>
PORTAL_HP_ACTIONBAR	<i>HTML object name in database.</i>
PORTAL_HP_ACTIONBAR_REFRESH	<i>HTML object name in database.</i>
PORTAL_HP_ACTIONBAR_REMOVE	<i>HTML object name in database.</i>
PORTAL_HP_ACTIONBAR_EDIT	<i>HTML object name in database.</i>
PORTAL_HP_ACTIONBAR_HELP	<i>HTML object name in database.</i>
PORTAL_HP_2COL_LAYOUT	<i>HTML object name in database.</i>
PORTAL_HP_3COL_LAYOUT	<i>HTML object name in database.</i>
PORTAL_HP_USER_TEMPLATE	<i>HTML object name in database.</i>

Adding Images to Homepage Tabs

You can specify up to 15 extra images for a homepage by using %BIND variables on PORTAL_HP_COMPONENT and PORTAL_HP_ACTIONBAR_xxx HTML objects. Homepage generation logic replaces %BIND12 to %BIND26 on the homepage tab by the images that are specified as attributes.

Note. Bind variables for images and all HTML components (tab and pagelet) should be nontranslatable. To make an item nontranslatable, deselect the Translate check box.

The following %BIND variables can be replaced by the Homepage Generation service (specify bind variables in PeopleSoft Application Designer):

Bind Variable	Object Replaced
%BIND(:1)	The PT_PORTAL_POWEREDBY image (Powered By PeopleSoft image).
%BIND(:2)	The result of a function call to %Request.Full URI.
%BIND(:3)	The result of a function call to %Request.GetHelpURL('PERSHOEPAGE').
%BIND(:4)	The result of a function call to refreshonExpire.
%BIND(:5)	The PT_PGLT_HELP image.
%BIND(:6)	The result of a function call to %Request.GetHelpURL(&Pgl.HelpID).
%BIND(:7)	The PT_PORTAL_IC_EDIT image (edit image for a pagelet).
%BIND(:8)	The result of a function call to SetRefreshPage.
%BIND(:9)	The PT_PORTAL_IC_CLOSE image.
%BIND(:10)	The PT_PORTAL_IC_COLLAPSE image.
%BIND(:11)	The PT_PORTAL_IC_EXPAND image.
%BIND(:12) to %BIND(:26)	The value of the tab attribute IMAGE_BIND_X, where X ranges from 12 to 26.
%BIND(:28)	The PT_PORTAL_IC_REFRESH image.

This code sample shows how to specify custom %BIND images to display custom images for the PORTAL_HP_COMPONENT HTML object:

```

<!-- Begin Pagelet=%BIND(:5) -->
<!-- PageletState=MAX -->
<TABLE id="" CLASS="PTPAGELET" WIDTH="100%">
  CELLPADDING="0" CELLSPACING="0" BORDER="1">
<TR>
  <TD>
    <!--Image 12 will be displayed here.-->
    <IMG SRC="%BIND(:12)" NAME="anything you like">
    <!--Image 13 will be displayed here.-->
    <IMG SRC="%BIND(:13)" NAME="anything you like">
    <!--Image 14 will be displayed here.-->
    <IMG SRC="%BIND(:14)" NAME="anything you like">
    <!--Image 26 will be displayed here.-->
    <IMG SRC="%BIND(:26)" NAME="anything you like">
  </TD>
</TR>
<TR>
  <TD>
    <TD CLASS="PTPAGELETHEADER">
      %BIND(:1)
      %BIND(:2)</nobr>
    </TD>
  </TR>
<TR>
  <TD CLASS="PTPAGELETBODY" WIDTH="100%">
    <Pagelet Name="%BIND(:2)">
      <Source Pagelet="%BIND(:4)" href="%BIND(:3)" />
    </Pagelet>
  </TD>
</TR>
</TABLE>
<BR>
<!-- End Pagelet=%BIND(:5) -->

```

The HTML objects PORTAL_HP_2COL_LAYOUT and PORTAL_HP_3COL_LAYOUT HTML can be modified similarly to display extra images.

The Homepage Generation service replaces the image that is specified by the attribute IMAGE_BIND_12 in place of bind variable %BIND(:12), and so on.

Warning! Pagelet HTML is parsed by the portal servlet. Be sure that the HTML is properly formatted and that none of the matching tags are missing. Improperly formatted HTML and missing tags may result in unexpected web server behavior.

Using Custom Portal Headers

The following list presents custom headers that you can use with the portal:

- PortalServletURI
- PortalURI
- PortalActualURL
- PortalContentURL
- PortalContentProvider

- PortalRegistryName
- PortalTargetFrame
- PortalPagelet

These are valid parameters that you can pass with the PeopleCode GetParameter method. PortalContentURL retrieves the registered URL, while PortalActualURL retrieves the entire URL, including any additional query string parameters.

Changing Link Colors

Use the PSSTYLEDEF style sheet to change link colors.

To change the link color:

1. Open the PSSTYLEDEF style sheet in PeopleSoft Application Designer.
2. Locate and open the PSHYPERLINK class.
 - a. Modify the attributes as appropriate.
 - b. On the Properties page, modify the pseudo classes.
 - c. Specify default font attributes for <A> and <TD> in your own style sheet or in an embedded style sheet at the top of PORTAL_UNI_HEADER_NEW.

This applies font attributes to text that has no class associated with it. For example:

```
<STYLE TYPE="text/css">
  td { font-family:arial; font-size:9pt; }
  a:visited { font-family:arial; font-size:9pt; color: #5c93ae; ⇒
text-decoration:underline }
  a:link { font-family:arial; font-size:9pt; color: #5c93ae; ⇒
text-decoration: underline }
  a:hover { font-family:arial; font-size:9pt; color: #5c93ae; ⇒
text-decoration: underline }
  a:active { font-family:arial; font-size:9pt; color: #5c93ae; ⇒
text-decoration: underline }
</STYLE>
```

Note. Oracle recommends that you do not alter the delivered PSSTYLESHEET style sheet definition. Make a copy of PSSTYLESHEET and modify the copy.

See Also

PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide, "Creating Style Sheet Definitions"

Applying Changes

To immediately view changes that you make to a page, click the browser Refresh button. This reloads the updated page, instead of waiting for the old page to expire and the new page to be loaded.

These functions, which are found in FUNCLIB_PORTAL.TEMPLATE_FUNC FieldFormula, are also available to force a refresh:

<i>Function</i>	<i>Description</i>
ForceRefreshPage (&url)	Refreshes the default tab for the current user.
&myPage = URLEncode ("?tab=DEFAULT&user=" %userid);	Forces the page that is identified by the parameter to refresh. The parameter is the escaped query string of the page that is being refreshed, including the question mark.
ForceRefreshPage (&myPage);	Refreshes the default tab for the current user.
ForceRefreshHomePageTab (&tab)	Refreshes the homepage tab for the current user with the tab name that is passed in.
ForceRefreshHomePage ()	Refreshes the default homepage tab of the current user.

Chapter 7

Working with Portal Templates

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

- Apply template types.
- Develop portal templates.

Understanding Portal Templates

The PeopleSoft portal integrates content from numerous sources and presents the merged content on a single web page in a coherent, consistent fashion that keeps users within the portal framework. A portal template tells the portal servlet what content to place on the assembled page and where each piece of content should be placed.

A portal template is HTML code. However, in addition to standard HTML tags, the portal template can include PeopleSoft tags that are used for giving instructions to the portal servlet during page assembly. A normal browser cannot interpret these special tags. The portal servlet follows instructions in the PeopleSoft tags and then removes them before passing the final page back to the user's browser.

The content in a template falls into four categories:

- HTML for the template itself.
- HTML for the main target content.
- HTML for the related target content.
- Additional HTML, such as for the navigation header.

Applying Template Types

This section provides an overview of template types and discusses how to:

- Use page-based static templates.
- Use frame-based static templates.
- Use frame-based dynamic templates.
- Use iframe-based dynamic templates.

Understanding Template Types

Each portal template is either page-based, frame-based, or iframe-based, and each template is either static or dynamic. Template types come in five possible combinations:

- Page-based static templates.

These are content references with a usage type of *HTML template* and a storage type of *Local* (in *HTML Catalog*).

- Page-based dynamic templates.

These are content references with a usage type of *HTML template* and a storage type of *Remote by URL*.

- Frame-based static templates.

These are content references with a usage type of *Frame template* and a storage type of *Local* (in *HTML Catalog*).

- Frame-based dynamic templates.

These are content references with a usage type of *Frame template* and a storage type of *Remote by URL*.

- iFrame-based dynamic templates.

These are content references with a usage type of *iFrame template* and a storage type of *Remote by URL*.

A page-based template uses HTML tables to generate a page. Content is placed in each table cell that is defined by the template. The portal servlet uses the page-assembly process to retrieve documents for each cell. The assembled page is then sent as a single unit to the user's browser.

A frame-based template uses frames, rather than HTML tables, to divide the page into parts. The portal servlet constructs the appropriate URL for each frame (<src> tag in the frameset) and sends the entire frameset to the browser. The browser then retrieves the content for each frame.

An iframe-based template uses HTML with embedded inline frames to divide the page into parts. The portal servlet constructs the appropriate URL for each section of the page and sends the HTML to the browser. The browser then retrieves the content for each section and each inline frame.

A static template is based on HTML code that is entered into the HTML area of a template content reference (a content reference that exists in the Portal Objects\Templates folder of the portal registry). Bind variables cannot be used within the HTML area. The HTML includes any HTML that is required for the template itself, plus tags that specify the URLs for template pagelets. You can view and update the contents of a static template on the portal administration pages.

A dynamic template is retrieved from the web in real time, as part of the page-assembly process, rather than stored in the database with the content reference. A dynamic template can use bind variables. You cannot view the HTML contents of the dynamic template directly on the portal administration pages. Instead, the portal administration pages enable you to identify the URL that points to the resource that provides the template content. If the dynamic template is implemented by an iScript, you can use PeopleSoft Application Designer to navigate to the record and field where the iScript exists. The content provided by the URL is the HTML that is used for the dynamic template at runtime.

See [Chapter 5, "Administering Portals," Managing Portal Objects, page 176.](#)

Using Page-Based Static Templates

This static template example combines a universal navigation header with target content. It is based on HTML tables, not frames.

```
<html>
<head>
</head>
<body>
<table>
<tr><td>
<Pagelet Name="UniversalNavigation">
  <SOURCE Node="LOCAL_NODE" href="s/WEBLIB_PORTAL.PORTAL_HEADER.FieldFormula.
    Iscript_UniHeader" />
</Pagelet>
</td></tr>
<tr><td>
<Target Name="TargetContent"/>
</td></tr>
</table>
</body>
</html>
```

This template is composed of HTML for three items:

- The template itself.

Some HTML is required for the overall template. The code for this template is represented in the preceding example by all the nonemphasized text. This HTML code remains on the assembled page that is sent to the user's browser after the page-assembly process.

- A template tag for the universal navigation header pagelet.

This is the first block of HTML code that is emphasized in the preceding example. At runtime, the portal servlet replaces the Pagelet tag with the navigation header in the template for the final assembled page.

- A template tag for target content.

At runtime, the portal servlet replaces the Target tag—the second block of emphasized HTML code in the previous example—with whatever content the user requested by clicking a link or button.

Using Frame-Based Static Templates

This is an example of a static template based on frames:

GeneralSecurity

Root > Portal Objects > Templates >

Content Ref Administration

Name:

PTNAVFRAMETPL

Author:

QEDMO

*Label:

PTNavFrameTpl

Parent Folder:

Templates

Copy object

Select New Parent Folder

Long Description:

(254 Characters)

This template is used with the 8.4x menu pagelet in non IE browsers on the homepage

Product:

PT

*Valid from date:

08/01/2001

Sequence number:

Valid to date:

Object Owner ID

PPT

PeopleTools

Usage Type:

Frame template

Creation Date:

08/01/2001

Storage Type:

Local (in HTML Catalog)

Add Content Reference

HTML Area:

```

<HTML>
<frameset border=0 framespacing=0 frameborder=no rows="65,*">
  <FRAME name=UniversalHeader scrolling=no frameborder = no noresize src=>
    <IClientComponent Name="UniversalNavigation">
      <Source Node="LOCAL_NODE" href="s/WEBLIB_PT_NAV.ISCRIPT2.FieldFormula.IScript_UniHeader_Frame?MIN=N" />
    </IClientComponent>
  </FRAME>
  <frameset border=0 frameborder=no framespacing=0 cols="195,*">
    <FRAME name=NAV frameborder=no scrolling=auto noresize src=>
      <Target Name="TargetContent"/>
    <FRAME name=TargetContent frameborder=no noresize src=>
      <IClientComponent Name="UniversalNavigation">
        <Source Node="LOCAL_NODE" href="s/WEBLIB_PT_NAV.ISCRIPT1.FieldFormula.IScript_TimeOutDrillDown" />
      </IClientComponent>
    </FRAME>
  </FRAMESET>
</FRAMESET>
</HTML>

```

Content reference for a frame-based static template

The Content Ref Administration page shows how the HTML exists within the context of the whole content reference. Note that the usage type is set to *Frame template*. In the previous example of a page-based template, which uses HTML tables, the usage type would have been set to *HTML template*.

Using Frame-Based Dynamic Templates

All dynamic frame-based templates are retrieved from a URL rather than from an HTML document. Most commonly, the URL for dynamic templates is an iScript URL. As with static templates, the template content reference must be defined as a template on the portal administration pages. However, instead of including specific HTML content, the dynamic template references an iScript. The iScript is associated with a specified field in a specified record.

This example shows a frame-based dynamic template named PT_85X_FRAMES_TEMPLATE in the Portal Objects, Templates folder:

GeneralSecurity

Root > Portal Objects > Templates >

Content Ref Administration

Name PT_85X_FRAMES_TEMPLATE
*Label 8.50 frames template
Long Description 8.50 frames template (254 Characters)
Product PT
Sequence number
Owner ID PPT PeopleTools
Usage Type Frame template
Storage Type Remote by URL

Author PTDMO
Parent Folder Templates
Copy object
Select New Parent Folder
*Valid from date 10/04/2008
Valid to date
Creation Date 10/04/2008

Add Content Reference

URL Information

*Node Name LOCAL_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_Frames_Template
Additional Parameters
Example: name1=value1&name2=value2

Producer
Portlet

Producer Details

IWC Message Events

Content Reference Attributes

Name
Label
Attribute value

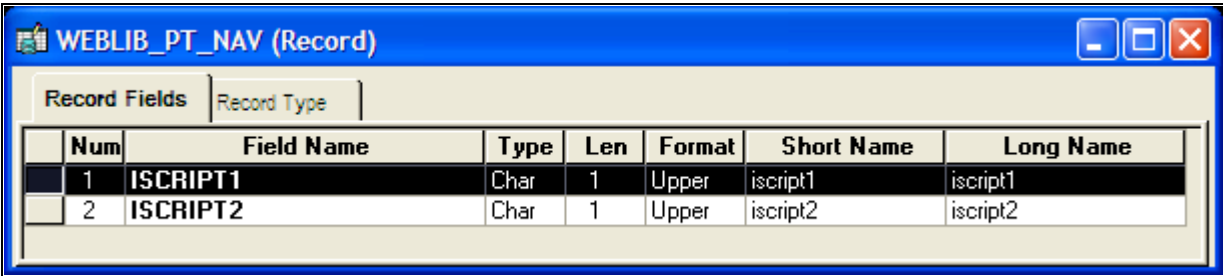
Delete
☒ Translate
Attribute Information

Add

Defining a frame-based dynamic template on the portal administration pages

The content reference storage type is set to *Remote by URL*. This setting is required for dynamic templates. Additionally, no HTML area displays the associated HTML code for the template, as you would expect with a static template. Instead, when the URL type for this example is set to *PeopleSoft Script*, an iScript parameters area appears on the page, in which you can define the record name, field name, PeopleCode event name, and PeopleCode function name that specify the iScript to use for the template.

The iScript that dynamically generates the template is located in the WEBLIB_PT_NAV record in the ISCRIP1 field, as shown in this example:

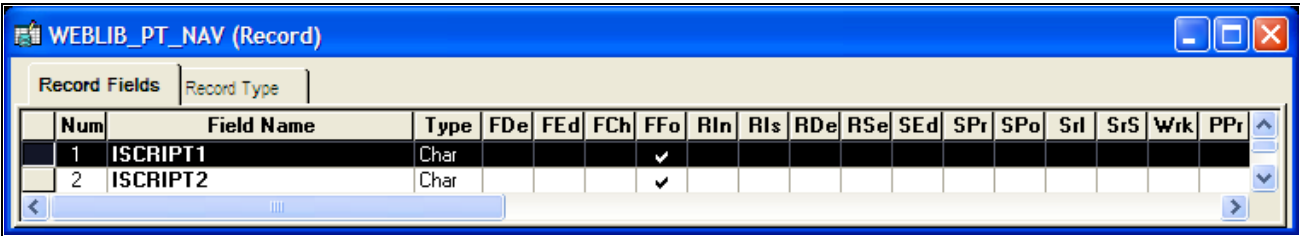


Num	Field Name	Type	Len	Format	Short Name	Long Name
1	ISCRIP1	Char	1	Upper	iscript1	iscript1
2	ISCRIP2	Char	1	Upper	iscript2	iscript2

WEBLIB_PT_NAV record

Additionally, the iScript function is associated with the FieldFormula PeopleCode event. The PeopleCode function name is IScript_PT_Frames_Template.

To view the iScript on which this dynamic template is based, open the record (WEBLIB_PT_NAV) in PeopleSoft Application Designer and select the appropriate field (ISCRIP1), as shown in this example:

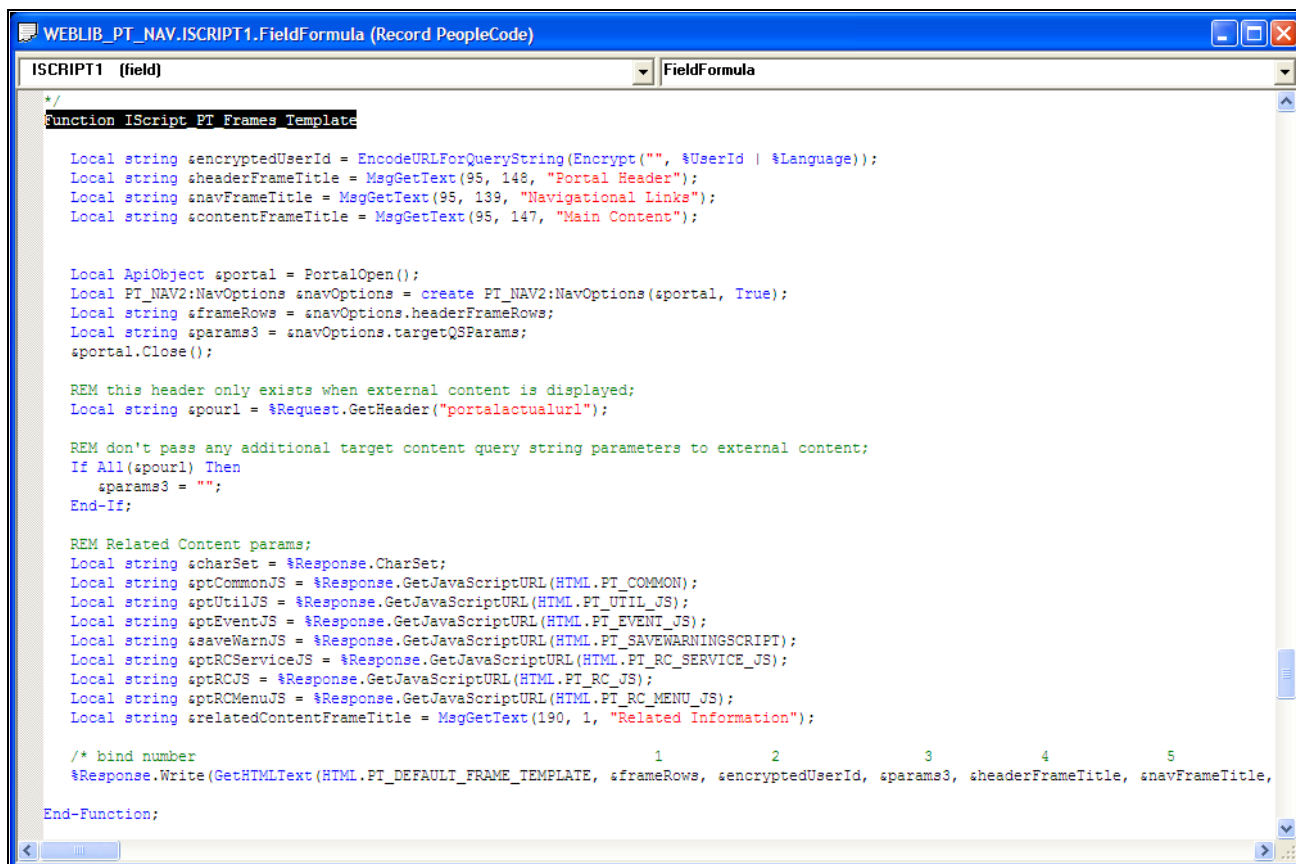


Num	Field Name	Type	FDe	FEf	FCh	FFo	RIn	RIs	RDe	RSe	SEd	SPr	SPo	Srl	SrS	Wrk	PPr
1	ISCRIP1	Char				✓											
2	ISCRIP2	Char				✓											

Opening the record and field associated with a dynamic template

Once you open the appropriate record and field in PeopleSoft Application Designer, you can view the FieldFormula PeopleCode. The iScript is referenced by the PeopleCode function name in the template's definition. An easy way to find the exact reference is to copy the PeopleCode function name from the template definition and paste it into the Find dialog box of the PeopleCode editor once you open the corresponding record and field.

You now can see the code of the iScript that is referenced by the portal template, as shown in this example:



The iScript upon which the dynamic template is based

The following code calls an HTML object called PT_DEFAULT_FRAME_TEMPLATE from the HTML catalog and passes the object bind variables:

```

%Response.Write(GetHTMLText(HTML.PT_DEFAULT_FRAME_TEMPLATE, &frameRows, &encryptedUserId, &params3, &headerFrameTitle,
    &navFrameTitle, &contentFrameTitle, &relatedContentFrameTitle, &charSet, &ptCommonJS, &ptUtilJS, &ptEventJS, &ptRCMenuJS,
    &ptRCServiceJS, &ptRCJS, &saveWarnJS));
  
```

The resulting HTML code forms the basis for the dynamic portal template at runtime. You can open the HTML object PT_DEFAULT_FRAME_TEMPLATE in PeopleSoft Application Designer, as shown in this example, and see where the bind variable is passed in (note that this is a frame-based template, as Frame tags are present):

```

<html dir="%Direction" lang="%LanguageISO">
<head>
<meta http-equiv='content-type' content='text/html; charset=%bind(:8)'\>
<script type="text/javascript" src="%bind(:9)"/></script>
<script type="text/javascript" src="%bind(:10)"/></script>
<script type="text/javascript" src="%bind(:11)"/></script>
<script type="text/javascript" src="%bind(:15)"/></script>
<script type="text/javascript" src="%bind(:12)"/></script>
<script type="text/javascript" src="%bind(:13)"/></script>
<script type="text/javascript" src="%bind(:14)"/></script>
</head>
<frameset border=0 frameborder=no framespacing=0 rows="%Bind(:1),*" title="%Bind(:6)">
  <frame name=UniversalHeader title="%Bind(:4)" scrolling=no frameborder = no noresize src=
    <IClientComponent Name="UniversalNavigation">
      <Source Node="LOCAL_NODE" href="/s/WEBLIB_PT_NAV.ISCRIPT1.FieldFormula.IScript_UniHeader_Frame?c=%Bind(:2)"/>
    </IClientComponent>
  </frame>
  <frameset ID=SubFrame border=0 frameborder=no framespacing=0 cols="195,*" title="%Bind(:6)">
    <frame name=NAV title="%Bind(:5)" frameborder=no scrolling=auto noresize src=
      <IClientComponent Name="Nav">
        <Source Node="LOCAL_NODE" href="/s/WEBLIB_PT_NAV.ISCRIPT1.FieldFormula.IScript_PT_NAV_INFRAME?c=%Bind(:2)"/>
      </IClientComponent>
    </frame>
    <frameset ID=TargetContentFrame border=0 frameborder=no framespacing=0 rows="75%,*">
      <frame name=TargetContent title="%Bind(:6)" ID=TargetContent scrolling=auto onload="ptrc.onLoadTC()" src=
        <Target Name="TargetContent" params="%Bind(:3)"/>
      </frame>
      <frame name=RelatedContent title="%Bind(:7)" ID=RelatedContent scrolling=no onload="ptrc.onLoadRC()" src=
        <RelatedContent Name="RelatedContent" params="%Bind(:3)"/>
      </frame>
    </frameset>
  </frameset>
</frameset>
</html>

```

Using bind variables with a dynamic template

See Also

Chapter 5, "Administering Portals," Administering Content References, page 143

Using iFrame-Based Dynamic Templates

All dynamic iframe-based templates are retrieved from a URL rather than from an HTML document. Most commonly, the URL for dynamic templates is an iScript URL. As with other templates, the template content reference must be defined as a template on the portal administration pages. However, instead of including specific HTML content, the dynamic template references an iScript. The iScript is associated with a specified field in a specified record.

Important! HTML templates are deprecated as of PeopleTools 8.50. PeopleSoft applications require iframe templates to function as designed. Features that are unsupported in HTML templates experience significant or total loss of functionality; such features include:

- Modal windows used for prompts, secondary pages, and grid zoom.
- Main Menu drop-down navigation.
- Related content.
- WorkCenter pages.
- Global search header.

This example shows an iframe-based dynamic template named `DEFAULT_TEMPLATE` in the Portal Objects, Templates folder:

General
Security

[Root](#) > [Portal Objects](#) > [Templates](#) >

Content Ref Administration

Name DEFAULT_TEMPLATE
*Label 8.50 default template

Author PTDMO
Parent Folder Templates
Copy object
Select New Parent Folder

Long Description 8.50 default template
(254 Characters)

Product PT
Sequence number

Owner ID PPT PeopleTools
*Valid from date 05/18/2000
Valid to date

Usage Type Inline 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 ISCRIPT1

*PeopleCode Event Name FieldFormula
*PeopleCode Function Name IScript_PT_NAV_TPL_FRAME

Additional Parameters
Example: name1=value1&name2=value2

Producer
Portlet

[Producer Details](#)

[IWC Message Events](#)

Content Reference Attributes

Name
Label
Attribute value

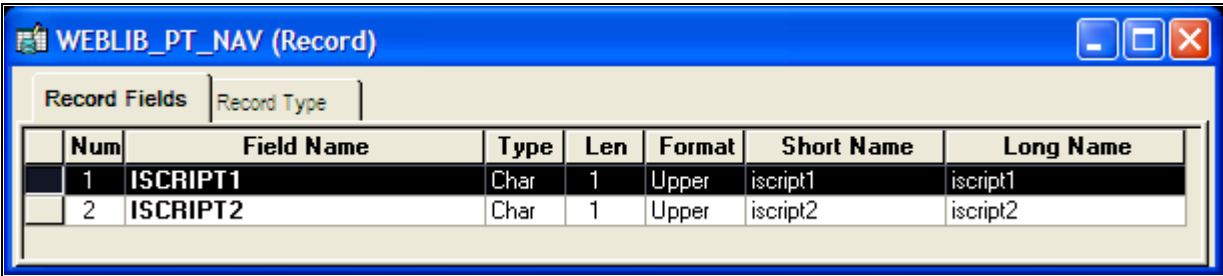
☒ Translate
[Attribute Information](#)

Delete
Add

Defining an iframe-based dynamic template on the Content Ref Administration page

The content reference storage type is set to *Remote by URL*. This setting is required for dynamic templates. Additionally, no HTML area displays the associated HTML code for the template, as you would expect with a static template. Instead, when the URL type for this example is set to *PeopleSoft Script*, an iScript parameters area appears on the page, in which you can define the record name, field name, PeopleCode event name, and PeopleCode function name that specify the iScript to use for the template.

The iScript that dynamically generates the template is located in the WEBLIB_PT_NAV record in the ISCRIPT1 field, as shown in this example:

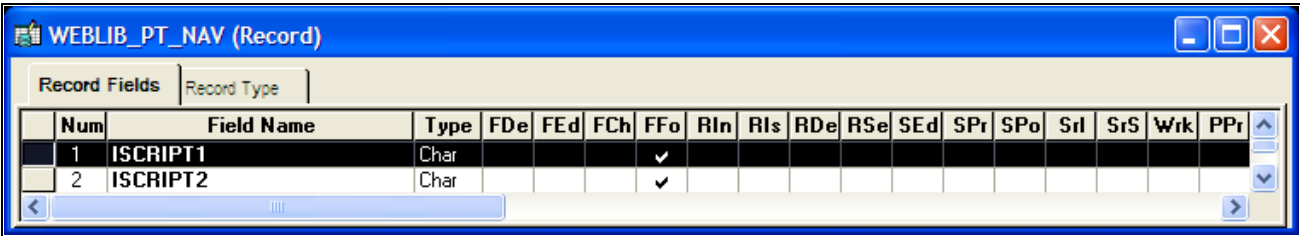


Num	Field Name	Type	Len	Format	Short Name	Long Name
1	ISCRIP1	Char	1	Upper	iscript1	iscript1
2	ISCRIP2	Char	1	Upper	iscript2	iscript2

WEBLIB_PT_NAV record

Additionally, the iScript function is associated with the FieldFormula PeopleCode event. The PeopleCode function name is IScript_PT_Frames_Template.

To view the iScript on which this dynamic template is based, open the record (WEBLIB_PT_NAV) in PeopleSoft Application Designer and select the appropriate field (ISCRIP1), as shown in this example:

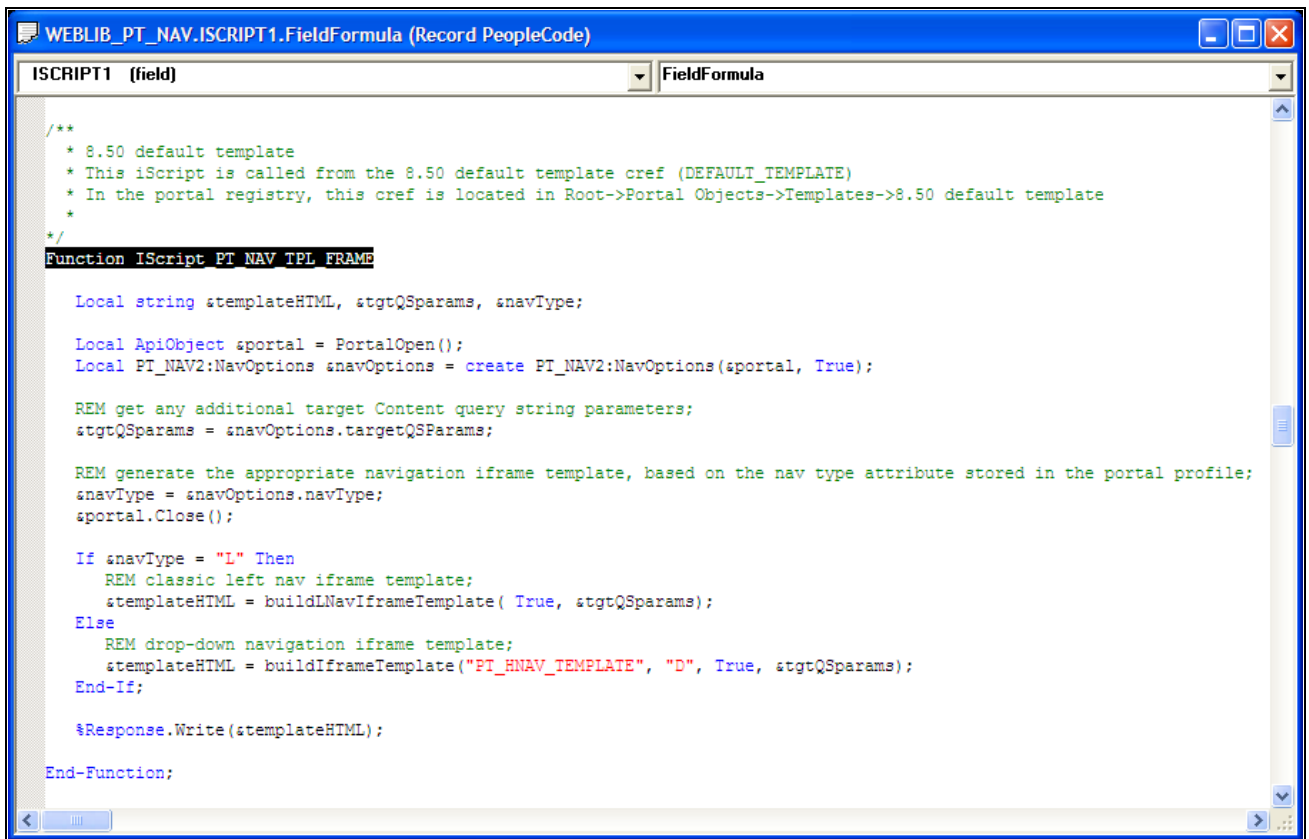


Num	Field Name	Type	FDe	FEf	FCh	FFo	RIn	RIs	RDe	RSe	SEd	SPr	SPo	Srl	SrS	Wrk	PPr
1	ISCRIP1	Char				✓											
2	ISCRIP2	Char				✓											

Opening the record and field associated with a dynamic template

Once you open the appropriate record and field in PeopleSoft Application Designer, you can view the FieldFormula PeopleCode. The iScript is referenced by the PeopleCode function name in the template's definition. An easy way to find the exact reference is to copy the PeopleCode function name from the template definition and paste it into the Find dialog box of the PeopleCode editor once you open the corresponding record and field.

You now can see the code of the iScript that is referenced by the portal template, as shown in this example:



The iScript upon which the dynamic iframe template is based

The following code calls the PT_IFRAME_TEMPLATE HTML object from the HTML catalog and passes the object bind variables:

```

Return GetHTMLText(HTML.PT_IFRAME_TEMPLATE, &charSet, &requiredToolsSS, &sys⇒
DefaultSS, &domainJS, &expireJS, &ptCommonJS, &ptEventJS, &ptUtilJS, &ptAjaxJS,⇒
&crossDomainJS, &saveWarnJS, &iframeHdrJS, &iframeJS, &rcJS, &navHeader.HTML,⇒
&nav2JS, &ptPopupJS, &params1, &encryptedUserId, &rcMenuJS, &rcHTML1, &tgtIframe⇒
Title, &rcServiceJS, &cssOverride);

```

The resulting HTML code forms the basis for the dynamic portal template at runtime. You can open the PT_IFRAME_TEMPLATE HTML object in PeopleSoft Application Designer, as shown in this example, and see where the bind variable is passed in (note that this is an iframe-based template, as <iframe> tags are present):

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html dir="%LanguageISO" 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">
%bind(:4)
%bind(:5)
<script type="text/javascript" src="%bind(:6)"></script>
<script type="text/javascript" src="%bind(:7)"></script>
<script type="text/javascript" src="%bind(:8)"></script>
<script type="text/javascript" src="%bind(:9)"></script>
<script type="text/javascript" src="%bind(:10)"></script>
<script type="text/javascript" src="%bind(:11)"></script>
<script type="text/javascript" src="%bind(:17)"></script>
<script type="text/javascript" src="%bind(:12)"></script>
<script type="text/javascript" src="%bind(:23)"></script>
<script type="text/javascript" src="%bind(:20)"></script>
<script type="text/javascript" src="%bind(:14)"></script>
%bind(:24)
</head>
<body class="PSPAGE" id="ptifrmtemplate">
  <div id="ptifrmhdr">
    <pagelet name="UniversalNavigation">
      <source node="LOCAL_NODE" href="/s/WEBLIB_PT_NAV.ISCRIPT1.FieldFormula.IScript_UniHeader_Frame?c=%Bind(:19)" />
    </pagelet>
  </div>
  <div id="ptifrmsbarcollexp" class="PTPAGELETHEADER">
    %bind(:15)
  </div>
  <div id="ptifrmcontent">
    <div id="ptifrmsidebar">
      <div id="ptifrmnav" class="PTPAGELET PTNAVBACKGROUND">
        <pagelet name="nav">
          <source node="LOCAL_NODE" href="/s/WEBLIB_PT_NAV.ISCRIPT1.FieldFormula.IScript_PT_NAV_INFRAME?c=%Bind(:19)" />
        </pagelet>
        <div id="ptnav2block">&nbsp;</div>
      </div>
      <script type="text/javascript" src="%bind(:16)"></script>
    </div>
    <div id="ptifrmtarget" class="ptifrmtgtstd">
      <iframe id="ptifrmtgtframe" name="TargetContent" title="%bind(:22)" frameborder="0"
        scrolling="auto" onload="ptcro.onLoadIC()" src=><target Name="target" params="%bind(:18)" />
      </iframe>
    </div>
    %bind(:21)
    <script type="text/javascript" src="%bind(:13)"></script>
  </div>
</body>
</html>

```

PT_IFRAME_TEMPLATE HTML definition showing inline frames

Inline frame-based, or iframe-based, templates enable you to embed a frame within an HTML document rather than as a frameset. You can use the `iframe` `<source>` attribute to load content into the template. PeopleSoft applications use this attribute for target content only. The header and navigation content are in the HTML document itself as you see in the preceding PT_IFRAME_TEMPLATE HTML definition:

Note the following points about the standard iframe template:

- The Usage Type is set to *Inline frame template*; this is a requirement for *all* iframe-based templates.
- The `iframe` HTML definition supports only the `<pagelet>` and `<target>` meta-tags.

Important! The `IClientComponent` meta-tag is not supported.

- 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 facilitates the header and navigation working together.
- When you navigate 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 more than one form exists 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 may 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, conflicting styles may be 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:

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

Source

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

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

Contents must be escaped. Ampersands must be written as *&*. 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.

Attributes	Name. Used to identify the component in a comment on the final assembled page.
Elements	None.
Contents	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 for which 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>Node</i>	<i>URI</i>	<i>Sample Match</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 the application posts 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 8.52 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 8

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 only when you select the Allow Public Access check box in the on 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

PeopleTools 8.52: System and Server Administration, "Setting Application Server Domain Parameters," Cache Settings

[Chapter 9, "Configuring the Portal Environment," Configuring Portal Security, page 247](#)

PeopleTools 8.52: 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"> • <i>user</i>: The portal maintains the cached target content independently for each user. • <i>role</i>: The portal maintains a single cache of the target content for all users in a role. • <i>application</i>: The portal maintains a single cache of the target content for all users who are using an application.
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 7, "Working with Portal Templates," page 203](#)

[Chapter 9, "Configuring the Portal Environment," Configuring Caching, page 258](#)

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"> <i>user</i>: The portal maintains the cached pagelet independently for each user. <p>Note. 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 whenever possible is strongly recommended.</p> <ul style="list-style-type: none"> <i>role</i>: The portal maintains a single cache of the pagelet for all users in a role. <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, 1000 copies of the pagelet will be stored in cache. Each of these pagelet copies has the associated time to live (TTL) variable. Each time the system builds the user's homepage, it reviews the TTL to determine whether the time has come to refresh the pagelet. When a TTL time expires, that copy of the meta pagelet is invalidated and refreshed from the application server. When each user's copy expires, his or her copy will be updated with the latest version.
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 that you want to cache.

Select 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"> • <i>private</i>: The portal maintains the cached pagelet independently for each user. <p>Note. 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"> • <i>role</i>: The portal maintains a single cache of the pagelet for all users in a role. • <i>public</i>: The portal maintains a single cache of the pagelet for all users who are using an application.
<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 Application 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 current; 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 the Caching of Pagelets

You can enable 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, you need 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 selecting a check box on the pagelet's content reference administration page.

See Also

[Chapter 5, "Administering Portals," Administering Content References, page 143](#)

[Chapter 12, "Developing and Registering Pagelets," Managing Pagelet Attributes, page 507](#)

[Chapter 8, "Using Portal Caching Features," Understanding Caching Scope, page 222](#)

PeopleTools 8.52: PeopleSoft Applications User's Guide

PeopleSoft Application Portal 9.2: Application Portal 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:

Cache Portal Objects

Select to enable metadata caching.

Metadata caching is enabled by default. You must restart the web server or wait for the stale interval to expire when you change:

- The default portal for a database.
- The default local node for a database.
- The remote node to local or a local node to remote.
- Templates that are associated with a content reference, node, or portal.
- HTML within a template.

Note. 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.

Cache Stale Interval

Specify the number of seconds that should pass before the portal refreshes the metadata cache.

The default value of this property is *86400* (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.

Cache Purge All Hit Count

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.

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.

The default value of this property is *1000*. Set to *-1* to disable the feature.

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 9, "Configuring the Portal Environment," Configuring Caching, page 258](#)

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 9, "Configuring the Portal Environment," Configuring Caching, page 258

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 in which multiple users access the same computer.

These include kiosk-type environments, as well as corporate hoteling environments, in which 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.

When 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 in which you are using web server-based navigation caching.

See Also

Chapter 9, "Configuring the Portal Environment," Configuring Caching, page 258

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 need 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 deselected, regardless of whether they're listed in the grid.

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

See Also

[Chapter 9, "Configuring the Portal Environment," Configuring Caching, page 258](#)

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 *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 signs in again, 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 8, "Using Portal Caching Features," Controlling Navigation Pagelet Caching, page 231](#).

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 that 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 9, "Configuring the Portal Environment," Configuring Portal Security, page 247](#)

[Chapter 9, "Configuring the Portal Environment," Configuring Caching, page 258](#)

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 when 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 8, "Using Portal Caching Features," Understanding Caching Scope, page 222.](#)

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.

Disabling Browser-Based Caching

To disable browser-based caching:

1. Select PeopleTools, Personalization, Personalization Options.
2. Select *PeopleTools (PPTL)*.
3. Remove the *METAXP* personalization definition.
4. Click Save.

Disabling Homepage Caching

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. Deselect the Cache Homepage check box for the appropriate browser.
5. Click Save.

See Also

Chapter 9, "Configuring the Portal Environment," Configuring Caching, page 258

Chapter 9

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 discussed in this chapter. A complete listing that maps PeopleTools release 8.43 and earlier portal settings to current web profile settings is documented in this PeopleBook.

See Also

[Chapter 9, "Configuring the Portal Environment," Configuring Web Profiles, page 238](#)

[Appendix A, "Understanding Changes in Portal Configuration Settings," page 695](#)

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 a URL, 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 among 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 9, "Configuring the Portal Environment," Configuring Cookie Rules, page 257.](#)

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 specify a base-level authentication domain during the setup of every PeopleSoft application with which your portal interacts. The authentication domain must be identical and be stored on the web server of each application.

See *PeopleTools 8.52 Installations* support.oracle.com.

You can define an optional extended authentication domain in your portal 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* would be valid.

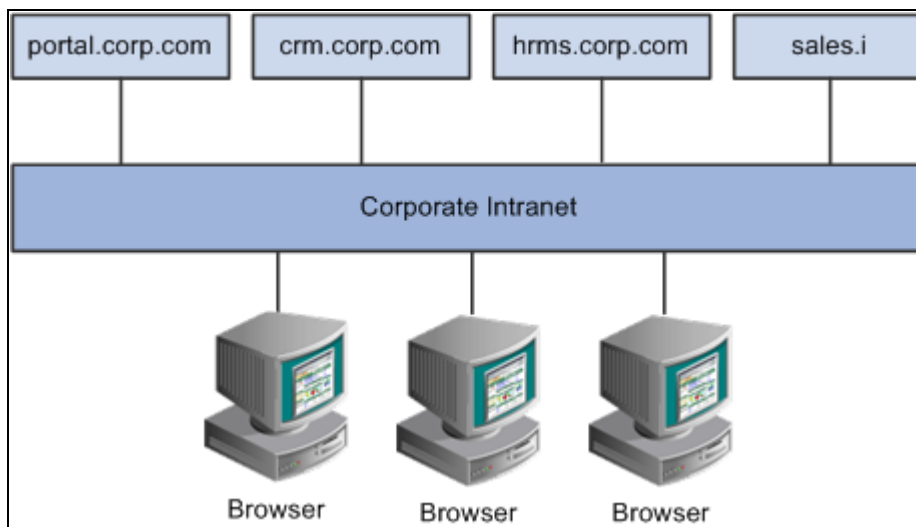
Note. If you defined a base-level or extended authentication domain, you must use it in all URLs that you specify in your portal. For example, if your authentication domain is *.mydomain.com*, then instead of using the URL <http://mymachine:8080/pshome/signon.html>, you must use the URL <http://mymachine.mydomain.com:8080/pshome/signon.html>.

You specify the extended authentication domain on General tab of the Web Profile Configuration page.

See [Chapter 9, "Configuring the Portal Environment," Configuring General Portal Properties, page 242](#).

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

PeopleTools 8.52: System and Server Administration, "Working with Oracle WebLogic"

PeopleTools 8.52: System and Server Administration, "Working with IBM WebSphere"

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 appearance and character.
- Configure custom properties.
- View web profile history.

Understanding Web Profiles

A web profile is a specific configuration of portal-related properties that control or affect the functionality of your PeopleTools portal. These predefined web profiles provide several configurations for use with your PeopleTools portal:

- *DEV*

This 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.

- *KIOSK*

This 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.

- *PROD*

This web profile uses the settings that are most commonly needed in a production environment that authenticates users.

- *TEST*

This web profile uses the same settings as the DEV web profile, except that fewer trace and debug properties are enabled.

Additional web profiles might be included as part of PeopleSoft Application Portal or an application portal pack.

You can modify the property settings of any web profile via the Web Profile Configuration component, which includes pages for these types of properties:

- General
- Portal security
- Virtual addressing
- Cookie rules
- Caching
- Trace and debug options
- Appearance and character
- Custom properties.

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 which the portal servlet can use to gain secure access to the web profile.

The default web profile is PROD.

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 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 will receive an error message about incorrect site configuration on the sign-in page.

You can unlock the web profile account by deselecting 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 *PeopleTools 8.52: 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 no profile exists 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 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\web_server\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

Change these values to specify a different user ID and password for the portal servlet to use to access the web profile. You must encrypt the password by using PSCipher.

To encrypt the user ID or password:

1. Run the PSCipher.bat file:

See Using PSCipher to Encrypt Text, *PeopleTools 8.52: Security Administration*

Note. On a UNIX machine, change the PSCipher.sh script file permissions so that you can run the program.

2. Copy the encrypted string and paste it into the configuration.properties file replacing the existing value, 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

PeopleTools 8.52: 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	Configure general portal properties.
Web Profile Configuration - Security	WEB_PROF_SECURITY	PeopleTools, Web Profile, Web Profile Configuration	Configure portal security.
Web Profile Configuration - Virtual Addressing	WEB_PROF_PROXIES	PeopleTools, Web Profile, Web Profile Configuration	Configure virtual addressing.
Web Profile Configuration - Cookie Rules	WEB_PROF_COOKIES	PeopleTools, Web Profile, Web Profile Configuration	Configure cookie rules.
Web Profile Configuration - Caching	WEB_PROF_CACHING	PeopleTools, Web Profile, Web Profile Configuration	Configure caching rules for browsers and web servers.
Web Profile Configuration - Debugging	WEB_PROF_DEBUGGING	PeopleTools, Web Profile, Web Profile Configuration	Configure trace and debug options.

Page Name	Definition Name	Navigation	Usage
Web Profile Configuration - Look and Feel	WEB_PROF_LOOKFEEL	PeopleTools, Web Profile, Web Profile Configuration	Configure appearance and character.
Web Profile Configuration - Custom Properties	WEB_PROF_CUST_PROP	PeopleTools, Web Profile, Web Profile Configuration	Configure custom properties.
Web Profile History	WEB_PROF_HISTORY	<ul style="list-style-type: none"> • PeopleTools, Web Profile, Web Profile History • PeopleTools, Web Profile, Web Profile Configuration • Click the View History link on the Web Profile - General page. 	View web profile history.

Configuring General Portal Properties

Access the Web Profile Configuration page. (Select PeopleTools, Web Profile, Web Profile Configuration).

GeneralSecurityVirtual AddressingCookie RulesCachingDebuggingLook and Feel

Profile Name:PROD

Save As ...View History

Description:Installation Defaults

Authentication Domain:

Help URL:

☒ Compress Responses

☒ Compress Response References

Compress Mime Types:application/x-javascript,text/javascript,text/css,text/html

☒ Compress Query

Save Confirmation Display Time:3,000 Milliseconds

☒ Enable Processing Message

☒ Enable New Window

☒ Enable PPM Agent

PPM Monitor Buffer Size:51,200 KB

☐ Single Thread Netscape

Single Thread Delay:1,000 Milliseconds

Non-standard Base Path:

Reports

☒ Enable Report Repository

Report Repository Path:

Compress Report Output

☒ All Browsers

☐ Exclude NetScape

☐ 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 this button 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 *srv1.oracle.com* and *srv2.oracle.com*, you must specify an authentication domain of *.oracle.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 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 converted to lowercase.

Note. You must enter your authentication domain to enable all functionality associated with PeopleSoft Related Content Framework.

Help URL

Specify the URL that is needed to link to the correct location in your HTML PeopleBooks. When users click the Help link, the appropriate context-sensitive PeopleSoft documentation should appear. To remove the Help link, omit this value, 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/htmldoc/f1search.htm?ContextID=%CONTE
XT_ID%&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	<p>Select to enable compression in the communication between the web server and the browser. Gzip and Compress protocols are supported.</p> <p>This check box is selected by default.</p>
Compress Response References	<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 deselected by default.</p>
Compress MIME Types	<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/javascript,text/javascript,text/css,text/html</i>.</p>
Compress Query	<p>This property applies to browser requests in which the content type section of the URL is <i>/q/</i>, indicating a query. Select to enable compression of query responses to the browser.</p> <p>This check box is selected by default.</p>
Save Confirmation Display Time	<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 <i>3000</i> milliseconds (3 seconds).</p>
Enable Processing Message	<p>Select to enable processing notification while the system processes a request.</p> <p>This check box is selected by default.</p>
Enable New Window	<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>
Enable PPM Agent (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>
PPM Monitor Buffer Size (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 <i>51200</i> KB (50 megabytes).</p>

Single Thread Netscape

Select to indicate that requests from a Netscape browser should be single-threaded to prevent crashes.

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.

This check box is deselected by default. When selected, the Single Thread Delay option appears.

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 9, "Configuring the Portal Environment," Viewing Web Profile History, page 278](#); [Chapter 9, "Configuring the Portal Environment," Understanding the Authentication Domain, page 236](#); [Chapter 9, "Configuring the Portal Environment," Configuring Caching, page 258](#); [Chapter 9, "Configuring the Portal Environment," Configuring Virtual Addressing, page 254](#) and *PeopleTools 8.52: Performance Monitor*, "Administering the Performance Monitor," Working with Performance Monitor Web Profile Properties.

See the PeopleTools 8.52 installation guide for your database platform.

Reports**Enable Report Repository**

Select to enable users to view the report repository. When this check box is deselected, 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 Microsoft Windows only) If the report repository resides on a different machine from 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

PeopleTools 8.52: PeopleCode API Reference, "Internet Script Classes (iScript)," AuthTokenDomain

PeopleTools 8.52: System and Server Administration, "Working with Oracle WebLogic"

PeopleTools 8.52: System and Server Administration, "Working with IBM WebSphere"

Configuring Portal Security

Access the Web Profile Configuration page. (Select PeopleTools, Web Profile, Web Profile Configuration. Select the Security tab.)

Web Profile Configuration - Security page

Days to Autofill User ID

If this field is populated, the system caches the user ID and automatically inserts it on the sign-in page. This is 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 that 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 depends on the size of the file.

If the value is *greater than 0*, then the amount of time that the file remains on the web server depends only on 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 specify 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 9, "Configuring the Portal Environment," Using SSL Accelerators, page 289.](#)

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. Deselecting this option prevents 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 whether the content link is from a registered node by checking whether 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 checks only against the Content URI Text and Portal URI Text if it contains text that is longer than 12 characters.

SSL (Secure Sockets Layer)

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, deselect this check box (the default setting).

Secure Cookie with SSL

Select to prevent the single sign on 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 sign on cookie (PS_TOKEN) to True.

PeopleSoft single sign on 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 uses a proxy server to relay 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 sign on 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, deselect this check box to enable single signon functionality between HTTPS and HTTP servers.

Note. Before you deselect 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).

HTTP Session Inactivity

Specify the number of seconds of inactivity after which the HTTP session times out for authenticated users. When the interval passes with no user activity, the web server discards all session information, including cached page states. The next time the user submits a request, the web server creates a new HTTP session. From the user experience, the user may be put in a search page or home page because the previous session state is lost at this point. This property is unaffected by any permission list setting.

When this property isn't specified, the HTTP session time-out interval is the same value as the authenticated user inactivity logout property that is specified on this page.

This property is similar to the public users HTTP Session Inactivity property that is specified on this page.

Use this property with caution. Don't use this property if you don't understand its usage.

Inactivity Logout

Specify the value in seconds of the inactivity time-out interval that applies to PeopleSoft applications to which a user is signed in. When the interval passes with no user activity, the user's browser displays the page that is specified by the Expire Page - Page field on the Web Profile Configuration - Look and Feel page. The web server applies this value in JavaScript.

Note. The inactivity logout can be overridden by a permission list inactivity time-out setting.

The default value is *1200* seconds.

Note. Depending on the application implementation, authenticated users might also experience an HTTP session inactivity time-out, which by default is the same as the inactivity logout value that you specify here.

However, you can override this behavior by using the HTTP Session Inactivity property on this page. This property independently controls the HTTP session inactivity time-out. The HTTP Session Inactivity property is unaffected by any permission list setting.

Timeout Warning Script

Displays the name of the time-out warning script that is currently in effect. The default value is

WEBLIB_TIMEOUT.PT_TIMEOUTWARNING.FieldFormula.IScript_TIMEOUTWARNING.

Override

Click to change the time-out warning script. The Override Timeout Warning Script page appears, with these fields:

- Record (Table) Name: The default value is *WEBLIB_TIMEOUT*.
- Field Name: The default value is *PT_TIMEOUTWARNING*.
- PeopleCode Event Name: The default value is *FieldFormula*.
- PeopleCode Function Name: The default value is *IScript_TIMEOUTWARNING*.

The values of these fields are concatenated to produce the value in the Timeout Warning Script field.

See *PeopleTools 8.52: Security Administration*, "Setting Up Permission Lists," Setting General Permissions.

Public Users

Public users are PeopleSoft application users who are not required to go through a PeopleSoft sign in page. As a result, public users are neither identified nor authenticated by their own user ID and password. Public users also are not forwarded to the sign in page after any period of inactivity.

Because they access sites without credentials, public users should be limited to informational websites where sensitive data is not accessible.

You should enable public user access under these conditions:

- When you provide users with direct links to nonsecure pages.

Note. If a user clicks a direct link that accesses a page for which the public user ID is not authorized, the system displays the sign-in page prompting the user to supply her PeopleSoft credentials.

- When you use an external authentication method.

Important! If you are using an external authentication method in conjunction with the PeopleCode *SwitchUser* function, then the Inactivity Warning and Inactivity Logout values specified for authenticated users are restored.

See *PeopleTools 8.52: PeopleCode Language Reference*, "PeopleCode Built-in Functions," SwitchUser.

- When you want crawlers to search your PeopleSoft site and need to provide access to the portal.

Allow Public Access

Select so that the system bypasses the user sign-in page when users click a direct link to a page. The system uses the values that you specify in the User ID and Password fields for authentication. This check box is deselected by default.

When you select this check box, the system enables the User ID, Password, and HTTP Session Inactivity fields.

User ID

Enter the account that the system uses to authenticate public users. This field is enabled when you select the Allow Public Access check box.

Note. This is *not* the same as the WebUserId property in the configuration.properties file, which you specify during the PeopleSoft Pure Internet Architecture installation for accessing the web profile.

Password

Enter the password for the account that you entered in the User ID field. This field is enabled when you select the Allow Public Access check box

Note. This is *not* the same as the WebPassword property in the configuration.properties file, which you specified during the PeopleSoft Pure Internet Architecture installation for accessing the web profile.

HTTP Session Inactivity

Specify the value in seconds of the inactivity period that applies to public users.

When the period passes with no user activity, the web server discards all session information, including cached page states. The next time the user submits a request, the web server creates a new HTTP session.

This field is enabled when you select the Allow Public Access check box.

Note. If you are using an external authentication method in conjunction with the SetAuthenticationResult() function in Signon PeopleCode or the SwitchUser() function in Non-Signon PeopleCode, you are logged out if you are inactive.

The default value is *1200* seconds.

See *PeopleTools 8.52: PeopleCode Language Reference*, "PeopleCode Built-in Functions," SwitchUser.

Web Server Jolt Settings**Disconnect Timeout**

Specify the amount of time to wait while disconnecting the Oracle Jolt connection. A value of *0* means no limit.

The default value is *0* seconds.

Send Timeout

Specify the maximum number of seconds that the servlet allows for a request to be sent to the application server. This setting does not indicate a maximum amount of time for the service to finish; it indicates only the maximum amount of time to send the request to the application server.

The default value is *50* seconds.

Receive Timeout

Specify the maximum number of seconds that the servlet should wait for a response from the application server.

Make sure that the value of this field is greater than your application server service time-outs, such as the Service Timeout setting for PSAPPSRV that appears in the PSAPPSRV.CFG configuration file on the application server.

The default value is *600* seconds.

XML Link**User ID**

Enter the user account that the system uses to authenticate users for XML Link technology.

Password

Enter the password for the account name that you entered in the User ID field.

XML Link 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 xmllink servlet for content that is hosted on the same server as the xmllink servlet.

This check box is selected by default.

See Also

Chapter 9, "Configuring the Portal Environment," Improving Same-Server Performance Under SSL, page 288

PeopleTools 8.52: System and Server Administration, "Working with Oracle WebLogic"

Chapter 9, "Configuring the Portal Environment," For IBM WebSphere, page 292

Configuring Virtual Addressing

Access the Web Profile Configuration page. (Select PeopleTools, Web Profile, Web Profile Configuration. Select the Virtual Addressing tab.)

General Security **Virtual Addressing** Cookie Rules Caching Debugging Look and Feel

Profile Name: PROD

☒ Generate Relative URLs ?

Default Addressing

Protocol: ? Name: ? Port: ?

Reverse Proxy Server Customize | Find | View All | First 1 of 1 Last

	*Protocol	*Host	*HTTP	*HTTPS	
1	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="+"/> <input type="button" value="-"/>

[What is Reverse Proxy Server ?](#)

Web Profile Configuration - Virtual Addressing page

Generate Relative URLs

Select to generate relative URLs. This setting is for use with proxy server implementations.

This check box is selected by default.

Default Addressing**Protocol**

Specify the default protocol—*HTTP* or *HTTPS*—to override the protocol that is used by the PeopleSoft Pure Internet Architecture and the portal to construct URL references. When this property is omitted, the protocol of the incoming request to the PeopleSoft Pure Internet Architecture servlet is used.

You must set this property when the browser makes an SSL connection and SSL is terminated before the PeopleSoft Pure Internet Architecture servlet, using a device such as an SSL accelerator or a reverse proxy server, or when you select the PIA use HTTP Same Server property on the Web Profile Configuration - Security page. In this situation, the protocol of the request to the PeopleSoft Pure Internet Architecture servlet is HTTP, but the PeopleSoft Pure Internet Architecture must generate URL references with an HTTPS protocol.

Name

Enter a default server name to override the server name that is used by the PeopleSoft Pure Internet Architecture and the portal to construct URL references. When this property is omitted, the server name of the incoming request to the PeopleSoft Pure Internet Architecture servlet is used to construct URL references.

You must set this property when the server on which the PeopleSoft Pure Internet Architecture servlet resides is different from the server to which the browser is connected. This can occur when a reverse proxy server or load balancer is configured in front of the web server. In these situations, the PeopleSoft Pure Internet Architecture must generate URL references with the server that is used by the browser for its initial connection.

Note. If you specified an authentication domain on the Web Profile Configuration - General page, it must be part of the server name that you specify here. The value that you enter in this field is automatically converted to lowercase.

Port

Enter a default port to override the port that is used by the PeopleSoft Pure Internet Architecture and the portal to construct URL references. When this property is omitted, the port of the incoming request to the PeopleSoft Pure Internet Architecture servlet is used to construct links on a PeopleSoft Pure Internet Architecture or portal page.

You must set this property when the port through which the PeopleSoft Pure Internet Architecture servlet is accessed is different from the port to which the browser is connected. This can occur when an SSL accelerator is used or when a reverse proxy server is configured in front of the web server, and the reverse proxy server or load balancer listens on a different port from the web server, or when you select the PeopleSoft Pure Internet Architecture use HTTP Same Server property on the Web Profile Configuration - Security page. In these situations, the PeopleSoft Pure Internet Architecture must generate URL references with the port that is used by the browser for its initial connection.

Reverse Proxy Server List

Use this grid to specify the reverse proxy servers through which the portal can expect to retrieve content. External content that is retrieved from these sources and contains relative references is rewritten by the portal to contain relative references instead of absolute references to preserve reverse proxy server requirements.

For each reverse proxy server on the list, you specify:

- The protocol to use.
- The server's host or machine name.
- The server's HTTP port number.
- The server's HTTPS port number.

Configuring Cookie Rules

Access the Web Profile Configuration page. (Select PeopleTools, Web Profile, Web Profile Configuration. Select the Cookie Rules tab.)

GeneralSecurityVirtual AddressingCookie RulesCachingDebuggingLook and Feel

Profile Name: PROD

Cookie Rules

Find | View All | First | 1-4 of 4 | Last

*Cookie Pattern ?	Cookies Passed to Server ?	Cookies Not Passed to Server ?	Delete Cookie on Logout ?	Proxied ?	Secure ?	
*	*%AuthTokenDomain		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	+ -
PSJSESSIONID			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	+ -
WebLogicSession			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	+ -
JServSessionId*			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	+ -

Web Profile Configuration - Cookie Rules page

This page defines rules that determine how the portal passes cookies to servers in the same domain.

Oracle WebLogic and IBM WebSphere web servers use browser cookies (containing the server path and domain) to establish session identity. These cookies have a default name that's used to retrieve the cookie on each request to the web server. In an environment in which multiple web servers are in use with the portal, you need to define unique session cookie names between web servers to prevent one cookie from overwriting another cookie of the same name set by a different web server. The path or domain of the cookie must be different to prevent overwriting.

Note. Use the first three fields to specify patterns to match, where the asterisk (*) is treated as a wildcard, matching zero or more characters.

The portal knows its own domain (from the authentication domain), and the first default rule instructs it to pass portal servlet cookies to any server in the same domain, over both secure and insecure connections. All other cookies can continue to be passed to any server on any domain.

Cookie Pattern	Specify the name pattern for cookies to which this rule applies. For example, to specify all cookies whose names start with <i>HR_</i> , enter the value <i>HR_*</i> .
Cookies Passed to Server	Specify the name pattern for the servers to which the specified cookies should be passed. For example, to specify all servers in the domain <i>.hr.peoplesoft.com</i> , enter the value <i>*.hr.peoplesoft.com</i> . You can use a substitution parameter <i>%AuthTokenDomain</i> to represent the authentication domain currently in effect.

Cookies Not Passed to Server	Specify the name pattern for the servers to which the specified cookies should not be passed. You can use a substitution parameter %AuthTokenDomain to represent the authentication domain currently in effect.
Note. This field takes precedence over the Cookies Passed to Server field.	
Delete Cookie on Logout	Select to indicate that the cookie should be deleted when the user signs out of the portal.
Note. This check box is deselected for the first default rule (matching all cookies), but any cookies that are generated by PeopleSoft with, for example, the PSJSESSIONID prefix, are deleted when the user signs out. If you're running multiple PeopleSoft portals and want cookies to be maintained after the user signs out, update this page to specify which cookies to preserve or delete.	
Warning! Oracle recommends that you use extreme caution when setting all cookies to be deleted on sign-out. Cookie pattern "*" means all cookies. Some cookies must be kept. For example, the Arrow Point cookie (ARPT) is used by some load balancers to ensure sticky http(s) sessions.	
Proxied	Select to indicate that the cookie should be proxied in which the cookie is retrieved through an HTML template. The cookie isn't set directly into the browser without proxying. This setting is useful for applications where a particular cookie name is not unique across the authentication domain.
Secure	Select to indicate that the cookie should be passed only over a secure connection.

This table shows how cookie names are set in WebSphere for the Portal and pspc (PeopleSoft Portlet Container) web modules:

<i>WAR Name</i>	<i>WebSphere Session Cookie Name</i>
Portal	%MACHINE_NAME%-%WEB_SERVER_HTTP_PORT%-PORTAL-WASPSJSESSIONID
pspc	%MACHINE_NAME%-%WEB_SERVER_HTTP_PORT%-PORTLET-WASPSJSESSIONID

Configuring Caching

Access the Web Profile Configuration page. (Select PeopleTools, Web Profile, Web Profile Configuration. Select the Caching tab.)

On the Browser

☒ Cache Generated HTML ?

Number of States Supported: ?

State Discard Interval: Seconds ?

☒ Cache Homepage ?

Homepage State Interval: Seconds ?

Browsers

Find | View All | First | 1-6 of 6 | Last

*User Agent ID ?

Cache Home Page ?

Mozilla/4.0 (compatible; MSIE 5.01; Windows NT; DigExt)	<input checked="" type="checkbox"/>	+ -
Mozilla/4.0 (compatible; MSIE 5.0; Mac_PowerPC)	<input checked="" type="checkbox"/>	+ -
Mozilla/4.0 (compatible; MSIE 4.5; Mac_PowerPC)	<input checked="" type="checkbox"/>	+ -
Mozilla/4.72 [en] (WinNT; I)	<input checked="" type="checkbox"/>	+ -
Mozilla/5.0 (Windows; U; WinNT4.0; en-US; m18) Gecko/20001108 Netscape6/6.0	<input checked="" type="checkbox"/>	+ -
Mozilla/5.0 (Macintosh; U; PPC Mac OS X; en-us) AppleWebKit/85 (KHTML, like Gecko) Safari/85	<input checked="" type="checkbox"/>	+ -

On the Web Server

☒ Cache Proxied JavaScripts ?

☒ Cache Portal Objects ?

Cache Stale Interval: Seconds ?

☒ Cache Target Content ?

☐ Cache Menu ?

Cache Purge All Hit Count: ?

Directories

Image Directory: <input type="text" value="/cache"/> ?	Chart Directory: <input type="text" value="/cache/chart"/> ?
Image Web Directory: <input type="text" value="/cache"/> ?	Chart Web Directory: <input type="text" value="/cache/chart"/> ?
CSS Directory: <input type="text" value="/cache"/> ?	JavaScript Directory: <input type="text" value="/cache"/> ?
CSS Web Directory: <input type="text" value="/cache"/> ?	JavaScript Web Directory: <input type="text" value="/cache"/> ?

☐ Copy Image/CSS (no Versioning) ?

Recent Search Results

☐ Enable Caching ?

☐ Enable Caching on Web Server ?

Web Profile Configuration - Caching page

On the Browser

- Cache Generated HTML

Select to indicate that PeopleSoft application pages should be cached by the user's browser. For security reasons, deselect this check box in kiosk environments.

This check box is selected by default.

Note. When you use menu navigation, the back button works even if this option is deselected because menu navigation does not generate the No-Cache directive.

Number of States Supported Specify how many browser states the portal should support when the user accesses PeopleSoft application pages.

Note. If you have applications that make numerous server requests, you may want to increase this value. This increases the virtual machine's memory requirements, so be prepared to allocate more memory accordingly.

The default value is 5 states.

State Discard Interval

When a user signs out (either explicitly or because of a session timeout), the system normally clears all application states for the user's session. This property enables clearing of application states for individual windows instead. If your users have a long session time-out and many browser windows open, use this property to force the portal web server to release cached states for windows that a user has closed.

Every time a user clicks the New Window link on a portal page, or when transfer PeopleCode opens a new window, the portal web server creates and maintains a state cache for that new browser window. Even if the user closes the window, the web server continues to maintain the cache during the interval that is specified by the Inactivity Logout field on the Web Profile Configuration - Security page. If users open many application browser windows simultaneously and then close most of them while still continuing to work actively in one or two remaining windows, you may be using more web server memory than is required and unnecessarily degrading its performance.

You can improve performance by using this property to specify an independent time-out interval that applies to individual browser windows, which forces the portal web server to release cached states for windows that users have closed.

When a user clicks the New Window link, and each time user activity in a window produces a request for data from the web server, the portal applies a current timestamp to its corresponding state cache. The next time the user clicks the New Window link, the portal examines the timestamp for each state cache. For any timestamp that's older than the status block time-out, the portal web server discards the associated window state cache on the assumption that the window must have been closed.

Note. The portal performs this comparison only when the user clicks the New Window link.

Enter a value that represents the number of seconds for the inactivity time-out per window. This time-out should be as short as possible, but still long enough to avoid discarding the state cache of any window that's still in use. A good starting value is the same value as the inactivity logout. If you don't specify this property, or if you enter 0, status blocks are tracked per session—all status blocks are deleted when the user signs out or when the inactivity logout interval expires.

Note. A potential inconvenience with this property is that users might be inactive in a window without closing it, then return to it after the interval that you specify here has expired. In that case, they're presented with the application's search page.

Cache Homepage

Select to apply homepage caching on the browser. When this check box is selected, the Homepage Stale Interval field and the Browsers grid become available.

This check box is selected by default.

Important! Do not enable homepage caching if any pagelets that appear on a homepage or dashboard use interactive processing mode.

Homepage Stale Interval

Enter the number of seconds that the browser should wait before requesting an updated homepage from the portal server.

The default value is *1200* seconds (20 minutes).

See [Chapter 8, "Using Portal Caching Features," Implementing Homepage Caching, page 230](#) and [Chapter 8, "Using Portal Caching Features," Implementing PeopleSoft Page Caching, page 232](#).

Browsers

Use this grid to identify the browsers that you don't want to cache the homepage. For each browser make, model, and version, supply the identifying user agent ID and deselect the Cache Home Page check box. To re-enable caching for a listed browser, select the browser's Cache Home Page check box.

Any browser that is not listed caches the homepage if the global Cache Homepage check box is selected.

Note. If the global Cache Homepage check box is deselected, you can't enable homepage caching for individual browsers.

See [Chapter 8, "Using Portal Caching Features," Implementing Homepage Caching, page 230](#).

On the Web Server**Cache Proxied JavaScripts**

Select to enable caching of proxied JavaScripts on the portal server. This check box is selected by default.

Cache Portal Objects

Select to enable metadata caching. This check box is selected by default.

Cache Stale Interval

Enter the number of seconds that should pass before the portal refreshes the metadata cache. The default value of this property is *86400* seconds (24 hours).

Note. This property competes for effect with Cache Purge All Hit Count.

Cache Target Content

Select to cache all target content HTML that has an appropriately defined Cache element. This check box is selected by default.

Cache Menu

Select to enable web server-based navigation caching.

For the DEV and TEST web profiles, the default value is *True*. For all other profiles, including those that are newly created, the default value is *False*.

Cache Purge All Hit Count Specify the maximum total number of HTTP requests that the web server should receive for metadata objects before it purges the metadata cache, forcing the portal to refresh the cache.

The default value is *1000* requests. Set this property to *0* to disable the feature.

Note. This property competes for effect with the cache stale interval.

See [Chapter 8, "Using Portal Caching Features," Implementing Target Content Caching, page 223](#); [Chapter 8, "Using Portal Caching Features," Implementing Metadata Caching, page 227](#); [Chapter 8, "Using Portal Caching Features," Implementing Proxied JavaScript Caching, page 229](#) and [Chapter 8, "Using Portal Caching Features," Administering Web Server-Based Navigation Caching, page 233](#).

Directories

Image Directory Enter the image file cache directory.

The default value is */cache*.

Image Web Directory Enter the image file web cache directory.

The default value is */cache*.

CSS Directory (cascading style sheet directory) Enter the cascading style sheet (CSS) cache directory.

The default value is */cache*.

CSS Web Directory Enter the CSS web cache directory.

The default value of this field is */cache*.

Copy Image/CSS (No Versioning) Select to have the system write a copy of the image and CSS cache with no version number. This check box is provided in case an external reference to the PeopleSoft stylesheet is needed.

Warning! Enable this setting with care. In a production environment, it can double the number of files that are stored in the site's cache directory. Unless a clear need exists to use a custom solution to access style sheets and the like, this check box should be disabled.

This check box is deselected by default.

Chart Directory Enter the directory of the cache for the chart image file.

The default value is */cache/chart*.

Chart Web Directory Enter the directory of the web cache for the chart image file.

The default value is */cache/chart*.

JavaScript Directory Enter the Java Script cache directory.

The default value is */cache*.

JavaScript Web Directory

Enter the JavaScript web cache directory.
The default value is */cache*.

Recent Search Results**Enable Caching**

Select to temporarily store recent (persistent) search results on the browser and web server. Deselecting this setting suppresses all search persistence property settings that you configure in PeopleSoft Application Designer definitions. The default value is *unchecked*.

Enable Caching on Web Browser

Select to temporarily store recent (persistent) search results on the web server. deselecting this setting suppresses all search persistence property settings that you configure in PeopleSoft Application Designer definitions. This option applies only when you use either Internet Explorer 7 or Safari 3 as your browser. This option does not affect Internet Explorer 8, Firefox 3.5, or Safari 4. The default value is *unchecked*.

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Creating Record Definitions," Setting Record Field Use Properties.

Configuring Trace and Debug Options

Access the Web Profile Configuration page. (Select PeopleTools, Web Profile, Web Profile Configuration. Select the Debugging tab.)

GeneralSecurityVirtual AddressingCookie RulesCachingDebuggingLook and Feel

Profile Name: PROD

☐ Trace Monitoring Server ?

☐ Trace PPM Agent ?

☐ Show Connection & Sys Info ?

☐ Show Trace Link at Signon ?

☐ Show Layout ?

☐ Show Overlapping Fields ?

☐ Show StyleSheet Inline HTML ?

☐ Show JavaScript Inline HTML ?

☐ Generate HTML for Testing ?

☒ Write Dump File ?

☐ Create File from PIA HTML Page ?

Web Profile Configuration - Debugging page

Trace Monitoring Server	<p>Select to have the PeopleSoft Performance Monitor and PPMI servlets write debug information to the web server log. This check box is deselected by default.</p> <hr/> <p>Note. This feature is not an agent trace. For an agent trace, use Trace PPM Agent.</p> <hr/>
Trace PPM Agent	<p>Select to enable PeopleSoft Performance Monitor tracing on performance agents.</p> <p>This check box is deselected by default.</p>
Show Connection Information	<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. The default is deselected.</p> <hr/> <p>Note. This information is useful for orientation and troubleshooting purposes, but might not be suitable for end users.</p> <hr/>
Show Trace Link at Signon	<p>Select to display a URL link at sign-in. The link opens a page for setting trace parameters.</p>

Show Layout	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.
Show Overlapping Fields	Select to include comments in generated HTML pages that may help in diagnosing page layout problems, such as fields overlapping other fields.
Show StyleSheet Inline HTML	Select to insert the page's stylesheet into its generated HTML.
Show JavaScript Inline HTML	Select to display all the JavaScript functions that are used for processing in the generated HTML page.
Generate HTML for Testing	<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>Note. Selecting this option may cause some pages or pagelets not to appear correctly. You can also set this in the application server configuration file with the TracePIA option. To ensure that all pages appear correctly, verify that TracePIA isn't enabled in the configuration file.</p> <hr/>
Write Dump File	Select to have the system write a log file to the web server if an Oracle Jolt exception error occurs.
Create File from PIA HTML Page (create file from PeopleSoft Pure Internet Architecture HTML page)	<p>Select to view and debug the source HTML that the application server generates. The system saves each generated page as <i>PS_CFG_HOME</i> \appserv\domain\LOGS\client\element\N.html.</p> <p>The variables in the name are:</p> <ul style="list-style-type: none"> • <i>Domain</i>: The name of the application server domain. • <i>Client</i>: The name of the machine or Internet Protocol (IP) address where the browser is running. • <i>Element</i>: The name of the process or object that generates the HTML (query name for query, program name for iScripts, and so on). • <i>N</i>: The state number for the generated page. <hr/> <p>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.</p> <hr/>

See [Chapter 9, "Configuring the Portal Environment," Configuring Custom Properties, page 274](#) and *PeopleTools 8.52: Performance Monitor*, "Administering the Performance Monitor," Working with Performance Monitor Web Profile Properties.

Configuring Appearance and Character

Access the Web Profile Configuration page. (Select PeopleTools, Web Profile, Web Profile Configuration. Select the Look and Feel tab.)

General	Security	Virtual Addressing	Cookie Rules	Caching	Debugging	Look and Feel	Custom Properties
Profile Name: PROD							
Start Page							
Page:		start.html,start.wml					?
Script:		WEBLIB_MENU.ISCRIPT3.FieldFormula.IScript_StartHtml					<input type="button" value="Override"/> ?
Expire Page							
Page:		expire.html,expire.wml					?
Content Name:		PT_EXPIRE,PT_EXPIRE_WML					?
Error Pages							
Exception Page:		exception.html,exception.wml					?
Pagelet Error Page:		portalerrorpagelet.html					?
Portal Detail Error Page:		portalerrordetail.html					?
Portal Target Error Page:		portalerrortarget.html					?
MCF Auth Failure Page:		mcferror.html					?
Other Pages							
Auth Token Enable Page:		authtokenenabled.html					?
Enable Trace Page:		signintrace.html					?
Cookies Required Page:		cookiesrequired.html					?
SSL Required Page:		sslrequired.html					?
User Profile Page:		userprofile.html					?
Signon/Logout Pages							
Signon Page:		signon.html,signon.wml					?
Signon Result Doc Page:		signonresultdoctext.html					?
Signon Error Page:		signin.html,signin.wml					?
Logout Page:		signin.html,signin.wml					?
Password							
Password Expired Page:		passwordexpired.html					?
Password Warning Page:		passwordwarning.html					?
Change Password On Expire:		MAINTAIN_SECURITY.EXPIRE_CHANGE_PSWD.GBL					<input type="button" value="Override"/> ?
Change Password On Warning:		MAINTAIN_SECURITY.CHANGE_PASSWORD.GBL					<input type="button" value="Override"/> ?
Language Support:		zh_HK=ZHT <div>?</div>					

Web Profile Configuration - Look and Feel page

Start Page

Page

Displays the page to which the system redirects users after a successful sign-in. This value references the iScripts that build the PeopleSoft navigation.

Do not change this setting.

The default value is *start.html,start.wml*.

Script

Displays the name of the start page iScript that is currently used for opening a new window.

The default value is

WEBLIB_MENU.ISCRIPT3.FieldFormula.IScript_StartHtml.

Override

Click to change the start page script. The Override Start Page Script page appears with these fields:

- Record (Table) Name: The default value is *WEBLIB_MENU*.
- Field Name: The default value is *ISCRIPT3*.
- PeopleCode Event Name: The default value is *FieldFormula*.
- PeopleCode Function Name: The default value is *IScript_StartHtml*.

The values of these fields are concatenated to produce the value in the Script field.

Expire Page

Page

Displays the HTML page containing text variables that are defined in text.properties. This page appears when user inactivity exceeds the limit that is specified by the Inactivity Logout field in the Authenticated Users region of the Web Profile Configuration - Security page.

Do not change this setting.

The default value is *expire.html,expire.wml*.

Content Name

Enter the content name that is stored in the HTML catalog. It appears when a page has expired due to reaching the limit that is specified by the Number of States Supported field in the On the Browser region of the Web Profile Configuration - Caching page.

Change this value with caution.

The default value is *PT_EXPIRE,PT_EXPIRE_WML*.

See [Chapter 9, "Configuring the Portal Environment," Configuring Portal Security, page 247](#) and [Chapter 9, "Configuring the Portal Environment," Configuring Caching, page 258](#).

Error Pages

Exception Page	Enter the name of the page that is used in Java to handle exceptions. Change this value with caution. The default value is <i>exception.html,exception.wml</i> .
MCF Auth Failure Page (multichannel framework authorization failure page)	Enter the name of the multichannel framework authorization failure page. The default value is <i>mcfererror.html</i> .
Pagelet Error Page	Enter the name of an initial error message page to be displayed in the user's browser when a portal pagelet is unavailable. The default value is <i>portalerrorpagelet.html</i> .
Portal Target Error Page	Enter the name of an initial error message page to be displayed in the user's browser when target content is unavailable. The default value is <i>portalerrortarget.html</i> .
Portal Detail Error Page	Enter the name of an error message page that can be displayed in the user's browser to provide more detail about an error when retrieving portal content. The default value is <i>portalerrordetail.html</i> .

Because your PeopleSoft portal can aggregate its content from many different sources, errors can occur for a variety of reasons. An error may occur if:

- A content server is down.
- An invalid URL is specified.
- A portal node from which content is being requested is inactive.
- A portal node fails to deliver requested content within the time-out interval that you specify by using the PSTIMEOUT content reference attribute.

See [Chapter 12, "Developing and Registering Pagelets," Configuring Pagelet Time-out Settings, page 510](#).

For practical purposes, all of these reasons result in the requested content being unavailable.

The values that you specify for the pagelet error page, portal target error page, and portal detail error page are the names of HTML pages that present customizable, translatable error messages. The default pages are delivered with the PeopleSoft system and provide useful error messages without any modification.

You can use the delivered error pages, modify them, or create your own. You can choose additional error message information to be displayed by using bind-type variables of two types in your error pages:

- Numeric message variables

These map to strings that are predefined in the following file:

PIA_HOME\webserv\peoplesoft\applications\peoplesoft\PORTAL\WEB-INF\psftdocs\sitename\text.properties

For example, the numeric variable 5006 maps to the phrase "The portal was unable to retrieve the page you are looking for."

- Session variables

These map to system information that depends on the current page and activity.

Session variables have values that are dynamically resolved by the portal engine at runtime. Available session variables include:

- ErrorTitle
- ErrorDescription
- ErrorURL
- AccessedURL
- OriginalURL
- TargetPage
- StackTrace
- DetailError

DetailError is the name of the page that is specified by the Portal Detail Error Page field. This can be used to specify a link target.

To invoke a numeric or session variable, you insert it between the strings `<%=` and `%>` in your HTML. For example: `<%=5009%>` or `<%=ErrorDescription%>`. Examine the delivered error pages for examples of how to use the numeric and session variables within your HTML.

Other Pages

Auth Token Enable Page (authorization token enable page)

Enter the name of the page to be displayed when the site being accessed is configured with an authentication domain (or web server session cookie domain), and the URL from the browser doesn't include the domain. This page contains a link to a sign-in page that produces the correct URL for the site. You can modify the content interface and design, but do not change the internal metatags that generate the correct URL.

The default value is *authtokenenabled.html*.

Enable Trace Page

Enter the name of the page to be used to set trace parameters. If you enable tracing, this page appears before the sign-in page so that you can set the trace parameters and then sign in to the system.

The default value is *signintrace.html*.

Cookies Required Page	<p>Enter the name of a page to be displayed when the browser does not accept cookies. You should configure browsers to accept cookies.</p> <p>The default value is <i>cookiesrequired.html</i>.</p>
SSL Required Page	<p>Displays the name of the page that should appear if the Secured Access Only check box is selected in the SSL region of the Web Profile Configuration - Security page, and the user is unable to proceed without SSL.</p> <p>Do not change this setting.</p> <p>The default value is <i>sslrequired.html</i>.</p>
User Profile Page	<p>Displays the name of the page that appears when the user clicks the link from a password-expired page.</p> <p>Do not change this setting.</p> <p>The default value is <i>userprofile.html</i>.</p>

See [Chapter 9, "Configuring the Portal Environment," Configuring General Portal Properties, page 242.](#)

Signon/Logout Pages

Signon Page	<p>Displays the name of the page that redirects to the servlet for the sign-in process.</p> <p>Do not change this setting.</p> <p>The default value is <i>signon.html,signon.wml</i>.</p>
Signon Result Doc Page (signon result document page)	<p>If your portal site is configured with sign-in PeopleCode to additionally validate the user, that code can call a result document if the user authentication failed. You can present the result document to the user in several ways by specifying one of these HTML documents in this field:</p> <p><i>signonresultdoctext.html</i>: The result text is wrapped in a standard portal error page. This is the default value.</p> <p><i>signonresultdocpage.html</i>: The result text itself is formatted as the error page.</p> <p><i>signonresultdocredirect.html</i>: This page redirects the user to your own result document, using the <code><%=resultDoc%></code> session variable element.</p>
Signon Error Page	<p>Enter the name of the page that should appear when the user makes an error signing in. To customize your sign-in page, clone <i>signin.html</i> as a starting point.</p> <p>Change this value with caution.</p> <p>The default value is <i>signin.html,signin.wml</i>.</p>

Logout Page

Enter the name of the page that should appear when the user signs out. You may define a custom logout page.

Change this value with caution.

The default value is *signin.html,signin.wml*.

Password**Password Expired Page**

Displays the name of the page that should appear when the user password is expired.

Do not change this setting.

The default value is *passwordexpired.html*.

Password Warning Page

Displays the name of the page that should appear when the user's password is about to expire in the number of days specified in PeopleSoft security.

Do not change this setting.

The default value is *passwordwarning.html*.

Change Password on Expire

Displays the Change Password page content ID. The system uses the value in *passwordexpired.html* to take the user to the Change Password page when a password is expired.

The default value is

MAINTAIN_SECURITY.EXPIRE_CHANGE_PSWD.GBL.

Override

Click to override the expired password change page. The Override Change Password on Expire Page page appears.

- Menu Name: The field is blank by default.
- Component: The field is blank by default.
- Market: The default value is *Global*.

Change Password On Warning

Displays the content ID for the Change Password page. The system uses the value that you enter in *passwordwarning.html* to take the user to the Change Password page when a password warning is required.

The default value is *MAINTAIN_SECURITY.CHANGE_PASSWORD.GBL*.

Override

Click to override the Password Change Warning page. The Override Change Password on Warning Page page appears.

- Menu Name: The field is blank by default.
- Component: The field is blank by default.
- Market: The default value is *Global*.

(Optional) Language Support

Specify additional mappings from International Organization for Standardization (ISO) locale codes to PeopleSoft language codes, using one line per entry.

The PeopleSoft system uses proprietary codes to represent user languages, and the PeopleSoft Pure Internet Architecture sign-in page appears in the PeopleSoft language corresponding to the user's browser language setting (an ISO locale code). This field enables you to provide mappings from additional ISO locale codes to PeopleSoft language codes to support a wider range of browser-specific language settings.

For example, the entry *zh_HK=ZHT* maps the ISO locale code *zh_HK* (Hong Kong Chinese) to the PeopleSoft language code *ZHT* (traditional Chinese) so that traditional Chinese is the default sign-in language for browsers with the preferred language set to Hong Kong Chinese.

Your PeopleSoft application is delivered with default mappings, shown on the Manage Installed Languages page. The default mappings don't require entries in this field.

See Also

PeopleTools 8.52: Global Technology, "Adding New Languages"

Configuring Custom Properties

Access the Web Profile Configuration page. (Select PeopleTools, Web Profile, Web Profile Configuration. Select the Custom Properties tab.)

Profile Name: PROD

*Property Name	Validation Type	Property Value	
PPMConsole	Boolean	true	+ -
checkForDuplicateCookies	Boolean	false	+ -
noDefaultSignonForWorkflow	Boolean	false	+ -

[What is Custom Properties ?](#)

Web Profile Configuration - Custom Properties page

This page enables the use of web profile properties that have been added since the current release of PeopleTools was shipped, or that are needed only for backward compatibility. For each property, you enter the property name, select its validation type, and provide the appropriate property value. The validation type ensures that the property value that you enter is the correct format. The currently available custom properties are:

AJAXConsole

This property, although not displayed, enables you to launch an interface for viewing diagnostic information related to AJAX parameters that are required by PeopleSoft support personnel, who will provide you with the correct values when necessary.

auditPWD

This property enables certain debug and control settings that are required by PeopleSoft support personnel, who will provide you with the correct values when necessary. Never set this property in a production environment unless directed to do so.

checkForDuplicateCookies

Duplicate cookies occur when all of the following are true:

- You have two PeopleSoft applications installed for which you have not implemented single signon functionality.
- The authentication domain that you specify for one application's web server is a subset of the authentication domain that you specify for the other, such as *.mycompany.com* and *.user.mycompany.com*.
- A user attempts to sign in to both applications simultaneously from the same browser.

When these conditions are met, the browser presents multiple cookies to each application, which produces unpredictable browser behavior, such as displaying the sign-in page or a page expiration message.

For this property, specify a validation type of *Boolean* and enter one of these property values:

True: The portal checks for sets of duplicate cookies. When a duplicate is found, the user is taken back to the sign-in page with this message: "Your browser sent multiple sets of cookies." You can change the text of this message by editing number 107 in the *errors.properties* file of the portal site.

False: The portal doesn't check for sets of duplicate cookies. This is the default setting, which applies when this property isn't specified.

To avoid this issue altogether with applications that don't use single signon functionality, make sure that you specify authentication domains that aren't subsets of each other, such as *.user.mycompany.com* and *.corp.mycompany.com*.

EnableAutocomplete

This property, although not displayed, enables AJAX features associated with autocomplete. For this property, specify a validation type of *Boolean* and enter *False* as the property value to disable autocomplete at the system level.

Note. If this property is not set to false, you can control autocomplete through personalizations.

mutualAuth

This property, although not displayed, controls whether your environment is set up for mutual authentication. The default value is *True*.

Note. If you are using Lightweight Directory Access Protocol (LDAP) authentication, you must use the default setting.

If you are not using LDAP authentication, a Java exception error caused by a failure to retrieve a certificate property may appear in your webserver log. If this occurs, you must change this value to *False* to disable the authentication from trying to retrieve the certificate property.

PPMConsole

This property, although not displayed, enables you to launch an interface for viewing diagnostic information related to agents and PPMI clients. The default value is *False*.

noDefaultSignonForWorkflow

Use the Web Profile Custom Property, `noDefaultSignonForWorkflow`, to control whether a user sees the default signon page after clicking a workflow link. When `noDefaultSignonForWorkflow` is set to true, the system disables Public Access for workflow notification URLs. By default, `noDefaultSignonForWorkflow` is set to *False*. See the following table for information related to the behavior of this setting.

Note. This property applies to worklist URLs, not all workflow URLs. Only the URLs that contain a /w/ use this property.

unauthSessionInterval

Use the Web Profile Custom Property to set session time-out (in seconds) during the authentication process. The default value is *120* seconds.

The purpose is to limit the effectiveness of DOS attacks on failed authentications. To determine the proper time for this property, check the time in seconds that it takes to send an HTTP(S) request from the browser to the web server and multiply the result by 2.

This table describes the behavior of the system based on the `noDefaultSignonForWorkflow` setting.

<i>Configuration</i>	<i>User Session</i>	<i>User Action</i>	<i>System Behavior</i>
Public user access (default signon) is enabled and the custom web property "noDefaultSignonForWorkflow" is set to true.	The user has no session with the system.	The user clicks a workflow notification URL in the email.	The system displays the sign-in page to the user. Once the user signs in to the system with a valid user ID and password, the system takes the user to the workflow notification based on the user profile privileges.
Public user access (default signon) is enabled and the custom web property "noDefaultSignonForWorkflow" is set to true.	The user already holds a valid session with the system through the public access.	The user clicks a workflow notification URL in the email.	Because the user already has a valid session, the system takes the user to the workflow notification. If the public access user profile does not have workflow privileges, the system displays an authorization failure page to the user.
Public user access (default signon) is enabled and the custom web property "noDefaultSignonForWorkflow" is set to true.	The user is already signed in to the system with a valid user ID and password (other than default sign-in)	The user clicks a workflow notification URL in the email.	The system takes the user to the workflow notification based on the user profile privileges.
When the public user access (default signon) is enabled and the custom web property "noDefaultSignonForWorkflow" is set to false.	Any	Any	The system behaves as configured.

Configuration	User Session	User Action	System Behavior
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 *PeopleTools 8.52: Security Administration*, "Implementing Single Signon" and *PeopleTools 8.52: Performance Monitor*, "Administering the Performance Monitor," Viewing Monitor Servlet Diagnostics.

Viewing Web Profile History

Access the Web Profile History page. (Select PeopleTools, Web Profile, Web Profile History.)

Web Profile History

Web Server Name: BUFFY
HTTP Listen Port: 8920
Web Site Name: QEDMO
Profile Name: DEV
Profile was Loaded: 07/22/09 12:18AM
HTTPS Listen Port: 8930
Cookie Name: BUFFY-8920-PORTAL-PSJSESSIONID
Cookie Domain:
Virtual Addressing URL: http://BUFFY/
Listen Address: 10.138.228.133

Properties:

```

appServer=BUFFY:9211
AuthTokenDomain=
configPath=C:\ptdoc\PT850\websrv\peoplesoft\applications\peoplesoft\PORTAL.war\WEB-INF\psftdocs\QEDMO
java.class.path=java.class.path
java.home=C:\bea\JROCKI~1\jre
java.vendor=BEA Systems, Inc.
java.version=1.6.0_05
java.vm.info=compiled mode
java.vm.name=BEA JRockit(R)
joltPooling=Y
os.arch=x86
os.name=Windows XP
os.version=5.1
PortalHTTPPort=8920
PortalHTTPSPort=8930

```

Web Profile History page

Use this page to review the current portal attributes of a web server, website, and web profile, including the web server's HTTP and HTTPS listen ports, the web profile that was last loaded, and the current HTTP session's cookie name and authentication domain.

In addition, the Properties field displays the relevant settings of the web server instance at the time it was loaded by the portal.

Defining Portal Nodes

This section provides an overview of portal nodes and discusses how to:

- Set portal nodes.
- Specify the node type.

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 only 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	Specify the node type to designate the URL format for the selected node definition.

Setting Portal Nodes

Access the Portal page. (Select PeopleTools, Portal, Node Definitions..)

Node DefinitionsConnectorsPortalWS Security

Node NamePT_LOCAL

Details

DescriptionPT_LOCAL☐ Local Node

Tools Release8.50.00.03

Application ReleaseFinancials/SCM 9.10

Example: http://someserver/psc/pshome/

Content URI Text

Example: http://someserver/psp/pshome/

Portal URI Text

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 want; however, Oracle recommends that you use the same PeopleTools 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 want; however, Oracle recommends that you use the same application name release that is 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>Important! 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>Note. The value that you enter in these fields is automatically forced to lowercase.</div>

280

Copyright © 1988, 2011, Oracle and/or its affiliates. All Rights Reserved.

See Also

PeopleTools 8.52: PeopleSoft Integration Broker Administration, "Adding and Configuring Nodes"

Specifying the Node Type

Most of the options on the Node Definitions page apply only to Integration Broker nodes, except for the Node Type. A node type designates the URL format for the node.

Node Type

External: Select if this node represents a non-PeopleSoft system, such as an external website, like www.yahoo.com.

ICType: Select if this node represents a PeopleTools database prior to PeopleTools 8.4. This allows compatibility between pre-PeopleTools 8.4 databases and PeopleTools 8.4 portal databases.

PIA (PeopleSoft Pure Internet Architecture): Select if this node represents a PeopleTools database that is using release 8.4. This is the default setting.

See Also

PeopleTools 8.52: PeopleSoft Integration Broker

PeopleTools 8.52: PeopleSoft Integration Broker Administration, "Adding and Configuring Nodes"

Implementing Single Signon Functionality

This section provides an overview of single signon functionality and discusses how to sign in to a non-default portal.

Understanding Single Signon Functionality

PeopleSoft PeopleTools supports single signon functionality for use with the PeopleSoft Pure Internet Architecture. Within the context of your PeopleSoft system, *single signon* means that after a user has been authenticated by one PeopleSoft application server, that user can access additional PeopleSoft application servers without entering a user ID or password. Although the user is actually accessing a different application server and database, the user navigates seamlessly through the system.

The single signon authentication service is shared by all PeopleTools web services and involves these steps:

1. The first application server (node) checks the HTTP request for a PeopleSoft authentication cookie.
2. If an authentication cookie is not found, then the server returns a language-specific sign-in page.
3. The user completes the sign-in page, entering a user ID and password.
4. The server authenticates the user based on the user ID and password.

5. A web browser cookie stores a unique access token for the user after the initial authentication.
6. An `httpSession` object is created for the newly authenticated session.
7. When the user connects to another PeopleSoft application server (node), the second application server uses the token in the browser cookie to reauthenticate the user without an additional sign-in process.

Because the portal integrates content from various data sources and application servers into a unified interface, you need to configure single signon functionality before deploying the portal. Users sign in once and then can navigate freely without encountering numerous sign-in pages.

However, when frame-based content is accessed across different machines, single signon functionality does not work when you access content on another web server. If the web servers are on the same internet domain, such as *.corp.peoplesoft.com*, single signon functionality remains effective and a sign-in page does not appear.

See Also

PeopleTools 8.52: Security Administration, "Understanding PeopleSoft Security," PeopleSoft Sign In

PeopleTools 8.52: Security Administration, "Employing Signon PeopleCode and User Exits"

PeopleTools 8.52: Security Administration, "Implementing Single Signon"

PeopleTools 8.52: PeopleCode API Reference, "Internet Script Classes (iScript)," `AuthTokenDomain`

Signing In to a Non-Default Portal

To sign in to a portal other than the default, use a URL such as:
`http://portalserver/ps/ps/MY_PORTAL?cmd=login.`

To create a sign-in page that signs in to a non-default portal, change `signon.html` like this:

```
<meta HTTP-EQUIV='Refresh' CONTENT='1; URL=../ps/ps/MY_PORTAL/?cmd=login'>
```

If you are a portal administrator and want to sign in to a database to work with PeopleTools-specific menus, replace the existing portal name with the site template name in the URL to make sure that you are administering your local PeopleTools menu and not inadvertently altering the database. Replace this:

`http://../ps/ps/EMPLOYEE/`

Instead, use this:

`http://../ps/ps/PS_SITETEMPLATE`

See Also

PeopleTools 8.52: PeopleCode API Reference, "Internet Script Classes (iScript)," `AuthTokenDomain`

PeopleTools 8.52: Security Administration

Redirecting Pages with Sensitive Data

If you have a page that contains sensitive data, and you want to redirect the user to it, you should use the provided template `HOMEPAGE_DESIGNER_TEMPLATE`. For the content reference for the component, specify this as the template name. You should also use `PeopleCode` to redirect the user instead of using an HTML area on the page.

The following example shows the type of `PeopleCode` that you could use:

```
&Res = SwitchUser(QE_SWITCHUSER.QE_SWITCH_USER, QE_SWITCHUSER.QE_SWITCH_USER_PWD,⇒
    QE_SWITCHUSER.QE_SWITCHUSERAUTHT, QE_SWITCHUSER.QE_SWITCHUSERINFO);

If (&Res = True) Then
    Transfer( False, MenuName.PROCESSMONITOR, BarName."INQUIRE", Item⇒
Name."PROCESSMONITOR", Panel."PMN_PRCSLIST", "U");
Else
    QE_SWITCHUSER.QE_SWITCHRES = "N";
    QE_SWITCHUSER.QE_PPCAUTHTOKEN = %AuthenticationToken;
    QE_SWITCHUSER.QE_RESULTDOC = %ResultDocument;

    Record.QE_SWITCH_SRCH.QE_CUR_USER.Value = %UserId;
End-If;
```

Overriding Page Encoding

You can override the character set encoding for any page that is registered in the portal. You define a content reference attribute called `PORTAL_ENCODING_OVERRIDE` with a value that specifies the character set to use when processing a page. The portal servlet then overrides all other indicators for encoding, including headers and meta tags.

The standard encoding for PeopleSoft pages is UTF-8. The portal can usually determine the encoding of a page; however, if an external, non-UTF-8 page is not displayed correctly, overriding the page encoding may help.

See Also

PeopleTools 8.52: Global Technology, "Selecting and Configuring Character Sets and Language Input and Output"

Importing Menu Groups into the Portal Registry

This section provides an overview of menu group importing and discusses how to:

- Import menu groups.
- Select components to include in portal navigation.

Understanding Menu Group Importing

The Menu Import Application Engine process (PORTAL_MENU) imports menu groups from the menu definition into a portal registry.

Important! Only system administrators should perform this process. If the user does not have security to the menus and pages, just a folder titled *Label* gets created.

The process creates folders and content references that reflect the menu definition hierarchy. This table shows the folder hierarchy that is imported:

<i>Level</i>	<i>Menu Item</i>
Folder level one	Menu group
Folder level two	Menu name
Folder level three	Bar name
Content reference	Component name

Note. Use the Menu Import process to migrate custom menus only. The navigational menus that are delivered with your PeopleSoft application have already been added to the appropriate portal registry. The Menu Import process has no effect on components that are already registered. However, you can use the process to pick up additional components that are not delivered with the PeopleSoft system.

Before you use the Menu Import process, note these points:

- Object names that are assigned by the Menu Import process are autogenerated.

If a folder name or content reference name already exists, the system appends a number to the generated object name. Consequently, you may have folder object names such as USE, USE1, USE2, and USE3. As a result, use the registration wizard if you plan to copy the application database registry structures into PeopleSoft Application Portal. When you use the registration wizard, you can control the object naming, and you can ensure the uniqueness of object names across databases.

- If a component resides in multiple menus, the process creates a separate content reference for each menu and component combination.
- The navigation hierarchy created by the Menu Import process does not match the delivered PeopleSoft hierarchy.

To keep the navigation consistent, consider putting all custom menus into a common menu group and generating the registry structures for that menu group. Then, once the registry structures are generated, manually move the content references to an appropriate folder in the existing navigation hierarchy.

To create a common menu group, open the custom menu in PeopleSoft Application Designer. On the Use tab of the Menu properties dialog box, enter a unique name in the Menu Group field. After you save the menu, it is available for other menus to use.

Using the Menu Import Process with PeopleSoft Application Portal

If you run the Menu Import process against multiple application databases and merge multiple registries into PeopleSoft Application Portal, bar names (level-three folders) can be overwritten because they frequently have duplicate names. The last folder that is copied takes precedence over all previous folders.

The autogenerated object names are based upon the current database portal registry. The Menu Import process cannot check external databases for existing object names, but it does check the current database.

If you run this process on different databases, the different databases may likely contain identical folder or content reference object names that refer to different objects. This can become problematic if you decide to copy these portal registry structures into PeopleSoft Application Portal. The same-named objects overwrite each other. You might need to delete the generated folders and content references to make them unique in the future.

Use the registration wizard to register menus and components into the portal registry. Although with the registration wizard you must register items one by one, you can control folder and object names at the time of the import.

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Using the Registration Wizard."

Page Used to Import Menu Groups

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Menu Import	PORTAL_MENU_EXPORT	PeopleTools, Portal, Menu Import	Migrate custom menus.

Importing Menu Groups

Access the Menu Import page. (Select PeopleTools, Portal, Menu Import.)

Menu Import

Run Control ID: 2

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

General Settings

Portal Name:

EMPLOYEE

*Node Name:

QE_LOCAL

Template Name:

DEFAULT_TEMPLATE

8.50 default template

Menu Group Selection

☐ All Menu Groups

Menu Group:

Optional

*Valid from date:

07/23/2009

Product:

Valid to date:

Menu Import page

The Menu Import process creates content references for all components that belong to the menu group. This process should be used only to migrate custom menus.

Node Name	<p>Select the node of the content provider where the registry structure resides. The default value is the currently running node.</p> <hr/> <p>Note. Never select the local node for an application database. If you copy the generated content references to a PeopleSoft Application Portal database, the content reference assumes that the component resides in the Application Portal database, when it does not.</p> <hr/>
Template Name	<p>(Optional) Specify the name of a template for the portal servlet to use. If you leave this field blank, the servlet uses the template that is associated with the selected node. If a template is not associated with the node, the default template is used for all content references that are created by the import process. The default value is the portal's default template.</p> <hr/> <p>Note. Best practice is to not specify a template so that you can perform a mass template change at the content provider node level instead of having to change each content reference template individually.</p> <hr/>
All Menu Groups	Select to import all menu groups.
Menu Group	Select a menu group to import.

Valid from date and Valid to date	<p>Specify dates to make visible content references and folders that are associated with the menu that you are importing.</p> <p>In general, use January 1, 1900 for the valid from date and leave the valid to date blank unless you need to limit the time frame.</p>
Product	<p>Enter a unique, 4-character identifier for the menu.</p> <p>If you perform queries against the database, you can use this identifier to identify items that are associated with the menu.</p>
Run	<p>Click to begin the import process.</p> <p>PeopleSoft Process Scheduler launches an Application Engine program that imports the selected menu group definitions into the portal registry. This process typically takes several minutes, but can require more time depending on the number of items to be processed.</p> <p>Permission list settings are initially set to conform with the corresponding permissions set for the menu groups being imported. Once the menu group definitions are imported into the portal registry, no dynamic connection exists between the menu group definition in PeopleSoft Application Designer and the portal registry. If you change the security for the menu or component, the registry security is synchronized as long as the menu or component is in the same database as the registry structure. If you change the menu or component name, you must manually update the registry structure.</p>
Process Monitor	Click to view the status of the menu import.

After you run the process, click the Home button in the universal navigation header and verify that the menu group has been imported into the portal registry. The new menu group should now appear along with the other folders at the top level.

Note. If the menu group does not appear, open the folder definition for the menu group and define a label for the folder.

Selecting Components to Include in Portal Navigation

During the menu import process, you can control which components are included in your portal navigation system.

To select components to include in portal navigation:

1. From PeopleSoft Application Designer, open the appropriate component.
2. Select File, Component Properties, and select the Use tab.

3. Select the Include in Navigation check box.

After a menu import, the component is visible to users with appropriate permissions. (You can still change the permissions for any content reference after the menu import as necessary.)

If you deselect the Include in Navigation check box, the component is not imported into the portal registry during the Menu Import process. You can also set this option at the menu level.

4. Click OK.

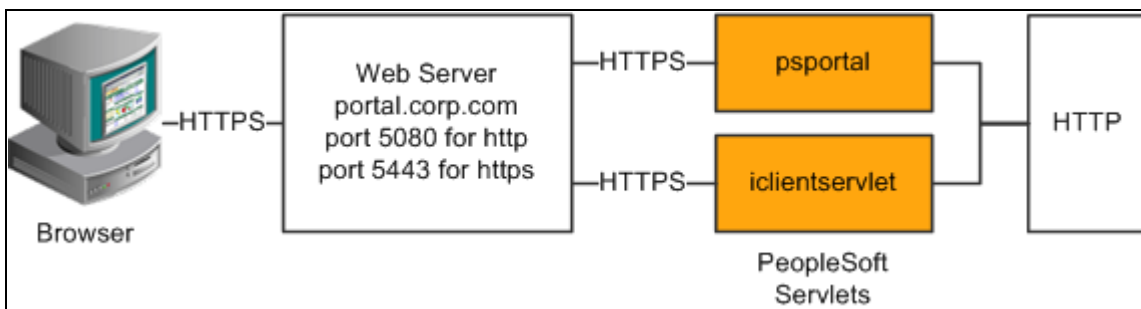
Changes that you make to the component in PeopleSoft Application Designer are not reflected in the portal until you do another menu import.

See Also

PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide, "Using PeopleSoft Application Designer"

Improving Same-Server Performance Under SSL

You might run portal and PeopleSoft Pure Internet Architecture applications over SSL connections to secure the transmissions between the browser and the web server. However, defining each homepage pagelet as an HTTPS request can cause portal performance to degrade significantly. To achieve secure transmissions between the browser and the server and to achieve acceptable homepage performance, configure the portal to use HTTP connections whenever it needs to talk to a PeopleSoft Pure Internet Architecture application that is hosted on the same server as the portal. This diagram shows this setup:



Example of the portal running under SSL by using HTTP connections to the PeopleSoft Pure Internet Architecture on the same server

This configuration requires changes to the PeopleTools web configuration properties file. It also requires that a WebServer filter be set up to block non-SSL connections, except those from the portal. The portal does not need to use HTTP or HTTPS to connect to the PeopleSoft Pure Internet Architecture servlet hosting the same website. This configuration is needed only when multiple sites are used on the same web server.

Web Profile Settings

You must tell the portal to use HTTP for all connections that it makes back to the same server on which it is hosted. To do so, modify the web profile configuration for the portal. Change these properties:

Web Profile Page	Field Name	Field Value
Security	PIA use HTTP Same Server	Selected.
Virtual Addressing	Protocol(in the Default Addressing region of the page)	<i>HTTPS.</i>
Virtual Addressing	Port(in the Default Addressing region of the page)	The port on which the web server is listening for HTTPS requests, for example, <i>5443</i> . You can leave this field blank if you use the default SSL portal of 443.

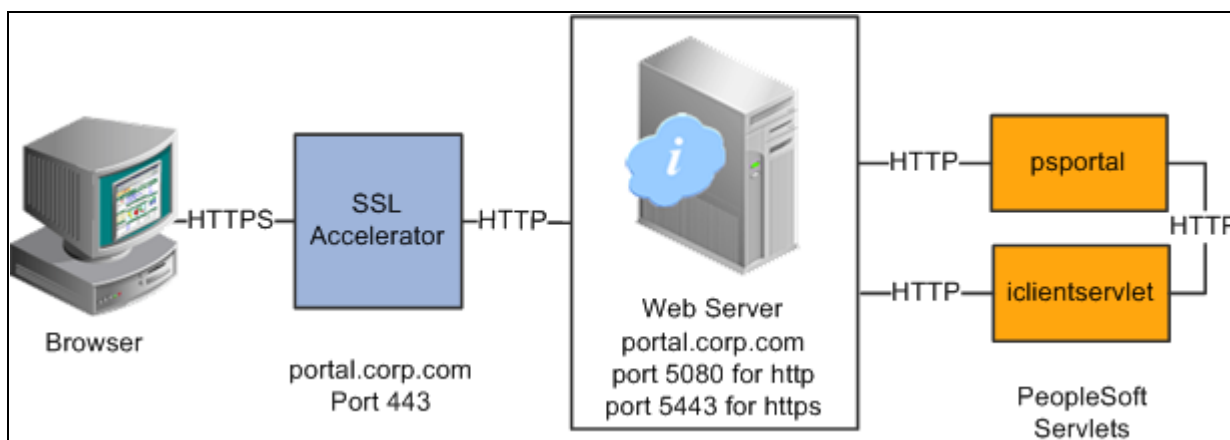
See Also

Chapter 9, "Configuring the Portal Environment," Configuring Web Profiles, page 238

Chapter 3, "Understanding Portal Technology," Proxy Architecture and Relative URLs, page 37

Using SSL Accelerators

You can improve SSL performance by adding an SSL accelerator in front of the web server. You need to have the portal use HTTP for requests to the PeopleSoft Pure Internet Architecture to prevent the portal connection to the PeopleSoft Pure Internet Architecture from going back through the accelerator. This diagram shows this setup:



Using an SSL accelerator

The configuration is similar to that for configuring a portal running under SSL to use HTTP connections to the PeopleSoft Pure Internet Architecture on the same server; however, you must change the default port to point to the port on which the SSL accelerator is configured to listen.

Important! You must continue to configure Web Server to listen for HTTPS requests, even though the accelerator intercepts all SSL connections. Failure to do so causes the portal to function incorrectly.

Web Profile Settings

Change these properties:

Web Profile Page	Field Name	Field Value
Security	PIA use HTTP Same Server	Selected. (This might reduce the amount of time it takes to make an HTTP request to the portal.)
Virtual Addressing	Protocol(in the Default Addressing region of the page)	<i>HTTPS</i> .
Virtual Addressing	Port(in the Default Addressing region of the page)	The port on which the SSL accelerator is listening for HTTPS requests, for example, <i>443</i> .

See Also

Chapter 9, "Configuring the Portal Environment," *Configuring Web Profiles*, page 238

PeopleTools 8.52: System and Server Administration, "Working with Oracle WebLogic"

PeopleTools 8.52: System and Server Administration, "Working with IBM WebSphere"

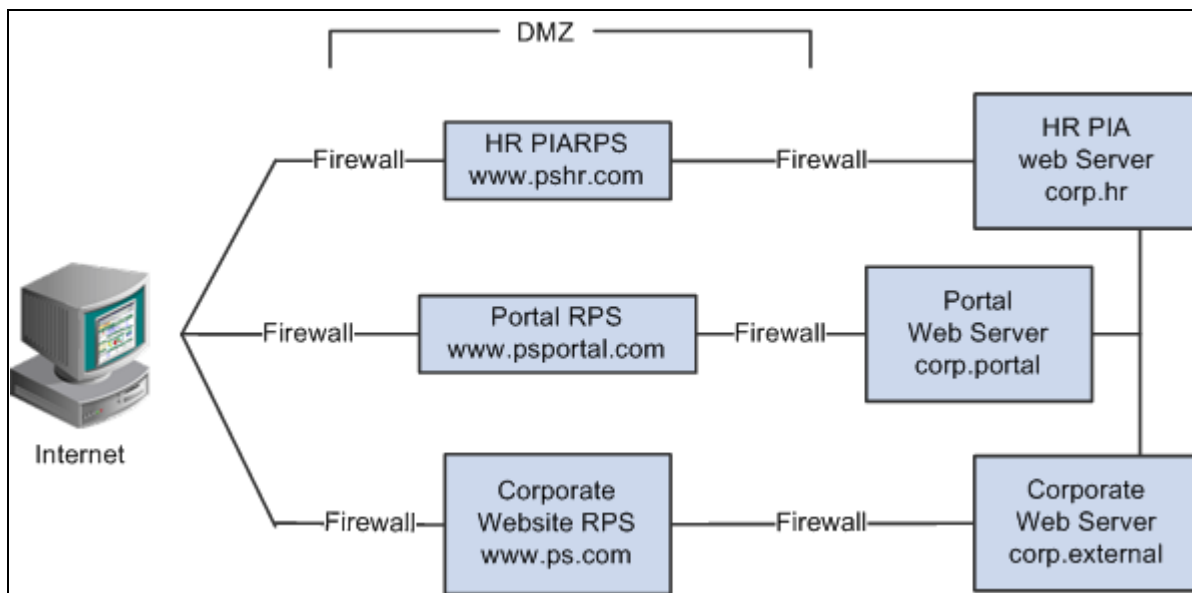
Using Reverse Proxy Servers

This section provides an overview of reverse proxy servers and discusses how to:

- Make reverse proxy settings.
- Issue requests by using a proxy server.
- Specify hosts that should not be proxied.

Understanding Reverse Proxy Servers

A reverse proxy server accepts requests from client browsers on behalf of another web server and then relays those requests to the web server. The reverse proxy server checks web requests and responses to ensure that they adhere to network security policies. For example, a reverse proxy server validates that a command intended for the target web server is a valid command for that server. In this diagram, the human resources PeopleSoft Pure Internet Architecture web server, corp.hr, is behind a reverse proxy server, www.pshr.com:



Example of the reverse proxy architecture

All internet requests for the human resources web server are relayed by `www.pshr.com` to `corp.hr`. The site URL routes the client to the proxy as if it were a web server. Replicated content is delivered from the proxy cache to the external client without exposing the origin server or the network residing safely behind the firewall. You can balance the load of an overtaxed web server by installing multiple instances of the web server.

Any external content must come from a node that's registered in the portal database so that the portal servlet can look up the node name based on the URL of the content. The external content HTML is rewritten to use that proxied format.

See Also

[Chapter 9, "Configuring the Portal Environment," Using Firewalls with Reverse Proxy Servers, page 293](#)

[Chapter 9, "Configuring the Portal Environment," Using Reverse Proxy Servers with SSL Accelerators, page 295](#)

Making Reverse Proxy Settings

On the Web Profile - Virtual Addressing page, specify the reverse proxy server to use by entering it in the Reverse Proxy Server List grid. Include the protocol, the host name, the HTTP port, and the HTTPS port.

See Also

[Chapter 9, "Configuring the Portal Environment," Configuring Virtual Addressing, page 254](#)

Issuing Requests by Using a Proxy Server

For you to make a portal issue requests by using a proxy server, the proxy server settings are controlled by these Java system properties:

- `http.proxyHost`
- `http.proxyPort`
- `https.proxyHost`
- `https.proxyPort`

To use a proxy server for the HTTP protocol, set these properties:

```
http.proxyHost=http_proxy_server_host_name
http.proxyPort=http_proxy_server_port
```

To use a proxy server for the HTTPS protocol, set these properties:

```
https.proxyHost=https_proxy_server_host_name
https.proxyPort=https_proxy_server_port
```

The method for setting the Java system parameters varies by web server (Oracle WebLogic or IBM WebSphere). The properties must be passed to the Java interpreter as command-line directives.

For Oracle WebLogic

Alter the `setEnv.cmd` or `setEnv.sh` file located in WebLogic domain's (PIA Install) bin folder. Define the properties on the command line that starts the Java Virtual Machine like this:

```
set PROXY=-Dhttp.proxyHost=proxyhostname -Dhttp.proxyPort=proxy_port⇒
-Dhttps.proxyHost=proxyhostname -Dhttps.proxyPort=proxy_port
```

Later add the preceding property in `startPIA.cmd` or `startPIA.sh` in the Java command line that starts the WebLogic server domain.

```
%JAVA_HOME%\bin\java -ms64m -mx64m
-classpath %JAVA_CLASSPATH% %PROXY%
-Dweblogic.class.path=%WEBLOGIC_CLASSPATH%
-Dweblogic.home=. -Djava.security.manager⇒
-Djava.security.policy==.\weblogic.policy weblogic.Server

goto finish
```

For IBM WebSphere

All the Java Virtual Machine settings for IBM WebSphere are set in

```
<was_home>\profileTemplates\default\documents\config\templates\servertypes\APPLICATION_SERVER\s
ervers\default.
```

Note. Here, the profile type is *default* and WebSphere has four types of profiles: cell, default, dmgr, and managed. The preceding path varies based on the type of profile.

Java Virtual Machine properties are embedded within the `<jvmEntries>` and `</jvmEntries>` elements.

```
<jvmEntries xmi:id="JavaVirtualMachine_1164791211226" verboseModeClass="false"=>
  verboseModeGarbageCollection="false" verboseModeJNI="false" runHProf="false"=>
  hprofArguments="" debugMode="false" debugArgs="-Djava.compiler=NONE -Xdebug ->
Xnoagent -Xrunjdw:transport=dt_socket,server=y,suspend=n,address=7785" generic=>
JvmArguments="-Dprofile.name=peoplesoft_auth">
  <systemProperties xmi:id="Property_1164791897101" name="ps_vault" value="c:=>
/pt849-802-r2-debug/webserve/peoplesoft_auth/installedApps/peoplesoft_authNodeCell=>
/peoplesoft_auth.ear/psvault" description="Vault location"/>
  <systemProperties xmi:id="Property_1164791897102" name="HttpSessionIdReuse"=>
value="false" description="HttpSessionIdReuse"/>
</jvmEntries>
```

Specifying Hosts That Should Not Be Proxied

If your portal implementation retrieves content from both an intranet and the internet, you may have a problem retrieving both internal and external content if the proxy server is not aware of the address of the intranet content. You can prevent this situation by specifying that certain hosts (such as your intranet hosts) should not be accessed through the proxy server.

For Oracle WebLogic, specify the hosts not to be proxied in the `HTTP_PROXY_NONPROXY_HTTPHOSTS` and `HTTP_PROXY_NONPROXY_HTTPSHOSTS` environment variables with the `setEnv.cmd` (`setEnv.sh` in Unix) script. Refer to comments in `setEnv` for specific syntax.

For IBM WebSphere, specify the hosts not to be proxied in the Java system variables `http.nonProxyHosts` and `https.nonProxyHosts` in the IBM WebSphere console:

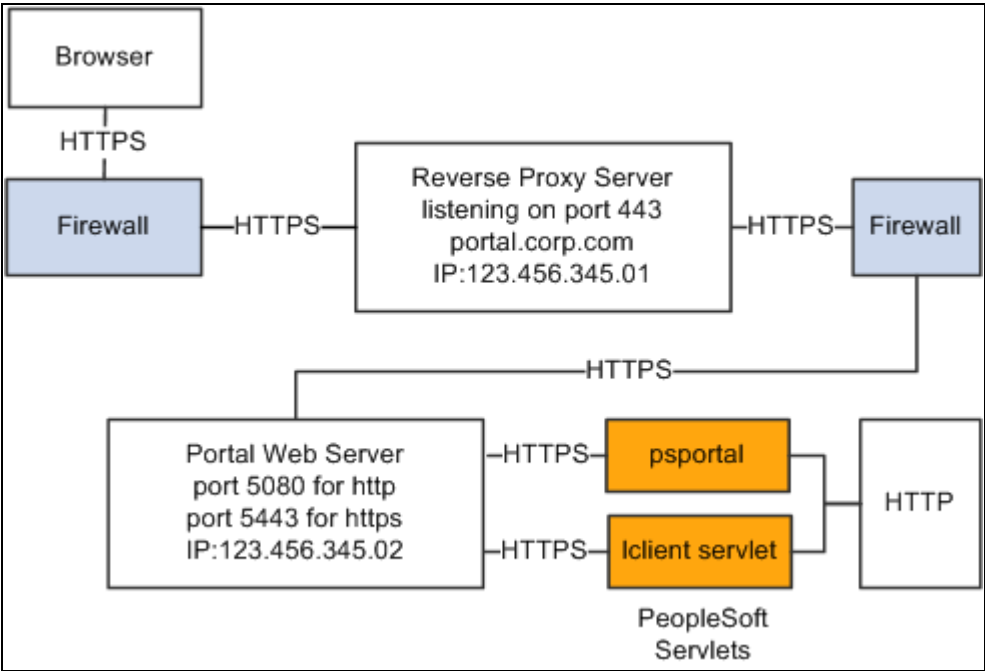
1. Access the IBM WebSphere administration console.
2. Select Application Servers, *server_name*, Process Definition, Java Virtual Machine, Custom Properties.
3. Click New.
4. Set the property `http.nonProxyHosts`: List the internal machines for which authentication is not required for routing through the proxy. Separate each machine name in the list with a vertical bar (|).
5. Save the configuration, then sign out and restart IBM WebSphere.

For all the web server platforms, you can list a set of patterns that should not be accessed through proxies, with each one separated by the pipe character (|). This example shows a setting that makes all HTTP connections to the eng domain bypass the proxy settings:

```
*.eng|*.eng.peoplesoft.com
```

Using Firewalls with Reverse Proxy Servers

You can set up a buffer zone in front of the portal web server to prevent unauthorized access to the portal web server and create a more secure environment. A buffer zone is typically configured with a firewall that allows access to a reverse proxy server, which relays incoming requests through a second firewall to the portal web server. This diagram shows this setup:



Firewall and reverse proxy setup

The configuration is similar to that for configuring a portal running under SSL to use HTTP connections to the PeopleSoft Pure Internet Architecture on the same server; however, you must make these adjustments in the web profile:

Web Profile Page	Field Name	Field Value
Security	PIA use HTTP Same Server	Selected.
Virtual Addressing	Protocol(in the Default Addressing region of the page)	HTTPS.
Virtual Addressing	Name(in the Default Addressing region of the page)	The DNS name of the reverse proxy server, for example, portal.corp.com.
Virtual Addressing	Port(in the Default Addressing region of the page)	The port on which the reverse proxy server is listening for HTTPS requests if it's different from the port on which the web server is listening.

Hosts File Setup

The hosts file on the portal web server must have an entry that directs DNS requests for the content provider server name (the reverse proxy server in the previous example) to the portal web server, for example, 123.456.345.02. This enables the portal to make requests for content that is hosted on the same server directly without going back through the reverse proxy server.

For example:

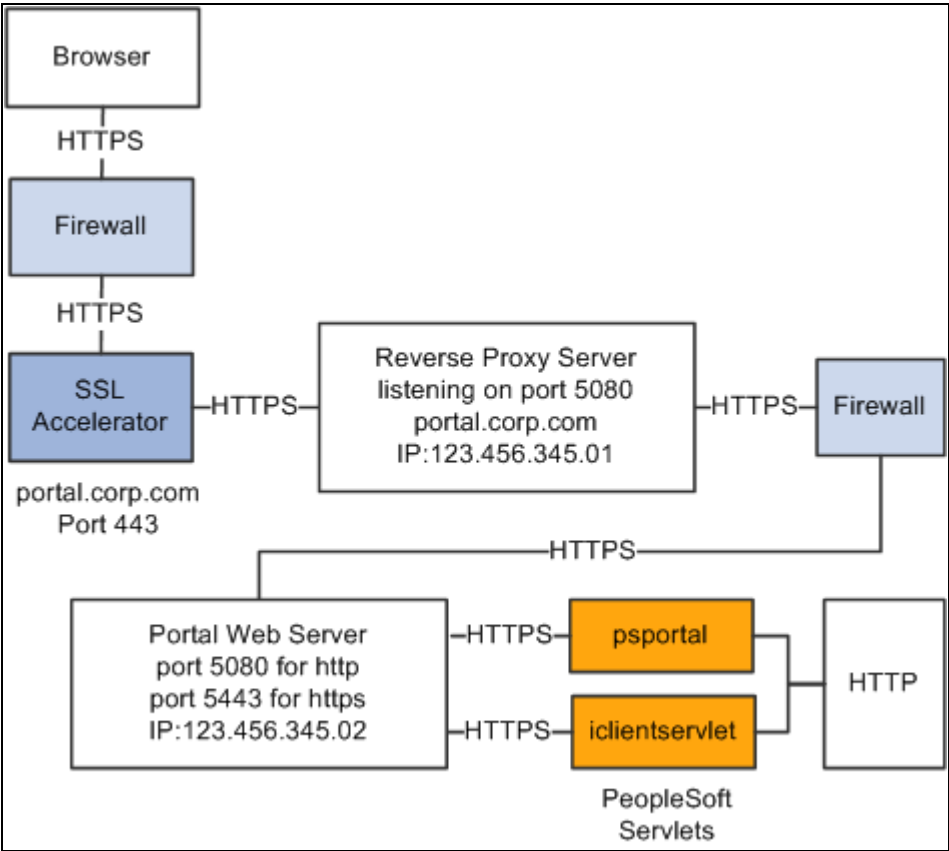
123.456.345.02 portal.corp.com

See Also

Chapter 9, "Configuring the Portal Environment," [Configuring Web Profiles](#), page 238

Using Reverse Proxy Servers with SSL Accelerators

You can combine an SSL accelerator with a reverse proxy server, as illustrated in this diagram:



Reverse proxy server with an SSL accelerator

The configuration is similar to that for configuring a portal running under SSL to use HTTP connections to the PeopleSoft Pure Internet Architecture on the same server; however, you must make these adjustments in the web profile:

Web Profile Page	Field Name	Field Value
Security	PIA use HTTP Same Server	Selected.

Web Profile Page	Field Name	Field Value
Virtual Addressing	Protocol(in the Default Addressing region of the page)	Set to <i>HTTPS</i> .
Virtual Addressing	Name(in the Default Addressing region of the page)	The DNS name of the reverse proxy server, for example, <i>portal.corp.com</i> .
Virtual Addressing	Port(in the Default Addressing region of the page)	The port on which the SSL accelerator is listening for HTTPS requests, for example, <i>443</i> .

Hosts File Setup

The hosts file on the portal web server must have an entry that directs DNS requests for the content provider server name (portal.corp.com) to the portal web server, for example, 123.456.345.02. This enables the portal to make requests for content that is hosted on the same server directly without going back through the reverse proxy server.

For example:

123.456.321.02 portal.corp.com

See Also

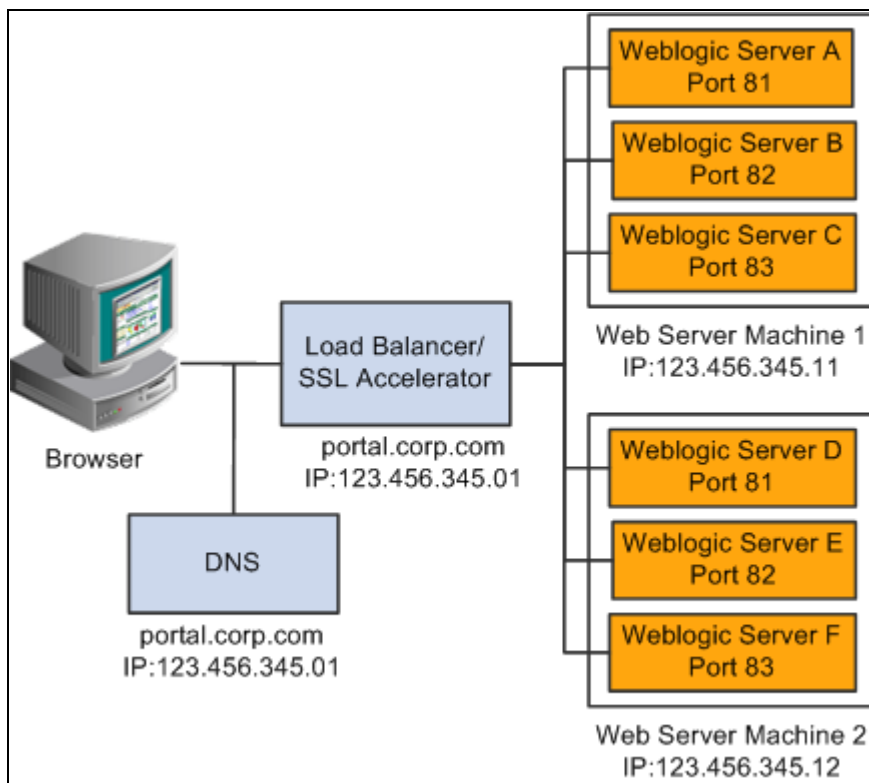
Chapter 9, "Configuring the Portal Environment," Configuring Web Profiles, page 238

PeopleTools 8.52: System and Server Administration, "Working with Oracle WebLogic"

PeopleTools 8.52: System and Server Administration, "Working with IBM WebSphere"

Applying Load Balancing

You can apply load balancing to a portal by using a hardware load-balancing solution to distribute load across multiple instances of the web server on multiple web server machines. Many of today's hardware load balancers have SSL accelerating options that you should also consider. The number and size of web server machines and the number of web server instances per machine vary and determine load-balancing requirements and hardware selection. This diagram shows a possible load-balancing setup:



Example of load balancing

The hardware load balancing solution must be configured to enable sticky sessions, which ensures that all requests from a user's browser session are delivered to the same web server session. In the example previous diagram, if the initial request from the browser was directed to Oracle WebLogic Server F, then all subsequent requests from that browser session must be directed to the same instance of Oracle WebLogic.

Keep multiple web server sessions on a single machine to keep the heap size of the Java Virtual Machine manageable and to allow the Java garbage collection to work optimally.

Hosts File Configuration

The hosts file on the portal web server machines in the previous diagram—which we'll call portalws1 and portalws2—must have an entry that directs DNS requests for the portal content provider server name (portal.corp.com) to the portal web server (123.456.345.11 on portalws1). This allows the portal to request content hosted on the same server directly without going back through the load balancer.

This is an example hosts file on portalws1.corp.com:

```

123.456.345.11 portal.corp.com
123.456.345.11 portalws1.corp.com
  
```


Part 3

Working with Homepages, Dashboard Pages, and Pagelets

Chapter 10

Administering Homepages, Dashboard Pages and Pagelets

Chapter 11

Using Pagelet Wizard to Create and Manage Pagelets

Chapter 12

Developing and Registering Pagelets

Chapter 13

Configuring OBIEE Report Pagelets

Chapter 10

Administering Homepages, Dashboard Pages and Pagelets

This chapter provides an overview of homepages and dashboards and discusses how to:

- Manage tabbed homepages.
- Manage dashboard pages.
- Use attributes to enhance pagelets.

Note. In this chapter, the terms *homepage* and *tabbed homepage* are used interchangeably. While PeopleTools regards the term *tabbed homepage* as the more accurate term, in this document the terms are equivalent.

In this chapter, the terms *dashboard page* and *dashboard* are equivalent.

See Also

PeopleTools 8.52: PeopleSoft Applications User's Guide, "Working With PeopleSoft Applications"

PeopleTools 8.52: PeopleSoft Applications User's Guide, "Setting User Preferences," Personalizing Your Homepage

Understanding Homepages and Dashboard Pages

Homepages, or tabbed homepages, and dashboard pages are PeopleSoft pages that aggregate and display pagelets that share some common or similar purpose. Homepages and dashboard pages provide quick access to useful information by presenting concise but feature-rich pagelets to the user. These pagelets can vary in function and complexity. Pagelets can enable you to perform routine tasks or provide you with a high-level overview of aspects of your company that are pertinent to you. Pagelets can interact with each other by exchanging data and then refreshing pagelet content based on any passed values. This exchange process, called interwindow communication, enables you to see not only an overview, but to drill into your data to access more granular and detailed information.

Note. Available homepage and dashboard functionality differs based on the products that you license.

Homepages

Note these points about homepages:

1. Homepages appear immediately after users sign in to PeopleSoft applications or when the user clicks the Home link.
2. PeopleSoft applications can display one or more homepages.
3. If you create multiple homepages, each displays a tab at the top to provide access to the homepage.
4. If you enable personalization, users can personalize homepages. The application stores each user's homepage personalizations in the PSPRUHDEFN (which is updated when a user personalizes the homepage), PSPRUHTAB, and PSPRUHTABPGLT tables.

You manage homepages from the Portal Objects folder in the portal registry, which you access by using the Structure and Content component. Access to this component gives any user unrestricted access to the entire portal registry. Rarely should anyone other than the portal administrator have access to Structure and Content.

Dashboards

Note these points about dashboard pages:

- Dashboard pages are a type of homepage and are technically called *operational dashboards*.
- Dashboards do not appear automatically after sign in, instead you access a dashboard page by clicking a content reference link, which you can configure to appear anywhere in the menu structure.
- PeopleSoft applications can have multiple dashboards.
- If you enable personalization, users can personalize dashboard pages. The application stores each user's dashboard page personalizations in the PSPRUHDEFN (which is updated when a user personalizes the dashboard), PSPRUHTAB, and PSPRUHTABPGLT tables.

To create and manage dashboards, you do *not* need access to the Structure and Content component. You can use the Manage Dashboard Pages component. No access to the structure and content of the portal registry is necessary. By eliminating the need to access the portal registry to manage dashboard pages, your PeopleSoft application offers increased flexibility of administration and decreased chance of down time due to portal registry mismanagement errors.

Note. When you use the Manage Dashboard Pages component to create new dashboard pages, the application automatically registers the dashboard as a homepage in the portal, places the content reference in the Tab folder, and selects the Operational Dashboard check box. No functionality has been removed from the Structure and Content pages; you can use them to create and manage homepages that you specify as operational dashboard pages.

Managing Homepages

This section discusses how to:

- Create homepages.
- Configure homepages.
- Set homepage permissions.
- Select the pagelets that appear on homepages.

- Arrange pagelets on homepages.
- Rearrange the homepage tab sequence.
- Troubleshoot homepages.

Pages Used to Manage Homepages

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Content Ref Administration	PORTAL_CREF_ADM	Select PeopleTools, Portal, Structure and Content. Click these links in the order presented: Portal Objects, Homepage, Tabs. Click Add Content Reference link.	Registering homepages in the portal registry.
Content Reference Security	PORTAL_CREF_SECURITY	Select the Security tab.	Set homepage permissions.
Tab Content	PORTAL_TAB_CON_ADM	PeopleTools, Portal, Structure and Content, Portal Objects, Homepage, Tabs, Content Ref Administration, Tab Content	Select the pagelets that appear on homepages.
Tab Layout	PORTAL_TAB_LAY_ADM	PeopleTools, Portal, Structure and Content, Portal Objects, Homepage, Tabs, Content Ref Administration, Tab Layout	Arrange pagelets on homepages.

Registering Homepages in the Portal Registry

Access the Content Ref Administration page. (Select PeopleTools, Portal, Structure and Content. Click the Portal Objects link. Click the Homepage link. Click the Tabs link. Click the Add Content Reference link).

Content Ref Administration page

To create a homepage entry in the portal registry:

Note. You must create the homepage or dashboard in the Tabs folder.

1. Enter a name and label for the homepage or dashboard.

Note. The label value appears as the homepage tab, dashboard name and content reference link label.

2. In the Usage Type field, select *Homepage tab*.

After you select *Homepage tab*, the page refreshes and two additional pages, the Tab Content and Tab Layout appear.

3. In the Valid from date field, enter the date that you want the tab to first appear for your users. Optionally, enter a valid to date to remove the tab from use.
4. If you want the homepage to be a dashboard page, select the Operational Dashboard check box in the Homepage tab attributes group box.

5. Set the homepage permissions.

See [Chapter 10, "Administering Homepages, Dashboard Pages and Pagelets," Setting Homepage Permissions, page 306.](#)

6. Define the homepage content and layout.

See [Chapter 10, "Administering Homepages, Dashboard Pages and Pagelets," Selecting the Pagelets That Appear on Homepages, page 308.](#)

See [Chapter 10, "Administering Homepages, Dashboard Pages and Pagelets," Arranging Pagelets on Homepages, page 309.](#)

7. Click the Save button.

8. If you selected the Operational Dashboard check box, create a content reference link to the dashboard in the navigational hierarchy.

See [Chapter 5, "Administering Portals," Defining Content Reference Links, page 160.](#)

This table describes the fields that appear on the Content Ref Administration page.

Label	Enter the label that the user sees on the homepage tab, as the dashboard page title, and as the content reference link in the navigation.
Name	Enter a unique name for the tab. The value that you specify in the Name field can consist of up to 50 letters, digits, and underscores (_). It cannot contain any spaces and cannot begin with a digit.
Sequence number	Enter a value to designate the absolute position of the homepage when multiple homepages exist for users. If you omit the Sequence number, the portal registry entry for the homepage is added to the top of the content reference list and at runtime, the portal displays the homepage tabs in alphabetical order. You can change the sequence only one item at a time, and each time you do so, you must save your changes to see the result.
Usage Type	Select <i>Homepage tab</i> to create a tabbed homepage or dashboard. When you select <i>Homepage tab</i> , two additional pages appear in the component: the Tab Content and Tab Layout pages.
Valid from date	Enter the date that the tab becomes visible to the end user. You can enter a date to remove the tab from display.

Homepage Tab Attributes

Use the Homepage tab attributes group box of the Content Ref Administration page to enable users to rename the homepage tab label, as well as associate a help topic with the homepage tab, hide the pagelet action bar, and designate a homepage as an operational dashboard.

Allow rename	Select to enable the application user to change the homepage tab label.
---------------------	---

Help ID

Enter the help ID for the pagelet. When users click the Help button on the pagelet, the system displays help information that is specific to the pagelet.

This feature works only if you specify a help URL on the Web Profile Configuration - General page, and the pagelet documentation is part of the HTML PeopleBooks identified by the help URL.

You must also ensure that the documentation HTML includes a properly formatted anchor element that uses the value that you specify in this field. For example, if you specify a help ID of *MY_PAGELET_CONTENT*, the pagelet documentation in the PeopleBook must contain the following element:

```
<a NAME="F1ID_MY_PAGELET_CONTENT"></a>
```

Hide pagelet action bar

Select to hide the pagelet headers on all pagelets on the homepage tab. This option overrides all properties for pagelets used on this tab.

Operational Dashboard

Select to use the homepage as a dashboard page, which hides the homepage so that it does not appear when the users signs in.

If you select this check box, you must create a content reference link to the dashboard, which you position in the menu navigation as your business rules require.

If you use the Dashboard Administration pages to create dashboards, then the system creates the homepage content reference, selects this check box, and creates the link automatically.

See Also

[Chapter 5, "Administering Portals," Setting Content Reference Permissions, page 157](#)

[Chapter 10, "Administering Homepages, Dashboard Pages and Pagelets," Managing Dashboard Pages, page 312](#)

Setting Homepage Permissions

Access the Content Reference Security page (Select PeopleTools, Portal, Structure and Content. Then click the Portal Objects link, the Homepage link, and the Tab link in that order. Click the Edit link for the homepage tab, then click the Security tab).

Content Reference Security

Label: My Page

☐ Public
☐ Author Access

Security Authorizations					
Personalize Find First 1-2 of 2 Last					
	Type	Name	Description	View Definition	
1	Permission List	PTPT1000	PeopleSoft User	View Definition	+ -
2	Role	Employee	Employee	View Definition	+ -

Inherited Security Authorizations			
Personalize Find First 1 of 1 Last			
Type	Name	Description	View Definition
			View Definition

Content Reference Security page

Public

Select to make the homepage or dashboard available to all users. When this check box is selected, all other fields except Label are hidden.

Author Access

Select to indicate that you (the currently signed-in user and the author of the homepage) can access the folder in the future, regardless of how other permissions are set.

Security Authorizations

You can base homepage security on roles or permission lists. You can specify any combination of roles and permission lists that allow homepage access.

Type

Select from these types:

Permission List: Enables access for users based on permission list membership. You then specify the permission lists that should have access to this folder.

Role: Enables access for users based on role membership. You then specify the roles that should have access to this folder.

Name

Select the permission list or role that you want have access to this homepage.

View Definition

Click to access the security administration component for the permission list or role specified on the current row.

Selecting the Pagelets That Appear on Homepages

Access the Tab Content page. (Select 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.

QE

☒ Include all?

<input checked="" type="checkbox"/> Display Cookies	Optional
<input checked="" type="checkbox"/> QE Portal Links	Optional
<input checked="" type="checkbox"/> QE Redirect Test	Optional
<input checked="" type="checkbox"/> QE_COMMAINC COOKIE	Optional
<input checked="" type="checkbox"/> QE_Counter	Optional
<input checked="" type="checkbox"/> Signin	Optional
<input checked="" type="checkbox"/> Test Global Variables	Optional
<input checked="" type="checkbox"/> Who Am I	Optional
<input checked="" type="checkbox"/> User-based Cache	Optional
<input checked="" type="checkbox"/> Role-based Cache	Optional
<input checked="" type="checkbox"/> Application-based Cache	Optional

PeopleSoft Applications

☒ Include all?

<input checked="" type="checkbox"/> Main Menu	Optional
<input checked="" type="checkbox"/> Menu	Opt-Dfit
<input checked="" type="checkbox"/> Menu - Classic	Optional
<input checked="" type="checkbox"/> My Reports	Optional
<input checked="" type="checkbox"/> Top Menu Features Description	Optional
<input checked="" type="checkbox"/> XML Publisher	Optional

Finance

☐ Include all?

<input type="checkbox"/> Yahoo Stock Quote	Optional
--	----------

Tab Content page

To select pagelets for homepages and dashboards:

1. Select the check box for the pagelet to appear on the homepage or dashboard.

Select the Include All? check box to select all pagelets in the category.

Note. The user must have security access to the pagelet component for the pagelet to be properly displayed on the homepage.

2. Use the drop-down list box to select the pagelet behavior. Values are:

<i>Optional</i>	The pagelet does not appear automatically on the homepage. However, it is available for users when they are personalize their homepages.
<i>Opt-Dflt</i> (option-default)	The pagelet appears on all user homepages if the users can access the pagelet. The pagelet can be removed when users personalize the homepage.
<i>Req-Fix</i> (required-fixed)	The pagelet appears on all user homepages if users can access the pagelet. The placement of the pagelet cannot be changed, and it also cannot be removed.
<i>Required</i>	The pagelet appears on all user homepages if users can access the pagelet. The placement of the pagelet can be changed, but it cannot be removed.

3. Click the Save button.

Note. Content outside of PeopleSoft applications is available only with PeopleSoft Application Portal.

Important! When you select content for inclusion 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.

Arranging Pagelets on Homepages

Access the Tab Layout page. (Select the Tab Layout tab.)

General Security Tab Content **Tab Layout**

Root > Portal Objects > Homepage > Tabs >

Tab Layout

Label: My Page

Basic Layout: ☐ ☒

2 columns 3 columns

Click arrows to move pagelets up and down or into neighboring columns. Click "Delete Pagelet" to remove the selected pagelet from the homepage tab definition.

= Required - fixed position pagelet
* = Required - moveable pagelet

Left Column:	Center Column:	Right Column:
Menu	--No Pagelets Selected--	--No Pagelets Selected--

Navigation arrows and Delete Pagelet button.

Tab Layout page

Label	Displays the label information from the Content Ref Administration page.
Basic Layout	<p>Select to display the pagelets in either two or three columns on the homepage.</p> <p>To specify the three-column layout, be sure that at least one pagelet does not have the <i>Req-Fix</i> behavior option selected on the Tab Content page.</p>
Columns	<p>Displays the pagelets selected on the Tab Content page in the column sections.</p> <p>If the basic layout is two columns, the pagelets are divided into left and right columns. If the basic layout is three columns, the pagelets are divided into left, center, and right columns.</p> <hr/> <p>Note. A pound sign (#) indicates a pagelet with the <i>Req-Fix</i> behavior option selected on the Tab Content page. An asterisk (*) indicates a pagelet with the <i>Required</i> behavior option selected on the Tab Content page.</p> <hr/> <p>For example, #Signon indicates that the Signon pagelet cannot be moved to a different location on the homepage.</p> <hr/>



Use the Move icons to position the pagelets. Highlight a pagelet, and then click the directional arrow buttons to move the selected pagelet above a pagelet, below a pagelet, to the next column to the right, or the next column to the left.

Delete Pagelet

Highlight a pagelet and then click to delete it from the homepage.

To arrange the tab layout:

1. Select a basic layout option.
2. Select the pagelet name.
3. Click the directional arrow buttons to position the pagelets in the columns.
4. Click the Save button.

Note. As the portal administrator, you determine the amount of flexibility that you give users to change the layout of the pagelets. The settings that you select on both the Tab Content and Tab Layout pages determine the degree of flexibility. Grant the end user as much or as little flexibility as your organization demands.

Specifying Homepage Tab Sequence

The sequence number determines the order in which the homepage tabs appear. The tabs appear with the lowest sequence number first. If any numbers are identical, then the order is alphanumeric. If no sequence numbers exist, the default is considered to be zero. For example, if no sequence numbers are entered, the default is zero, and the order is alphanumeric.

To rearrange the tab order:

1. Select PeopleTools, Portal, Structure and Content.
2. Click the Portal Objects link.
3. Click the Homepage link.
4. Click the Tabs link.
5. Click the Edit link for each tab that you want to reorder.
6. Enter a number in the Sequence field.

If caching is turned on for the portal, the tab order change does not take effect until the caching recycles or the web server is restarted.

Troubleshooting Homepages

Under most circumstances, PeopleTools portal technology assembles homepages by simultaneously retrieving content for all the pagelets referenced on the homepage. As a result, the portal renders the homepage as soon as the user signs in and immediately begins loading individual pagelets. Under certain circumstances, pagelets are not loaded simultaneously, but sequentially.

Pagelet Loading

Portal homepages load each pagelet individually when you sign in. The pagelet appears as you see in this example, indicating that it's currently loading its data.



Pagelet loading on homepage

Sequential Pagelet Loading

When `document.write` or `document.writeln` scripts are detected inside a pagelet, the homepage loads all pagelets sequentially. This is required to properly assemble the content of the pagelet. For example, consider a homepage with the following pagelets, each of which is loaded in the time indicated:

- Pagelet A: 0.5 seconds
- Pagelet B: 1.2 seconds
- Pagelet C: 33.5 seconds
- Pagelet D: 2.3 seconds

Note. These pagelets can all have relatively quick average load times, but any pagelet might be excessively slow on occasion.

The total time required to retrieve all the pagelets is 37.5 seconds, which is disproportionately skewed by pagelet C. Some reasons for this could be the URL for pagelet C is incorrect or has changed, the domain name server can't resolve the URL, the content server is down or unreachable, the content server is behind a firewall, or the content server is temporarily unavailable. As a result of such high retrieval times, the total time to render the homepage is unacceptable to the user, and for practical purposes the pagelet is unavailable.

You can respond to unavailable pagelets by specifying a time-out period during which each pagelet must be loaded. This improves portal homepage performance by enabling the system to present the homepage quickly in spite of an individual pagelet's unavailability. To specify a pagelet time-out, add the `TIME-OUT` attribute to the content reference for the pagelet in the portal registry.

See [Chapter 12, "Developing and Registering Pagelets," Configuring Pagelet Time-out Settings, page 510.](#)

Managing Dashboard Pages

This section discusses how to:

- Create and manage dashboard pages.
- Specify dashboard page properties.

- Specify the location of dashboard pages.
- Specify dashboard page security properties
- Specify the location of the dashboard page link.
- Specify the pagelets that appear dashboard pages.
- Arranging the pagelets on dashboard page.

Pages Used to Manage Dashboard Pages

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Manage Dashboard Pages	PTAL_DASHBD_PAGES	PeopleTools, Portal, Dashboard, Manage Dashboard Pages	Create and manage dashboard pages.
Dashboard Page Administration	PTAL_DASHBD_ADMIN	For new dashboard pages, click the Create new dashboard page link. For existing dashboard pages, click the Properties button.	<ul style="list-style-type: none"> • Specify dashboard page properties. • Specify dashboard security details.
Select the Dashboard Link Parent Folder	PTAL_SELECT_FREF	Click the look up icon for the Parent Folder field.	Specify the location of the dashboard page.
Content	PTSYSTABCONTENT	Click the Pagelets button.	Specify the pagelets that appear on dashboard pages.
Layout	PTSYSTABLAYOUT	Click the Pagelets button. Then, click the Layout tab.	Arrange the pagelets on dashboard pages.

Creating and Managing Dashboard Pages

Access the Manage Dashboard Pages page (Select PeopleTools, Portal, Dashboard, Manage Dashboard Pages.)

Manage Dashboard Pages

Dashboard pages in the current portal are shown below.

Dashboard Pages			
	Dashboard Page	Pagelets	Properties
1	My Dashboard	Pagelets	Properties

[Create new dashboard page](#)

Manage Dashboard Pages page

Create new dashboard page	Click to access the Dashboard Page Administration page and configure the properties of new dashboard pages.
Dashboard Page	Displays an active link to the dashboard page. Click the link to access the dashboard page.
Pagelets	Click to configure the content and layout of the dashboard.
Properties	Click to access the Dashboard Page Administration page and specify dashboard navigation and security details.
Delete	Click to remove the dashboard content reference definition from the portal registry.

Specifying Dashboard Page Properties

Access the Dashboard Page Administration page (Click the [Create new dashboard page](#) link. Alternatively, click the [Properties](#) button).

Dashboard Page Administration

Specify properties and security of the dashboard page.

Portal Name EMPLOYEE

*Dashboard Name

*Dashboard Label

Long Description

Owner ID

Dashboard Page Link

*Parent Folder

Sequence number

Security

*Security Type Public Access

Dashboard Page Administration page

Enter the following dashboard page properties:

Dashboard Name	Enter a unique identifier for the dashboard. This field is required.
Dashboard Label	Enter the label that appears in the navigation. This field is required.
Long Description	Enter a description that appears as the tool tip text.
Owner ID	Select the organization that owns the object.

Dashboard Page Link

Use these fields to specify the location of the dashboard page link in the menu navigation.

Parent Folder	Enter the name of the navigation folder in which you want to place the dashboard page link. Click the look up button to access the Select the Dashboard Link Parent Folder page.
Sequence number	Enter a number to specify the order of the dashboard link within the parent folder. The application sorts the items alphabetically if you omit a sequence number.

Specifying Dashboard Page Security Properties

Use the fields in the Security group box to specify whether access to the dashboard page is restricted by permission list and role or unrestricted and available to all users.

Security

Security Type

Select from these values:

- *Public Access:* Select to allow all users to view and access the dashboard.
- *Select Security Access:* Select to specify PeopleTools Security access, which is based on permission lists and roles. When you select this option, the *Selected Security* grid appears.

Selected Security

Insert rows into the Selected Security grid to specify the permission lists and roles that provide access to the dashboard page. You can combine permission lists and roles by inserting multiple rows in the grid.

Click the *Insert Row* button (+) to add another permission list or role name.

Click the *Delete Row* button (-) to remove the permission list or role name.

Type

For each row, select from these values:

- *Permission List:* Select to secure the dashboard link by permission list.
- *Role:* Select to secure the dashboard link by role.

Name

Specify the permission list or role name that provides access to the dashboard page. Click the look up button to see a list of available names.

Save

Click to save the configuration and return to the Manage Dashboard Pages page.

Cancel

Click to discard any changes and return to the Manage Dashboard Pages page.

See Also

PeopleTools 8.52: Security Administration, "Setting Up Permission Lists"

PeopleTools 8.52: Security Administration, "Setting Up Roles"

Specifying the Location of Dashboard Page Link

Access the Select the Dashboard Link Parent Folder page (Click the look up button for the Parent Folder field).

Select the Dashboard Link Parent Folder

Click the folder name to pick a folder to be used as the parent folder of the dashboard page link. Click the "Cancel" button to go back to the previous page.

Left | Right

 Root

 [Administer Procurement](#)

 [Control Inventory](#)

 [Define Business Rules](#)

 [Maintain Items](#)

 [Manage Production](#)

 [Manage Sales Activities](#)

 [Manage Treasury](#)

 [PeopleTools Quality](#)

 [Plan Production](#)

 [Process Financial Information](#)

 [Schedule Resources](#)

 [Self Service](#)

 [Structure Manufacturing](#)

 [PeopleTools SDK](#)

 [Mobile Demo](#)

 [Manage Assets](#)

 [Worklist](#)

 [Application Diagnostics](#)

 [Tree Manager](#)

 [Reporting Tools](#)

 [PeopleTools](#)

 [My Utilities]

 [Usage Monitoring]

 [Change My Password]

 [My Personalizations]

 [My System Profile]



 [My Dictionary]

 [My Feeds]

Cancel

Select the Dashboard Link Parent Folder page

To select the parent folder of the dashboard page you must navigate the tree until you see the destination folder.

 (collapsed folder)	Click to expand the folder and view the subordinate menu items.
 (expanded folder)	Click to collapse the folder and hide the subordinate menu items.
Employee (folder name)	Click to select the folder as the parent of the dashboard page.
Cancel	Click to return to the previous page.

Specifying the Pagelets That Appear on Dashboard Pages

The page that you use to specify which pagelets can appear on dashboard pages is similar to the page that you use for the same purpose on homepages. The process is also the same.

Access the Tab Content page (Select PeopleTools, Portal, Dashboard, Manage Dashboard Pages. Click the Pagelets button.)

Content

Layout

Tab Content

Label: Role Management

- * Select the pagelets that can be used for this homepage tab definition.
- * 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.
- * Set the pagelet behavior with the drop down list next to the selected pagelet.

<div>▶ QE</div> <div>▶ News</div> <div>▶ Miscellaneous</div>	<div>▶ PeopleSoft Applications</div> <div>▶ Finance</div> <div>▶ Talent Summary</div>
--	---

[Return to Manage Dashboard Pages](#)

Save

Notify

[Content](#) | [Layout](#)

Tab Content page

To select pagelets for that appear on dashboards:

1. Expand any collapsed group boxes and then select the check box for the pagelet that you want to appear on the dashboard page.

Select the Include All? check box to select all pagelets in the category.

Note. The user must have security access to the pagelet component for the pagelet to be properly displayed on the homepage.

2. Use the drop-down list box to select the pagelet behavior. Values are:

<i>Optional</i>	The pagelet does not appear automatically on the homepage. However, it is available for users when they are personalize their dashboard pages.
<i>Opt-Dflt</i> (option-default)	The pagelet appears on all user homepages if the users can access the pagelet. The pagelet can be removed when users personalize the dashboard page.
<i>Req-Fix</i> (required-fixed)	The pagelet appears on all user homepages if users can access the pagelet. The placement of the pagelet cannot be changed, and it also cannot be removed.
<i>Required</i>	The pagelet appears on all user homepages if users can access the pagelet. The placement of the pagelet can be changed, but it cannot be removed.

3. Click the Save button.

Note. Content outside of PeopleSoft applications is available only with PeopleSoft Application Portal.

Important! When you select content for inclusion on any PeopleSoft dashboard page, only one pagelet or portlet should use the calendar prompt. If more than one pagelet or portlet per dashboard page uses the calendar prompt, unexpected results might occur.

See [Chapter 10, "Administering Homepages, Dashboard Pages and Pagelets," Arranging Pagelets on Homepages, page 309.](#)

Arranging the Pagelets on Dashboard Pages

The page that you use to arrange the layout of the pagelets that appear on dashboard pages is similar to the page that you use for the same purpose on homepages. The process is also the same.

Access the Layout page (Select PeopleTools, Portal, Dashboard, Manage Dashboard Pages. Click the Pagelets button.)

Content

Layout


Tab Layout

Label:


Role Management

Basic Layout:

☒



☐



2 columns

3 columns

Click arrows to move pagelets up and down or into neighboring columns. Click "Delete Pagelet" to remove the selected pagelet from the homepage tab definition.

= Required - fixed position pagelet
* = Required - moveable pagelet

Left Column:

--No Pagelets Selected--

Right Column:

--No Pagelets Selected--

Delete Pagelet

[Return to Manage Dashboard Pages](#)

Save

Notify

[Content](#) | [Layout](#)

Tab Layout page

Label	Displays the label information from the Content Ref Administration page.
Basic Layout	<p>Select to display the pagelets in either two or three columns on the dashboard page.</p> <p>To specify the three-column layout, be sure that at least one pagelet does not have the <i>Req-Fix</i> behavior option selected on the Content page.</p>

Copyright © 1988, 2011, Oracle and/or its affiliates. All Rights Reserved.

321

Columns

Displays the pagelets selected on the Content page in the column sections.

If the basic layout is two columns, the pagelets are divided into left and right columns. If the basic layout is three columns, the pagelets are divided into left, center, and right columns.

Note. A pound sign (#) indicates a pagelet with the *Req-Fix* behavior option selected on the Content page. An asterisk (*) indicates a pagelet with the *Required* behavior option selected on the Content page.

For example, #Signon indicates that the Signon pagelet cannot be moved to a different location on the homepage.



Use the Move icons to position the pagelets. Highlight a pagelet, and then click the directional arrow buttons to move the selected pagelet above a pagelet, below a pagelet, to the next column to the right, or the next column to the left.

Delete Pagelet

Highlight a pagelet and then click to delete it from the homepage.

To arrange the dashboard page layout:

1. Select a basic layout option.
2. Select the pagelet name.
3. Click the directional arrow buttons to position the pagelets in the columns.
4. Click the Save button.

Note. As the portal administrator, you determine the amount of flexibility that you give users to change the layout of the pagelets. The settings that you select on both the Tab Content and Tab Layout pages determine the degree of flexibility. Grant the end user as much or as little flexibility as your organization demands.

See [Chapter 10, "Administering Homepages, Dashboard Pages and Pagelets," Selecting the Pagelets That Appear on Homepages, page 308.](#)

Chapter 11

Using Pagelet Wizard to Create and Manage Pagelets

This chapter provides overviews of Pagelet Wizard, and its data sources, display formats, display sizes, and pagelet types. This chapter also discusses how to:

- Define Pagelet Wizard setup values.
- 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.
- Administer Pagelet Wizard pagelets.

Understanding 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 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, Pagelet Wizard presents a series of numbered steps. Each step appears in a numbered path at the top of the page to indicate where you are in the pagelet creation process.

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.

Resource Center

*Data Type

▼ Description

The OBIEE Report Data Type displays a report from an Oracle Business Intelligence Server.

Data Source

Report ID:

Numbered steps on the Select Data Source page

This labeled path assists you in navigating forward and back through the steps in the wizard, which enables you to review or edit pagelet definition values. Once you've completed a step, Pagelet Wizard automatically takes you to the next appropriate step.

Pagelet Wizard leads you through the following steps that are typically used to create a pagelet:

1. Specify pagelet information.
2. Select data source.
3. Specify data source parameters.
4. Select display format.
5. Specify display options.
6. Specify publishing options.

Note. Before you start building a pagelet with Pagelet Wizard, you should familiarize yourself with all data sources. This enables you to evaluate the best way to retrieve the data that you want to display on the pagelet.

See Also

Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Step 2: Selecting a Pagelet Data Source, page 358

Understanding Pagelet Wizard Data Sources

Pagelet Wizard supports the creation of pagelets using the following data sources:

HTML	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.
IB Connector (Integration Broker Connector)	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 be used only when you do not want to use the Integration Broker data type.
Integration Broker	Use to create pagelets that use 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.
Navigation Collections	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.
OBIEE Report (Oracle Business Intelligence Enterprise Edition Report)	Use to create pagelets from existing reports on the OBIEE server.
PeopleSoft Query	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.
Pivot Grid	Use to create pagelets from existing pivot grid instances, which use interactive pivot tables and charts that display application data that you can pivot, slice, and dice across multiple dimensions.
Rowset	<p>Use to access a rowset record for which the data is supplied 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 run 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>

Search record

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.

Use the basic search record data type for simple search lookup pages.

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.

Search record data source pagelets are compatible with the search list display format.

URL

The URL data type retrieves data from external sources such as the internet. This includes sources such as Real Simple Syndication (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 Service Oriented Architecture Protocol (SOAP) data type used in previous releases is now included in the functionality of the IB data type. As of 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 you use Application Portal, Pagelet Wizard includes other data sources that are specific to Application Portal installations.

See Also

[Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Defining Display Formats, page 340](#)

[Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Selecting a Navigation Collection Data Source, page 364](#)

[Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Defining Data Types, page 342](#)

[Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Step 2: Selecting a Pagelet Data Source, page 358](#)

Understanding Pagelet Wizard Display Formats

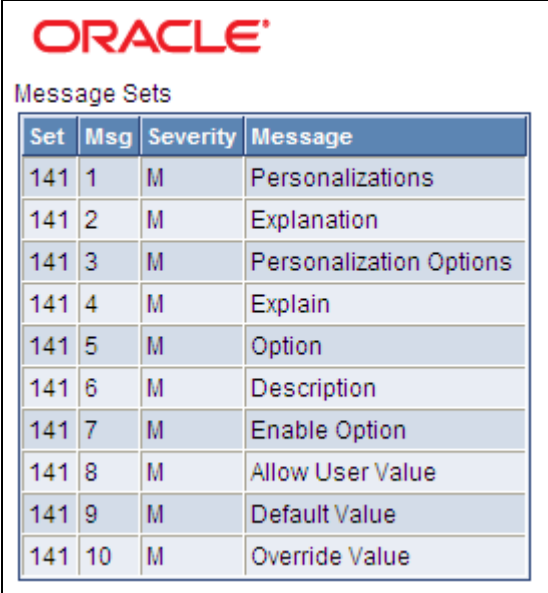
Pagelet Wizard supports the creation of pagelets based on the following display formats:

- 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. The following example shows a table pagelet displaying data from a PeopleSoft Query data source:



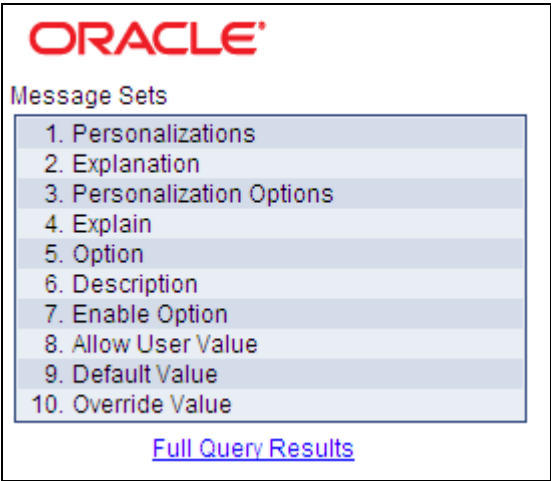
The image shows a table pagelet example. At the top left is the Oracle logo. Below it is the title "Message Sets". The table has four columns: "Set", "Msg", "Severity", and "Message". The data rows are as follows:

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. The following example shows a list pagelet displaying data from a PeopleSoft Query data source:



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. The following example shows a chart pagelet displaying data from a PeopleSoft Query data source:

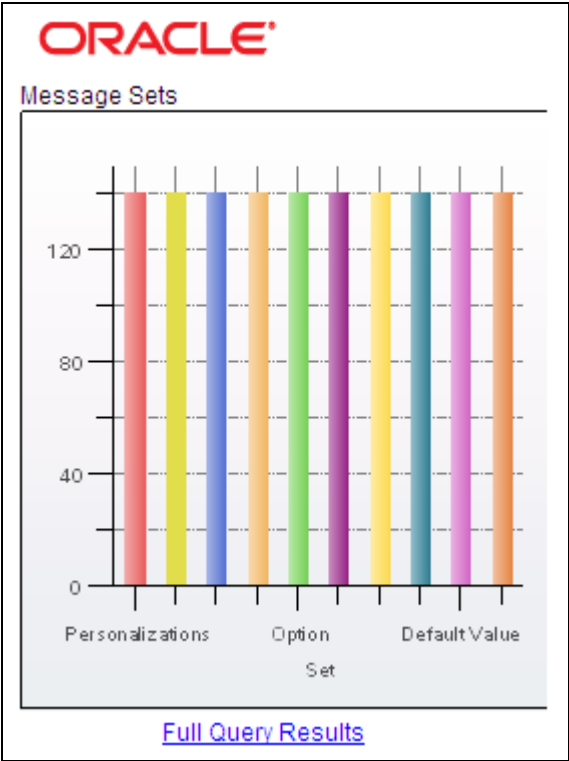


Chart pagelet example

Menu Display Format

The menu display format is appropriate for creating a pagelet that displays data from a navigation collection data source. The following example shows 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. The following examples show search list pagelets displaying data from a search record data source:

Search list pagelet example (Show Results options set to N)

Header ID	Description
PAPPBR_HEADER1_BASIC	Sample header 1
PAPPBR_HEADER1_SIMPLE	Sample header 1 simple
PAPPBR_HEADER2_GBI	Sample GBI header
PAPPBR_HEADER2_GBNOTAB	Sample GBI header (no tabs)
PAPPBR_HEADER3_IEONLY_CURVED	Blue curve header - IE only
PAPPBR_HEADER4_TOOLS	Tools 8.42 delivered header
PAPPBR_HEADER4_TOOLS_NOTAB	Tools 8.42 delivered header
PAPPBR_HEADER5_TOOLSCLASSIC	Tools "Classic" header
PAPPBR_HEADER5_TOOLSNOTAB	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 be used only with data sources that return HTML that can be directly displayed as a pagelet. The following example shows 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 data sources that return XML. Generally, most data types return XML, but circumstances potentially occur in which IBCConnector, URL, and HTML datatypes do 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 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.

The following example shows a custom pagelet displaying data from a PeopleSoft Query data source:



Custom pagelet example

This custom pagelet uses the same PeopleSoft Query data source as the table pagelet example shown previously; however, this example has a custom XSL template applied.

See Also

PeopleTools 8.52: PeopleCode API Reference, "Charting Classes"

Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Defining Pagelet Wizard XSL Prototypes, page 346

Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Step 4: Selecting a Pagelet Display Format, page 393

Understanding 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 so that pagelets are sized appropriately for PeopleSoft portal homepages. 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 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.

Understanding Pagelet Wizard Pagelet Types

Pagelet Wizard supports the publication of the following pagelet types:

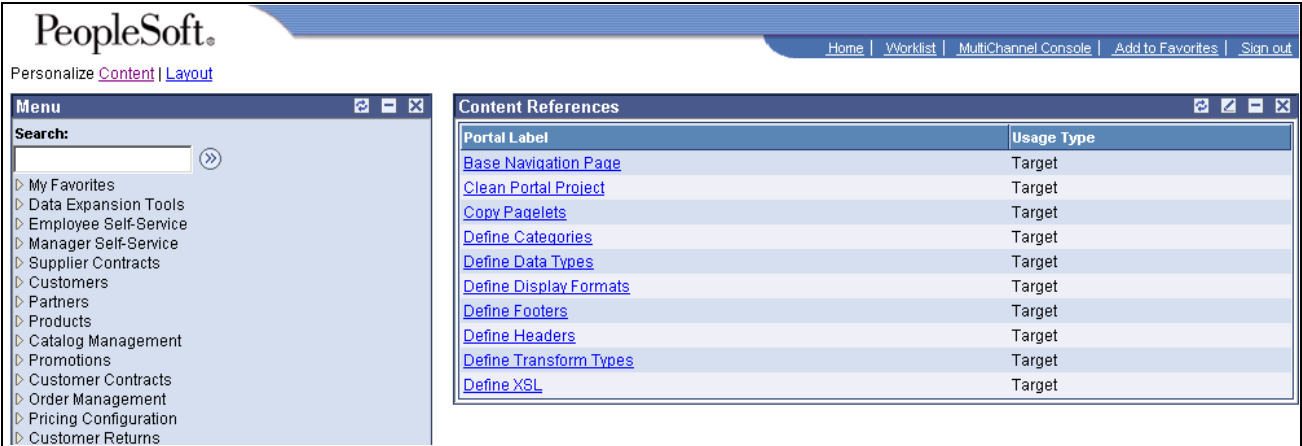
- Homepage pagelets.
- Template pagelets.
- Embeddable pagelets.
- Web Services for Remote Portlets (WSRP) portlets.

Homepage Pagelets

Homepage pagelets are pagelets that are specifically for use on portal homepages. 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 on a homepage uses the calendar prompt, unexpected results may occur.

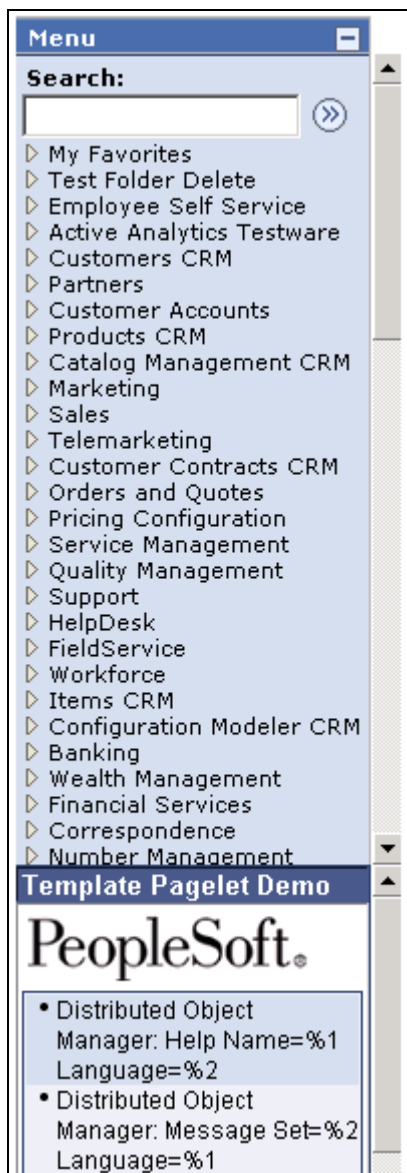
See [Chapter 10, "Administering Homepages, Dashboard Pages and Pagelets,"](#) [Selecting the Pagelets That Appear on Homepages](#), page 308.



Homepage pagelet example

Template Pagelets

Template pagelets are pagelets that can be added to a PeopleSoft portal 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 you use Context Manager to display context-sensitive data in the template pagelet. Pagelets that you intend to use in the task panel of WorkCenter pages can be template pagelets.



Template pagelet example

Embeddable Pagelets

Embeddable pagelets are pagelets that you can embed in PeopleSoft transaction pages or iScripts within an HTML area. For example, you might create an OBIEE-based pagelet that displays a report of sales by region to embed on a Regional Sales page so that you could compare the sales of individual regions or each region to the whole.

WSRP Portlets

A WSRP portlet is a pagelet that can be used with external portals that support the WSRP standard.

See Also

[Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Step 6: Specifying Pagelet Publishing Options, page 419](#)

[Chapter 17, "Creating and Configuring WorkCenter Pages," Configuring WorkCenter Pages, page 636](#)

[Chapter 18, "Understanding Web Services for Remote Portlets \(WSRP\)," page 643](#)

PeopleTools 8.52: PeopleSoft Applications User's Guide

Defining Pagelet Wizard Setup Values

This section discusses how to:

- Set up the PeopleSoft Integration Broker (IB) gateway for Pagelet Wizard.
- Set up the default gateway for use with Pagelet Wizard data sources.
- Set up the Pagelet Wizard IB Connector data source security.
- Set up the Pagelet Wizard Integration Broker data source.

Pages Used to Set Up Pagelet Wizard Setup Values

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
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>PeopleTools 8.52: PeopleSoft Integration Broker Administration</i> , "Managing Integration Gateways."
Define 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.

Page Name	Definition Name	Navigation	Usage
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.
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 for 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 Pagelet Wizard with PeopleSoft Application 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. (Select PeopleTools, Integration Broker, Configuration, Gateways.)

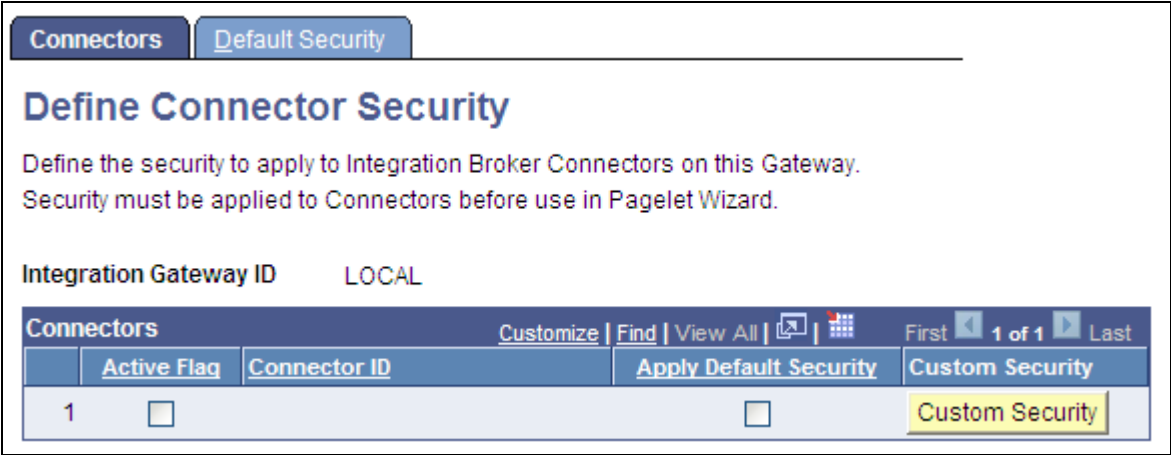
Set up the default gateway according to the instructions in the Integration Broker PeopleBook.

See *PeopleTools 8.52: 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 - Connectors page. (Select PeopleTools, Portal, Pagelet Wizard, Define IB Connector Security, Connectors.)



Define Connector Security - Connectors page

- 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.

See Also

PeopleTools 8.52: PeopleSoft Integration Broker Administration, "Using the Integration Network," Configuring the Integration Network

Setting Up the Pagelet Wizard Integration Broker Data Source

When you use Pagelet Wizard with PeopleSoft Application 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 *PeopleTools 8.52: PeopleSoft Integration Broker Administration*, "Using the Integration Network," Configuring the Integration Network.
2. Set up the appropriate service, service operation, handlers, and routings for the operation that you want to use.

See *PeopleTools 8.52: PeopleSoft Integration Broker Administration*, "Using the Integration Network," Configuring the Integration Network.

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. (Select 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.

The 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 to Create and Manage Pagelets," Understanding Pagelet Wizard Display Formats, page 326](#).

Transformation Type

Enter the internal name that you want to assign to the transformation type.

Description

For reference purposes only, enter a short description of the transformation type.

Long Description

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.

Package Name	Enter the name of the application class package that you want to use for the transformation type.
Path	Enter the names—delimited by colons—of each subpackage in the hierarchy that drills down to the class that contains the method to use for the transformation type.
Application Class ID	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. (Select 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 ☒ 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 to Create and Manage Pagelets," Understanding Pagelet Wizard Display Formats, page 326.](#)

Display Format ID	Enter the name that you want to assign to the display format.
Description	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.
Active	<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 by means of a deactivated display format are unaffected; however, you can no longer use Pagelet Wizard to change the display settings for those pagelets.</p>
Long Description	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.
Sort Order	<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 Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Step 4: Selecting a Pagelet Display Format, page 393.</p>
Image Name	Select the image that you want to appear with the display format name and description on the Select Display Format page.

Transformation Type 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.

See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Defining Transformation Types, page 338.](#)

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 names—delimited by colons—of each subpackage in the hierarchy that drills down to the class that contains the method used to create 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. (Select 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 be used only 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 the analytics server.
- PIVOTGRID data type enables you to display pivot tables and charts in pagelets.
- 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 such as the internet. This includes sources such as RSS newsfeeds.

Warning! Do not alter delivered field values for these data types.

See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Understanding Pagelet Wizard Data Sources, page 325.](#)

Data Type	Enter the internal name that you want to assign to the data type.
Description	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.
Active	<p>Select to 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>
Long Description	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:

Package Name	Enter the name of the application class package that you want to use for the data type.
Path	Enter the names—delimited by colons—of each subpackage in the hierarchy that drills down to the class that contains the method that defines the data type.
Application Class ID	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 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 default search application class at the data type level 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 are included in the subsequent list.

- Changing the code in the search class.
- Adding a new search class.
- Removing a search class.
- Replacing a search class.

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.

Note. Pagelet search is supported for homepage and embedded pagelets only.

This table describes the Search Application Class fields:

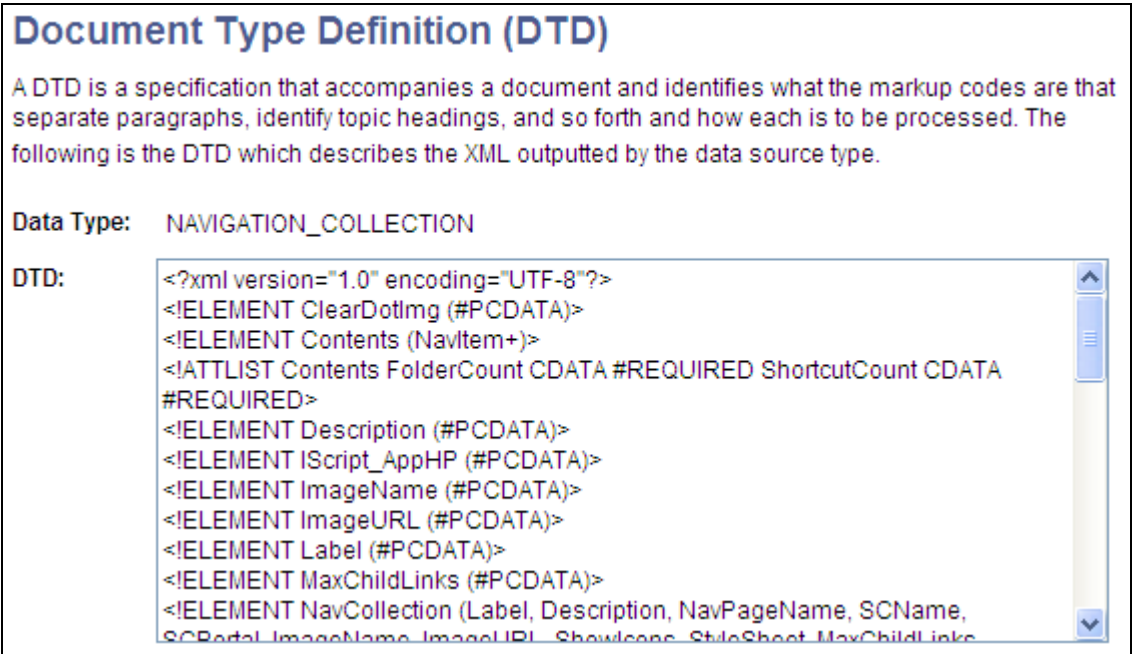
Package Name	Enter the Search application class package name that you want to use for the data type.
Path	Enter the names—delimited by colons—of each subpackage in the hierarchy that drills down to the class that contains the method to define the search.
Application Class ID	Enter the name of the Search 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

Display Format ID	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. <u>See Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Defining Display Formats, page 340.</u>
View Document Type Definition (DTD)	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 that are used to describe content in the XML from the data source type. This information is useful when you use third-party tools to generate XSL templates for the datatype.



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" ?>

<!--
    Generated XSLT 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>XSL Prototype Name</i>	<i>Data Type</i>	<i>Description</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.

XSL Prototype ID	Enter an ID for the XSL prototype.
Description	<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 Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Specifying Custom Display Options, page 414.</p>
Active	<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 by means of the deactivated XSL prototype are unaffected.</p>
Long Description	For reference purposes only, enter a long description of the XSL prototype.
Data Type	Select the applicable data type for the XSL prototype. You can assign only one data type per XSL prototype.
Define XSL	Enter the XSL code for formatting pagelet XML data in the desired pagelet format.
Delete XSL	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. (Select 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	<p>Select to activate the pagelet header and make it a selectable field value in the Header field on the Specify Display Options page.</p> <p>If you deactivate a header that is in use, the header no longer appears on associated pagelets. In addition, the header is not available for selection on the Specify Display Options page.</p>
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. (Select PeopleTools, Portal, Pagelet Wizard, Define Headers, Define Header, Specify Image.)

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. (Select 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 value in the Footer field when users assign footers to pagelets on the Specify Display Options page.

352

Copyright © 1988, 2011, Oracle and/or its affiliates. All Rights Reserved.

Active	Select to activate the pagelet footer and make it a selectable value in the Footer field on the Specify Display Options page. If you deactivate a footer that is in use, the footer no longer appears on associated pagelets. In addition, the footer is not available for selection on the Specify Display Options page.
Long Description	For reference purposes only, enter a long description of the pagelet footer.
HTML Area	Enter the 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.
Delete Footer	Click to delete the pagelet footer design. You will be prompted to confirm your deletion.

Defining Footer Images

Access the Footer Image page. (Select PeopleTools, Portal, Pagelet Wizard, Define Footers, Specify Image..

Define FooterSpecify ImagePreview

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	<p>Select the image source. Options are:</p> <p><i>System Catalog:</i> Select to specify an image from the PeopleTools system image catalog.</p> <p><i>External Source:</i> Select to specify an image from a source external to the PeopleTools system image catalog.</p>
Image Name	If you select <i>System Catalog</i> as the image source, click the Search button to access a list of existing images.
Image URL	If you select <i>External Source</i> 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	<p>Select the alignment of the image. Options are:</p> <p><i>Left:</i> Select to align the image with the left margin of the pagelet.</p> <p><i>Right:</i> Select to align the image with the right margin of the pagelet.</p>
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 by means of Pagelet Wizard.

Defining Pagelet Categories

Access the Define Categories page. (Select 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	<p>Click to delete a pagelet category. You are prompted to confirm the deletion.</p> <p>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.</p>
------------------------	---

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 to Create and Manage Pagelets," [Understanding Pagelet Wizard Data Sources](#), page 325

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. (Select PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard.)

Pagelet Wizard
Step 1 of 6

1 2 3 4 5 6

Next >

Specify Pagelet Information

The following information will be used to identify and categorize your pagelet.

Pagelet Information	
Pagelet ID:	PT_1
*Pagelet Title:	New Construction Update
Description:	
Owner ID:	PeopleTools
Category ID:	Portal Administration
Help URL:	http://help.oracle.com/construction.html

Specify Pagelet Information page

Pagelet ID

Enter an ID for the pagelet. Each pagelet that you build with Pagelet Wizard must have a unique ID.

Note. Pagelet IDs can contain only 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 you remove illegal characters from the pagelet ID the name is found to match an existing pagelet ID, then Pagelet Wizard appends numbers to the end of the pagelet ID to make the new pagelet ID unique.

Pagelet Title

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.

Description

Enter additional descriptive information to identify the pagelet. This reference text is used in Verity and search record searches.

Owner ID

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.

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.

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 to Create and Manage Pagelets," Defining Pagelet Wizard Pagelet Categories, page 354.](#)

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 does 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 pivot grid 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 to Create and Manage Pagelets," Understanding Pagelet Wizard Data Sources, page 325](#)

[Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Defining Data Types, page 342](#)

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. (Select PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 2.)

[illegible]

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 also, no end-user personalization exists for HTML data source pagelets.

Note. Source code mode is disabled in the Rich Text Editor on the Step 2 page of Pagelet Wizard for HTML type of pagelets. This is for security reasons because malicious code could be inserted in Source code mode. If you want to enable the Source mode for some special reason, 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 to Create and Manage Pagelets," Defining Data Types, page 342.

Selecting an IB Connector Data Source

Access the Select Data Source page. (Select 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

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:
Connector ID:

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. (Select 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** Integration Broker

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: PT_CDB_UPDATE 🔍

Receiver Node Name: PT_CDB_UPDATE 🔍

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. (Select PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 2.)

Select *Navigation Collection* in the Data Type field.

Pagelet Wizard

Step 2 of 6

123456

< 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 already have been created using the navigation collections pages. 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 you can create a navigation collection pagelet directly from the navigation collections component, pagelets created using this component 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 <i>Navigation Collection</i> .
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. (Select 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 13, "Configuring OBIEE Report Pagelets," Loading OBIEE Report Definitions, page 518.](#)

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. (Select 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

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: PS Query

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: MESSAGE_SETS

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.

Note 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 *PeopleTools 8.52: 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. This field 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 Pivot Grid Data Source

Access the Select Data Source page. (Select PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 2.)

Select *Pivot Grid* 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.

Sales by Region

*Data Type

Description

Pivot Grid allows users to create table and/or chart representation of the data from various sources. Pivot Grid Wizard is the utility which allows users to create, customize, and secure these pivot grid definitions.

Data Source

Pivot Grid Name:

Select Data Source page (Pivot Grid data source)

Pivot Grid is a PeopleSoft component that enables you to display application data in interactive pivot tables and charts so that you can pivot, slice, and dice your application data across multiple dimensions. Pivot grids are ideal data sources for pagelets that you use on dashboards and homepages as a form of operational reporting.

Data Source

Pivot Grid Name Select the pivot grid that you want the pagelet to display.

See Also

PeopleTools 8.52: PeopleSoft Pivot Grid, "Understanding PeopleSoft Pivot Grid"

Selecting a Rowset Data Source

Access the Select Data Source page. (Select 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

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

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 also, no end-user personalization for rowset data source pagelets.

When creating a rowset data source pagelet, you skip step 3 and proceed 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 that 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. (Select PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 2.)

Select *Search Record* 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.

Facilities Page

***Data Type** Search Record ▼

▼ Description

The Search Record Data Source displays the search box or the search results from a given component.

Data Source

***Portal Name:** CUSTOMER 🔍

***Content Reference:** PT_MESSAGE_CATALOG1_GBL 🔍

***Access Type:** Basic Search Record ▼

▼ 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 *PeopleTools 8.52: 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. (Select 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

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

The URL DataType retrieves data from external sources specified by a URL.

Data Source

URL:

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 system 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 pivot grid data source parameters.
- Specify query data source parameters.
- Specify rowset data source parameters.
- Specify search record data source parameters.
- Specify URL data source parameters.
- Specify search record 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, this field is populated with that default value.

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 to Create and Manage Pagelets," Understanding System Variables Supported as Data Source Parameters, page 373](#).

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 to Create and Manage Pagelets," Step 3: Specifying Pagelet Data Source Parameters, page 372](#).

Pages 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 that are 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 you build 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>Note. This step is not applicable when you create 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>Note. 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 proceed 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 Parameters page. (Select 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.

Facilities

Data Source Parameters					
Field Name	Description	*Usage Type	Required	Default Value	
CLIENTCERT	CLIENTCERT	Not Used	<input type="checkbox"/>		Values
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 *PeopleTools 8.52: PeopleSoft Integration Broker*, "Understanding PeopleSoft Integration Broker."

See Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Specifying User-Defined Data Source Parameters, page 392.

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 Parameters page. (Select PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 3.).

Pagelet Wizard

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.
Facilities

Data Source Parameters					
Field Name	Description	*Usage Type	Required	Default Value	
MCFEM_REQ.MCF_USER		System Variable	<input checked="" type="checkbox"/>		
MCFEM_REQ.MCF_UID_LIST		User Specified	<input type="checkbox"/>		Values
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_ATTRROOT		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 . 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 Parameters page. (Select 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
.MAXCHILDLINKS	Maximum Child I	Fixed	<input checked="" type="checkbox"/>	10

[Values](#)

[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.

Field Name

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 is 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 *PeopleTools 8.52: PeopleSoft Integration Broker*, "Understanding PeopleSoft Integration Broker."

Personalization Instructions

Use the Personalization Instructions text box to enter instructional informs 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 Parameters page. (Select 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				
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 Parameters 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.

Field Name	Displays the data source parameters that correspond to the OBIEE prompts associated with the selected OBIEE report. This field 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.
Description	This field contains a description that identifies the purpose or location of the OBIEE field.
Required	<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 is display-only, the value is set in the application class at the application programming interface (API) level.</p> <p>If you leave this option deselected, the data source parameter value can be cleared at runtime, or it can be entered to refine the data selection criteria.</p>
Reset to Default	Click to reset the values on this page to the default values specified by the OBIEE data source API.

Specifying Pivot Grid Data Source Parameters

After selecting *Pivot Grid* as the data type on the Select Data Source page, access the Specify Data Source Parameters page. (Select 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.

Sales by Region

Data Source Parameter Details

[Configure pivot grid views](#)

Data Source Parameters
Find | First 1-3 of 3 Last

Field Name	Description	*Usage Type	Required	Default Value
PVG_VIEWNAME	<input type="text" value="View Name"/>	Fixed	<input checked="" type="checkbox"/>	PVGTEST.View
.REPORTWIDTH	<input type="text" value="Report Width"/>	Fixed	<input checked="" type="checkbox"/>	<input type="text" value="470"/>
.REPORTHEIGHT	<input type="text" value="Report Height"/>	Fixed	<input checked="" type="checkbox"/>	<input type="text" value="550"/>

Reset to Default

Specify Data Source Parameters page (Pivot Grid data source)

Data Source Parameter Details

Displays the Configure pivot grid views link, which you click to open the Pivot Grid Views Component page where you can define the pivot grid options.

See *PeopleTools 8.52: PeopleSoft Pivot Grid*, "Creating and Viewing a Pivot Grid Pagelet Using the Pagelet Wizard," Defining the Pivot Grid Display Options.

Data Source Parameters

Pagelet Wizard analyzes the selected pivot grid and generates the list of pagelet parameters shown in this step. You cannot add to the list of parameters.

Description Enter text that describes the field.

Usage Type	Select the way the system uses the parameter. Select one of these options: <ul style="list-style-type: none">• <i>Admin Specified</i>• <i>Context Sensitive</i>• <i>Fixed</i>• <i>Not Used</i>• <i>System Variable</i>• <i>User Specified</i>
Required	Select to require a value for the parameter.
Default Value	Enter the parameter value that the system uses by default.
Reset to Default	Click to set all values on this page to the default values specified by the pivot grid data source API.

Pivot Grid Views Component

Access the Pivot Grid Component (Click the Configure pivot grid views link on the Specify Data Source Parameters page).

Pivot Grid NamePVGTEST

View NamePVGTEST.View

View Description

View Options

Display Options

Grid Only

Chart Only

Grid and Chart

Specify Axis Information

PersonalizeFind

First1-6 of 6Last

	Data Source Columns	Field Format	Grid Axis	Chart Axis
1	Month	Char30	Filter	Filter
2	Region	Char30	Row	X-Axis
3	Product	Char30	Row	
4	Unit Cost	Num8.2	Column	Y-Axis
5	Sales	Num8.2	Column	
6	Prd Sales	Num8.2	Column	

Grid Options

☒ Collapsible Data Area

☐ No Drag and Drop

☒ Expanded State

Chart Options

Chart TitleSales Information

Chart Type2D Bar Chart

X-Axis LabelRegion

X-Axis Label Angle

Y-Axis LabelUnit Cost

Y-Axis Label Angle

Advanced Options

3D Rotation Angle

Chart Height

Chart Width

Pivot Grid Views Component page

- View Name

Select the pivot grid view to use for this pagelet.
- View Description

Enter text that describes the pivot grid view.

Display Options

Select which elements appear on the pagelet.

Choose one of these values:

- *Grid Only*
- *Chart Only*
- *Grid and Chart*

Grid Options

- Collapsible Data Area- select to enable the user to collapse the pivot grid into the grid header.

If you select this option, the system automatically selects the Expanded State check box.

- Expanded State - select to display the grid in the expanded state when the pagelet initially appears on the homepage or dashboard.
- No Drag and Drop - select to disable the drag and drop functionality of elements within the pivot grid.

See *PeopleTools 8.52: PeopleSoft Pivot Grid*, "Using Pivot Grid Wizard," Grid Options.

Chart Options

Enter or select values for the title, type, x and y-axis labels and x and y-axis angles (in degrees) of the chart.

See *PeopleTools 8.52: PeopleSoft Pivot Grid*, "Using Pivot Grid Wizard," Chart Options.

Advanced Options

Enter values for the three-dimensional rotation angle (in degrees), height (in pixels), and width (in pixels) of the chart.

See *PeopleTools 8.52: PeopleSoft Pivot Grid*, "Using Pivot Grid Wizard," Chart Options.

Save As

Click to open the Save View As dialog box, where you can either save the current pivot grid model as a new view or update an existing view.

See *PeopleTools 8.52: PeopleSoft Pivot Grid*, "Using Pivot Grid Viewer," Viewing Pivot Grid Options.

See Also

PeopleTools 8.52: PeopleSoft Pivot Grid, "Using Pivot Grid Viewer"

Specifying Query Data Source Parameters

After selecting *PS Query* as the data type on the Select Data Source page, access the Specify Data Source Parameters page. (Select 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

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.

You can 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 that you have created. The five vendor ID prompts will be shown as data source parameters. However, they will be marked as not required. End users will then be able 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 can supply up to five vendor IDs.

Field Name	Displays the data source parameters that correspond to the query prompts associated with the selected query. This field 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.
-------------------	---

Required

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.

If this option is selected and display-only, the value is set in the application class at the application programming interface (API) level.

If you leave this option deselected, the data source parameter value can be cleared at runtime, or it can be entered to refine the data selection criteria.

Prompts tied to the IN clause of the query will not be marked as required.

Reset to Default

Click to reset the values on this page to the default values specified by the query data source API.

Values

Click the [Values](#) 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 proceed 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 Parameters page. (Select 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

Data Source Parameters					
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	Values

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.

Field Name

Displays the data source parameters that correspond to the search keys required for the search record. This field also displays the required system-supplied .MAXROWS and SHOWRESULTS search record data source parameters.

Use the .MAXROWS field to limit the amount of data returned to the pagelet.

Use the SHOWRESULTS field to determine whether search results appear on the pagelet.

If this field value is set to *Y*, search record results appear directly on the pagelet with links to transaction data.

In addition, if this field value is set to *Y*, 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.

If this field value is set to *N*, the pagelet displays a search prompt for the selected search record. The Search By drop-down list box values that are available on the pagelet are the search keys for the search record.

If data source parameter default key field values have been defined, then when the pagelet is initially accessed, the default value appears in the begins with pagelet field.

Required

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.

If this option is selected and display-only, the value is set in the application class at the API level.

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.

If you leave this option deselected, the data source parameter value can be omitted at runtime, or the value can be provided to refine the data selection criteria.

Reset to Default

Click to reset the values on this page to the default values specified by the search record data source API.

Values

Click the Values 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 URL Data Source Parameters

After selecting *URL* as the data type on the Select Data Source page, access the Specify Data Source Parameters page. (Select 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.
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 pass only 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 because 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

Page Name	Definition Name	Navigation	Usage
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 the format for the pagelet.

Selecting a Display Format

Access the Select Display Format page. (Select PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 4.)

Pagelet Wizard **Step 4 of 6**





1 2 3 **4** 5 6

< Previous Next >

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"/>	 Table	Display your pagelet data in tabular format, with customizable columns, visual display and ordering		
<input type="radio"/>	 List	Display your pagelet data as a numbered or bulleted list		
<input type="radio"/>	 Chart	Display your pagelet data as line, bar, pie or histogram chart, complete with customizable display options and drilldown capabilities		
<input type="radio"/>	 Custom	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 that are available for selection on this page depend on the data source that 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, OBIEE, and pivot grid data source pagelets.
Custom	Available for HTML, navigation collection, query, rowset, and search record data source pagelets.

See Also

Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Step 5: Specifying Pagelet Display Options, page 395

Step 5: Specifying Pagelet Display Options

This section lists common elements and discusses how to specify 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.

Common Elements Used in This Section

This section discusses common elements used in this section.

Common Links

This table describes the common links used in this section.



Click the Link Options icon to display the Link Details page, where you can associate URLs with field values, which at runtime appear on the pagelet as active links that you click to access additional relevant information.

See Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Specifying Data Link Details, page 416.

Additional Text

The Additional Text group box contains options to add headers, footers, opening text, and closing text to a pagelet.

Header	<p>Select a pagelet header. You can select from headers that are defined on the Define Header page.</p> <p>See Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Defining Headers, page 349.</p>
Opening Text	<p>Select the type of opening text that you want to appear on the pagelet. Options are:</p> <p><i>Freeform Text/HTML:</i> The Text field appears. Enter the opening text that appears on the pagelet.</p> <p><i>Message Catalog Entry:</i> 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.</p> <p>See <i>PeopleTools 8.52: System and Server Administration</i>, "Using PeopleTools Utilities," Message Catalog.</p>
Show "View Source Data" Link	<p>Displays when <i>PS Query</i> 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.</p>
Closing Text	<p>Select the type of closing text to appear on the pagelet. Options are:</p> <p><i>Freeform Text/HTML:</i> The Text field appears. Enter the closing text that appears on the pagelet.</p> <p><i>Message Catalog Entry:</i> 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.</p> <p>See <i>PeopleTools 8.52: System and Server Administration</i>, "Using PeopleTools Utilities," Message Catalog.</p>
Footer	<p>Select a pagelet footer. You can select from footers that are defined on the Define Footer page.</p> <p>See Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Defining Footers, page 352.</p>

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 to Create and Manage Pagelets," Search Application Class, page 345.](#)

Search Box

Select a value from these options to determine the location of the search box field:

- *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 multiple PSSEARCHBOX nodes are 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 names—delimited by colons—of each subpackage in the hierarchy that drills down to the class that contains the method used 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 that you have defined so far. When you click Customize, you switch from a noncustom display format selected on the Select Display Format page to a custom format of your definition.

The XSL field is populated based on preexisting display options that are set for the original noncustom display format. You can edit the XSL directly in the XSL field and retain as much of the original noncustom display options as you want. Remember that if you apply an XSL template using the XSL Template field, the XSL is regenerated based on the selected template, and the original noncustom display options that 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 noncustom display format.


When you click the Customize button, the current pagelet definition is saved for restoration purposes. You can revert to this precustomization 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 access the version of the page for the Custom display format.

See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Specifying Custom Display Options, page 414.](#)

Pages Used to Specify Pagelet Display Options

Page Name	Definition Name	Navigation	Usage
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 step 5.	Enter pagelet table formatting details and header and footer options. Preview the pagelet.
Set Thresholds	PTPPB_WIZ_TRSHVALS	Click 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.

Page Name	Definition Name	Navigation	Usage
Specify Display Options (search list format)	PTPPB_WIZ_DISP_SRC	PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard Select <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 formatting details for the pagelet search list, 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 associate URLs with pagelet field values so that at runtime data appears on the pagelet as active links that you click to access additional relevant information.

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

1
2
3
4
5
6

< Previous
Next >

Specify Display Options

Specify the visual options related to the display format for your pagelet.

New Construction Update

Table Options

☒ Display Gridlines

☒ Display Alternate-Row Shading

☐ Display Row Numbers

Column Name	*Alignment			
MESSAGE_SET_NE	Defau	⛔	+	-
DESCRSHORT	Defau	⛔	+	-
DESCR	Defau	⛔	+	-

Set Thresholds

Additional Text

Header

Opening Text

☐ Show "View Source Data" Link

Closing Text

Footer

Search Options

Search is supported for homepage pagelets and embeddable pagelets only.

*Search Box

▶ Custom Search Class

Customization

Customize

Pagelet Preview

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

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*.



Click the Link Options icon to display the Link Details page, where you can associate URLs with field values, which at runtime appear on the pagelet as active links that you click to access additional relevant information.

See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Specifying Data Link Details](#), page 416.

Setting Thresholds

Access the Set Thresholds page by clicking the Set Thresholds button in the Table Options of step 5 in Pagelet Wizard.

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 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. (Select PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 5.)

Note. For OBIEE Reports and pivot grids, *Passthru* is the only display format option.

Pagelet Wizard
Step 5 of 6

123456

< Previous
Next >

Specify Display Options

Specify the visual options related to the display format for your pagelet.

Market by Region

Additional Text

Header ▼

Opening Text ▼

Freeform Text/HTML

Text ▼

Text entered here appears directly ABOVE pagelet details.

Closing Text ▼

Message Catalog Entry

Message Set ▼

Number ▼

1

Footer ▼

Search Options

Search is supported for homepage pagelets and embeddable pagelets only.

*Search ▼

No Search Box

▶ Custom Search Class

Pagelet Preview

Text entered here appears directly ABOVE pagelet details.

Market Detail

Region	Market
CENTRAL REGION	CHICAGO
	CINCINNATI
	COLUMBUS
	DES MOINES
	DETROIT
	GRAND RAPIDS
	INDIANAPOLIS
	KANSAS CITY
	LOUISVILLE
	MILWAUKEE
	MINNEAPOLIS
	OKLAHOMA CITY/TULSA
	OMAHA
ST. LOUIS	

This closing text is derived from from Message Set 21000, Message Number 1.

Specify Display Options page (passthru display format)

Additional Text

Header

Select a value from this field to display a header below the pagelet header, but above any opening text.

Opening Text

Select a value to include text *above* the pagelet details.

Select from these values:

- *Freeform Text/HTML*

Select to enable the Text field and manually enter static text or HTML.

- *Message Catalog*

Select to enter the message set and message number from which to derive the opening text from the message catalog.

Text

Enter text or HTML. This field appears when you select *Freeform Text/HTML* as the opening or closing text.

Closing Text

Select a value to include text *below* the pagelet details.

Select from these values:

- *Freeform Text/HTML*

Select to enable the Text field and manually enter static text or HTML.

- *Message Catalog*

Select to enter the message set and message number from which to derive the opening text from the message catalog.

Footer

Select a value from this field to display a footer above the Footer bar, but below any 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. (Select 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.

Search Pagelet

Additional_text

Header PeopleSoft Logo

Opening Text

Closing Text

Footer Powered by PeopleSoft

Customization

Customize

Pagelet Preview

PeopleSoft®

Access Group

APPLICATION ENGINE

APPLICATION MSG

BUSINESS COMPONENTS

BUSINESS INTERLINK

BUSPROC DEFINITION

CHANGE CONTROL

COBOL DEFINITION

CONTENT DEFINITION

EDI MANAGER

EO BAM FRAMEWORK

powered by PeopleSoft

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 (Select 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.

NC Pagelet

Menu Options

☒ **Top Level Title**

☐ **Top Level Description**

***Maximum Columns** 1

***Display Order** Left/Right

☐ **Alternate Row Shading**

***Parent Images** Large Images

☒ **Parent Descriptions**

***Child Display Type** List with Images

Additional_text

Header

PeopleSoft Logo

Opening Text

Closing Text

Footer

Customization

Customize

Pagelet Preview

PeopleSoft®

Tax Center

Tax Setup
 Set up tax structures

- [Sales and Use Tax](#)
- [Value Added Tax](#)
- [Intrastat](#)
- [Excise and Sales Tax IND](#)

Related Setup
 Set up tax-related structures in applications

- [Asset Management](#)
- [Common](#)
- [Expenses](#)
- [General Ledger](#)
- [Procurement](#)
- [Sales](#)
- [Treasury](#)

Tax Processing
 Perform tax processing and reporting

- [Sales and Use Tax](#)
- [Value Added Tax](#)
- [Intrastat](#)
- [Excise and Sales Tax IND](#)

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 to Create and Manage Pagelets," Understanding Pagelet Wizard Display Sizes, page 332.

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 that 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. (Select 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 bullet 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 for which 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. (Select 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.

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 Category	Usage Value
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)

The screenshot shows a web-based configuration interface. At the top is an 'Overlay Field' section with a dropdown menu and '+' and '-' buttons. Below this is the 'Additional Text' section, which contains several labeled input fields: 'Header' with the value 'PeopleSoft Logo', 'Opening Text' with 'Freeform Text/HTML', 'Text' with 'Popular pagelet usage.', a checked checkbox for 'Show "View Source Data" Link', 'Closing Text', and 'Footer' with 'Powered by PeopleSoft'. At the bottom is the 'Customization' section, which features a prominent yellow button labeled '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 and y axes. 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 might have a y axis that displays the number of times that a pagelet has been accessed. This access count could be displayed against an x axis that shows the name of the pagelet accessed.

Chart Options

Chart Type	Select the type of chart to use to display the query results on the pagelet. Available chart formats are those supported by the PeopleCode Chart class. <i>See PeopleTools 8.52: PeopleCode API Reference, "Charting Classes."</i>
X-Axis Field	Select the field to use to derive x -axis values for the chart. Available values are determined by the output fields of the data source.
X-Axis Label	Select the field that you want to use to derive the labels for the x -axis values for the chart. Available values are determined by the output fields of the data source.
X Label Angle	Enter the angle at which you want the text of the x -axis labels to appear along the x 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 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 here:

Quarter	Revenue	Profit
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:

Quarter	Type	Value
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 output fields of the data source.

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 <i>Rows</i> . Select the field to use 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 for the chart legend to appear. Options are: <ul style="list-style-type: none"> • <i>Bottom</i> • <i>Left</i> • <i>None</i> • <i>Right</i> • <i>Top</i>
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). <u>See Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Understanding Pagelet Wizard Display Sizes, page 332.</u>
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 chart 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. (Select PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard. Select button 4.)

Pagelet Wizard

Step 5 of 6

123456

< PreviousNext >

Specify Display Options

Specify the visual options related to the display format for your pagelet.

Custom Pagelet

Custom Options

XSL Template

Table

Generate

XML

<?xml version="1.0" standalone="yes"?>
<queryresult><queryproperties><querylink
><URL>http://adas0112/psp/pa890dvl/EMP
LOYEE/EMPL/inf?

XSL

<?xml version="1.0" ?>

<!--
Description: XSL T supplied with the

Additional_text

Header

PeopleSoft Logo

Opening Text

☒ Show "View Source Data" Link

Closing Text

Footer

Powered by PeopleSoft

Pagelet Preview

PeopleSoft®

Set Msg Severity Message

| | | | |
|---|---|---|--|
| 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 |

[Click here for more...](#)

[Full Query Results](#)

powered by

PeopleSoft

Specify Display Options page (custom display format)

Custom Options

XSL Template

Select the XSL template to use 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 to Create and Manage Pagelets," Defining Pagelet Wizard XSL Prototypes, page 346.](#)

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 on the Specify Display Options page.)

Pagelet Wizard
Link Details

Specify the details of the Link.

Link Type: ☐ Content Reference ☒ External URL

Base URL: Fixed Text Entry [dropdown] [text field]

Link Paths First 1 of 1 Last

| *Path Source | Parameter Value |
|-----------------------------|-----------------|
| Fixed Text Entry [dropdown] | [text field] |

Link Parameters First 1 of 1 Last

| *Name Source | Parameter Name | Parameter Value |
|-----------------------------|----------------|-----------------|
| Fixed Text Entry [dropdown] | [text field] | [text field] |

Link Suffix: Fixed Text Entry [dropdown] [text field]

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 appear on the Specify Display Options page if you select the table, rowset, list, and chart formats. Entering information on the Link Details page enables you to associate URLs with field values, which at runtime appear on the pagelet as active links that you click to access additional relevant information.

Pagelet Wizard builds the column data link in the follow way 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 hard-code the link information by entering a value.

Link Paths**Path Source**

Select the method to use 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 hard-code 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 to use 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 to use 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 hard-code 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 selected *Fixed Text Entry* as the name source, enter a parameter value.

Link Suffix

Select the method to use 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 hard-code the link information by entering a value.

If you selected *Specified by Data* as the link suffix, select a data field.

If you 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 you select 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 appear in the pagelet preview on the right side.

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.

URL Test

Custom Options

XSL Template RSS 0.91-0.93

Generate

XML
 <?xml version="1.0" ?><DOCTYPE rss
 PUBLIC "-//Netscape
 Communications//DTD RSS
 0.91//EN" "http://my.netscape.com/publishif

XSL
 <?xml version="1.0"?>
 <!-- RSS 0.91, 0.92, 0.93 XSLT -->
 <xsl:stylesheet version="1.0"
 xmlns:xsl="http://www.w3.org/1999/XSL/Tra

Additional_text

Header

Opening Text

Closing Text

Footer

Pagelet Preview

Wired News

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.

Careful Where You Put That Tree

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.

All the World's a Podcast

It's incredibly easy to create and syndicate your own audio content. Here's how. By Michael Calore.

Step 5 of Pagelet Wizard with formatted newsfeed

Step 6: Specifying Pagelet Publishing Options

This section provides an overview of caching options and discusses how to specify pagelet publication options. This is step 6 in the Pagelet Wizard pagelet creation process.

Understanding Caching Options

Enabling caching improves performance because after the pagelet is stored in the cache memory, the next time the pagelet is accessed, the information is retrieved and displayed from cache memory instead of requiring a trip to the server. Three caching options exist:

- Private.

Private caching is user-based; pagelet caching is stored per user. For example, if user A accesses the pagelet, only user A will access the pagelet from the cache.

- **Public.**

Public caching is application-based; pagelet caching is stored per web server instance. For example, 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.**

Role caching is role-based; pagelet caching is stored per *user role combination*. For example, if user A has membership in roles 1, 2, and 3, only another user with all roles 1, 2, and 3 will access the pagelet from the cache.

Content Reference Attributes

If you enable caching, Pagelet Wizard adds the following attribute information to the pagelet content reference:

- Name: *PSCACHECONTROL*
- Label: *PSCache-Control*
- Attribute value: *Caching Type field value;max-age=Minutes to Cache field value*

For example, *Public;max-age=30*.

After the pagelet is saved and registered, you can use Pagelet Wizard to alter caching options in the same way that you can alter other registration information. However, altering caching option values requires that you reregister the pagelet.

See [Chapter 5, "Administering Portals," Administering Content References, page 143](#).

Common Elements Used in This Section

Enable Caching	Select to enable caching for the homepage or template pagelet. Selecting this option displays the Caching Options group box.
Caching Type	<i>Private:</i> Select to enable user-based caching. <i>Public:</i> Select to enable application-based caching. <i>Role:</i> Select to enable role-based caching.
Minutes to Cache	Enter the number of minutes that the pagelet remains available in the cache. If you do not enter a value, a default value of 30 minutes is used.

WSRP Portlet

Select this option if you intend this pagelet to be produced by PeopleTools as a 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 enable 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 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 sign-in 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 the portlet is 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 causes the pagelet to be registered for WSRP production.

See [Chapter 18, "Understanding Web Services for Remote Portlets \(WSRP\)," page 643](#) and [Chapter 20, "Using WSRP to Consume and Produce Remote Portlets," page 669](#).

Pages Used to Specify Pagelet Publishing 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.

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
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 Pagelet Publication Options

Access the Specify Publishing Options page. (Select PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard; then click the icon for step 6.)

Pagelet Wizard
Step 6 of 6

1
2
3
4
5
6

< 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
☐ Hide Pagelet Border
☐ Requires Personalization

☐ 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

☐
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](#) | [First](#) | [1 of 1](#) | [Last](#)

*Type	Name	Description
<input type="text" value="Permission List"/>	<input type="text" value="PTPT1100"/>	Security Administrator

Specify Publishing Options page (2 of 2)

When you access a pagelet definition on this page, the pagelet registration options (Homepage Pagelet, Template Pagelet, Embeddable Pagelet, or WSRP Portlet) are reflected only as they exist in the portal that you are currently signed in to. Select the registration option to publish the pagelet to the portal where you are currently signed in.

For example, if you access ABC123 pagelet, the Homepage Pagelet option is selected only if the portal where you are currently signed in contains a content reference for the ABC123 pagelet. If no content reference for this pagelet is in the current portal, even if a content reference for this pagelet is in another portal on the same database, the Homepage Pagelet option will be deselected. Select the Homepage Pagelet option to publish the pagelet to the current portal.

Homepage Pagelet

Select to publish the pagelet as a homepage pagelet that can be added to a WorkCenter page, homepage or dashboard page.

Folder

Select the portal registry folder in which you want to create the pagelet content reference. The pagelet also appears in this pagelet category on the Personalize Content page, which administrators and users access to manage pagelets on homepages and dashboards.

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 deselected, the pagelet displays data using the default data source parameter values provided.

Template Pagelet

Select to publish the pagelet as a 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 or WorkCenter pages 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 passed 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. Because users cannot customize template pagelets, user-specified parameters can show only the default values or the values that come through Context Manager key passing.

See [Chapter 17, "Creating and Configuring WorkCenter Pages," Configuring WorkCenter Pages, page 636.](#)

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 to make your pagelet available for use with external portals that support the WSRP standard.

See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Specifying Pagelet Publication Options, page 422.](#)

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 reregister the pagelet.

Security Type	<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-based or permission list-based security to the pagelet. The Selected Security scroll area appears.</p>
Author Access	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.
Update Security	<p>This button appears when Pagelet Wizard determines that an inconsistency exists 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>

Selected Security

Type	<p>Select the type of security to assign 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>
Name	Prompt to select or enter the name of the permission list or role to which to authorize access to the pagelet.

Buttons

Previous	Click to move backwards one step in the pagelet creation process in Pagelet Wizard.
-----------------	---

Finish

Click to complete the creation of the pagelet. The Pagelet Creation Confirmed page appears.

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.

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.
- Disable WSRP style substitution.
- 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

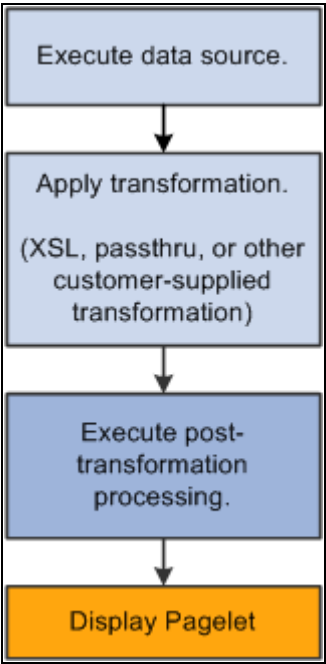
Pagelet Wizard enables you to use delivered Pagelet Wizard builder transformation tasks in custom and passthru transformations.

Listed in the order in which they are run in post-transformation processing, the following transformation tasks are not automatically generated for custom and passthru transformations, but they can be added by means of 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 Date, Time, DateTime, Currency, or Number elements. Variables that are user-supplied are set in italics.

Note. This processing step occurs only 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><PSFORMAT TYPE="DATE" [FORMAT="format string "] [FORMAT_3 character language code="format string "]>DateToFormat</PSFORMAT></pre> <p>Note. FORMAT has the same syntax as the DateTimeToLocalizedString() PeopleCode Function.</p>	<p>Formatted Date.</p> <p>Language-specific Formatted Date.</p>
Time	<pre><PSFORMAT TYPE="TIME" [SOURCETZ="Source Time Zone "] [DESTTZ="Destination Time Zone "] [FORMAT="format string "]>TimeToFormat</PSFORMAT></pre> <p>Note. 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><PSFORMAT TYPE="DATETIME" [SOURCETZ="Source Time Zone "] [DESTTZ="Destination Time Zone "] [FORMAT="format string "]>DateTimeToFormat</PSFORMA T></pre> <p>Note. 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><PSFORMAT TYPE="NUMBER" FORMAT="format string" [WIDTH="width"] [PRECISION="precision"]> NumberToFormat</PSFORMAT></pre> <p>Note. FORMAT, WIDTH, and PRECISION have the same format as the NumberToDisplayString() PeopleCode Function.</p>	Formatted Number.
Currency	<pre><PSFORMAT TYPE="CURRENCY" FORMAT="format string" [WIDTH="width"] [PRECISION="precision"] [CURRENCY_CODE="CODE "] [CURRENCY_DECORATION="SYMBOL/CODE/BOTH"]> NumberToFormat</PSFORMAT></pre> <p>Note. FORMAT, WIDTH, and PRECISION have the same format as the NumberToDisplayString() PeopleCode Function.</p> <p>CURRENCY_CODE can be one of the supported PeopleSoft currency codes, or if left blank, the default is the user's currency code.</p> <p>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. <PSNOWSRPSTYLESSUBST/> 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 Content Reference-Based Links

Use the following syntax to insert content-reference-based links into a Pagelet Wizard pagelet. Variables that are user-supplied are set in italics.

Note. This processing step occurs only 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	<PSREGISTEREDLINK PORTAL=" <i>portal name</i> " CREF=" <i>content reference name</i> " target=" <i>target</i> " APPEND=" <i>append</i> ">Text of Link</PSREGISTEREDLINK>	Text of Link

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 occurs only 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	<PSREGISTEREDURL 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 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 Pagelet Wizard Pagelets

This section lists common elements used in the section and discusses how to:

- Delete and copy Pagelet Wizard pagelets.
- Generate export and import scripts for Pagelet Wizard pagelets.
- Review Pagelet Wizard pagelet details.

Common Elements Used in This Section

This section discusses common elements used in this section.

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.*

Published Type	<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"> • <i>All Pagelets.</i> • <i>Embeddable Pagelet.</i> • <i>Homepage Pagelet.</i> • <i>Non-published Pagelet.</i> • <i>Template Pagelet.</i> • <i>WSRP.</i>
Portal Name	<p>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 signed in when creating the pagelet.</p>
Pagelet Keyword	<p>Enter keywords to narrow the range of pagelets that appear in the Pagelets scroll area. Pagelet elements that are used as keywords include pagelet ID, title, and description.</p>
Search	<p>Click to perform a pagelet search using the criteria entered in the Filter Pagelets group box.</p>
Reset	<p>Click to clear the Pagelet Keyword field.</p>

Pagelets

The Pagelets scroll area provides the following pagelet information.

Pagelet ID	Displays the pagelet ID. Click to access the pagelet definition in Pagelet Wizard.
Pagelet Title	Displays the pagelet title.
Homepage	If selected, indicates that the pagelet has been published as a homepage page.
Template	If selected, indicates that the pagelet has been published as a template pagelet.
WSRP	If selected, indicates that the pagelet has been published as a WSRP portlet.
Embeddable	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	<p>Delete one or more pagelets.</p> <p>Deleting a pagelet removes the pagelet definition from Pagelet Wizard, as well as deleting associated metadata and registry entries from the database and portal registry.</p>
Copy Pagelets	PTPPB_ADMIN_SAVEAS	PeopleTools, Portal, Pagelet Wizard, Copy Pagelets	<p>Create a new pagelet that is a clone of an existing pagelet. The new pagelet has a unique pagelet ID.</p> <p>The pagelet copy inherits all aspects of the original pagelet, with the exception of registration information.</p>
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 Pagelet Wizard defined pagelets.
Review Pagelets	PTPPB_ADMIN_REPORT	PeopleTools, Portal, Pagelet Wizard, Review Pagelets	Review the existing Pagelet Wizard-defined pagelets.

Deleting and Copying Pagelet Wizard Pagelets

This section discusses how to:

- Delete pagelets.
- Copy pagelets.

Deleting Pagelets

Access the Delete Pagelets page. (Select PeopleTools, Portal, 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:

*Published Type:

*Portal Name: 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"/>	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](#)

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.

Homepage If selected, indicates that the pagelet has been published as a homepage page.

Template	If selected, indicates that the pagelet has been published as a template pagelet.
WSRP	If selected, indicates that the pagelet has been published as a WSRP portlet.
Embeddable	If selected, indicates that the pagelet has been published as an embeddable pagelet.

Copying Pagelets

Access the Copy Pagelets page. (Select PeopleTools, Portal, 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

Pagelet ID	Pagelet Title
<input type="radio"/> DEMO FAC BUILDING	New Construction Update
<input type="radio"/> EOPP CREF BY DATE ID	Content References
<input type="radio"/> EOPP POPULAR PGLT LNG SCR	Popular Pagelets by Language
<input type="radio"/> EOPP POPULAR PGLT SRC	Popular Pagelets
<input type="radio"/> NEW EMPLOYEE UPDATE	For New Employees
<input type="radio"/> PAPI CFO INSIGHTS TOOLS	Investor Insights & Tools
<input type="radio"/> PAPP BRANDING HOMEPAGE	Branding Center
<input type="radio"/> PAPP COMPANY NEWS	Company News
<input type="radio"/> PAPP CONTENT MANAGER HOMEPAGE	Content Manager Center
<input type="radio"/> PAPP CONTEXT MANAGER HOMEPAGE	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.

Select	Select the pagelet to copy.
New Pagelet ID	<p>Enter a unique pagelet ID for the new pagelet that will be copied from the selected existing pagelet.</p> <hr/> <p>Note. Pagelet IDs can contain only 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/>
Include Personalization Data	Select to copy any end-user personalizations defined for the existing pagelet along with the new pagelet.
Save New Pagelet	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 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.	
Homepage	If selected, indicates that the pagelet has been published as a homepage page.
Template	If selected, indicates that the pagelet has been published as a template pagelet.
WSRP	If selected, indicates that the pagelet has been published as a WSRP portlet.
Embeddable	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 an overview of export and import scripts for Pagelet Wizard pagelets and discusses how to:

- Generate export and import pagelet scripts.
- Generate export and import scripts for Pagelet Wizard setup data.

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 content reference 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 or imported.

Generating Export and Import Pagelet Scripts

Access the Export/Import Pagelets page. (Select PeopleTools, Portal, 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:

*Published Type:

*Portal Name: Employee-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"/>	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**

Export/Import Pagelets page

Select Select the pagelets for which you want to generate export and import scripts.

Include Personalize Data Select to include any end-user personalizations defined for the selected pagelets in the export and import.

Generate Scripts

Click to generate export and import scripts for the selected pagelets. The scripts appear on the Data Mover Scripts page.

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.

See *PeopleTools 8.52: Data Management*, "Using PeopleSoft Data Mover."

Note. Other Pagelet Wizard-related Data Mover scripts are 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 and Import Scripts for Pagelet Wizard Setup Data

PeopleTools Portal Technologies delivers the following Data Mover scripts to export and 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, for example, from a development database to a production database:

- 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. (Select PeopleTools, Portal, 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
DEMO_FAC_BUILDING		VP1	01/13/2006 11:23:57AM	VP1
EOPP_CREF_BY_DATE_ID	02/09/2004 2:30:02.000000PM	PS	02/11/2004 3:02:21PM	PS
EOPP_POPULAR_PGLT_LNG_SCR		VP1	03/14/2004 1:15:29AM	VP1

Review Pagelets page

On any of the tabs for this component, you can click the Pagelet ID link 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

Category	Displays the category ID from step 1 of Pagelet Wizard.
Data Type	Displays the pagelet data type.
Display Format ID	Displays the format ID of the pagelet.

Pagelet Publishing Information

Homepage	If selected, indicates that the pagelet has been published as a homepage page.
-----------------	--

Template	If selected, indicates that the pagelet has been published as a template pagelet.
WSRP	If selected, indicates that the pagelet has been published as a WSRP portlet.
Embeddable	If selected, indicates that the pagelet has been published as an embeddable pagelet.
Cache	Displays a check box if caching is enabled for this pagelet.
Time	Displays 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 about any updates to the pagelet.

Chapter 12

Developing and Registering Pagelets

This chapter provides overviews of pagelets, PeopleSoft portal architecture, pagelet development, and single signon and pagelets. This chapter also discusses how to:

- Develop PeopleSoft component-based pagelets.
- Develop iScript-based pagelets.
- Develop Java-based pagelets.
- Develop contextual embeddable pagelets.
- Administer pagelets in the portal registry.
- Use attributes to enhance pagelets.
- Handle special situations.

Understanding Pagelets

This section discusses:

- Pagelets.
- Pagelet development.
- Pagelet extensions.
- Pagelet personalizations.

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, Oracle recommends 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 8.52 Hardware and Software Requirements*
- *Enterprise PeopleTools 8.52 PeopleBook: System and Server Administration*

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, widgets, or modules.

Homepage pagelets are designed to appear on tabbed homepages or dashboard pages. Homepage pagelets typically present general, high-level information; however, many are designed to provide the user with easy access to more detailed information either in another pagelet on the homepage or a target transaction page. Users can select the homepage pagelets that appear on his or her homepages by way of the Personalize Content link that is in the upper left corner of the homepage.

Template pagelets are small, narrow-format pagelets that are designed to appear in a template iframe, much like the Menu pagelet. Usually, you incorporate template pagelets into a portal template when you create and configure the template. You also use template pagelets when you configure the task panel of WorkCenter pages or context-sensitive pagelets that use Context Manager. You should design template pagelets so that they 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 PeopleTools portal homepages, dashboard pages, related content frames, or in the pagelet area of WorkCenter pages. Homepage pagelet size should correspond 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 pagelets should use the narrow column format of a pagelet. Banner pagelets span the width of the homepage or dashboard page. 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 compact as possible.

The homepage pagelets that appear on the portal homepage depend on several factors, including the type of portal that you implement, the permissions that you grant to the user, and the manner in which the user personalizes his or her homepage.

PeopleSoft pagelets are URL-accessible HTML documents that are designed to be size-appropriate for display on a homepage. Pagelets can also originate from remote, non-PeopleSoft systems such as static HTML files or HTML that is generated dynamically from a variety of technologies.

Pagelet Development

You can use PeopleTools to develop pagelets as smaller-than-normal pages, just as you would any other PeopleSoft transaction 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 PeopleSoft transaction 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 well-designed and well-developed pagelet serves these primary functions:

- Summarizes data that is relevant, personalized, and actionable. Page space is valuable; pagelets must be more than attractive.
- Provides links to more detailed information, such as content in another application.

In addition, pagelets should also:

- Aggregate data from multiple sources.
- Be simple and intuitive.
- Be sized appropriately.

The size of pagelets corresponds to the homepage layout that a user chooses. Any pagelet that you design must first conform to the dimensions of the narrow column and optionally, of the wide column. Pagelet width is the only dimension to which you must adhere. The height of pagelets can vary; however, good design principles suggest keeping them as short as possible.







- Avoid the use of *scrollable grids*.

Grids that you insert on pagelets are nonscrollable by default. If the grid appears on a pagelet and on a standard transaction page, Oracle recommends that you use the nonscrollable grid to ensure consistent user experiences.

Pagelet Header Icons

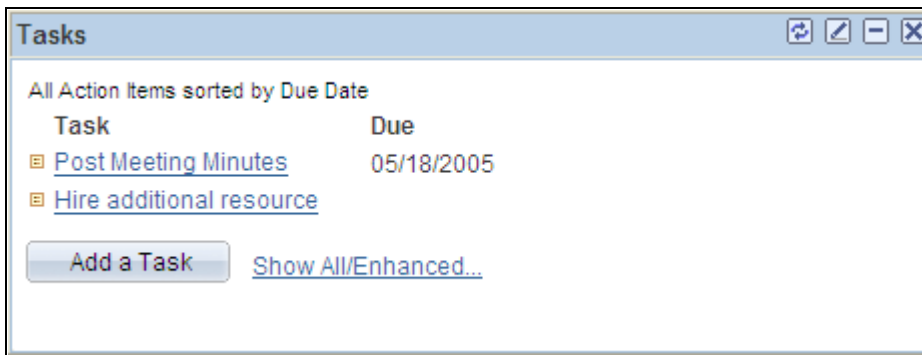
The icons that you see in the pagelet header function similarly across most pagelet types. You do not have to use all header icons on all pagelets. You can vary the icons that appear in the header based on business rules or technical constraints.

This table lists and describes the icons that you can use in pagelet headers.

 Expand	Click to show the content of the pagelet so that more than just the pagelet header is visible. When you click this icon, in its place you will see the Hide icon.
 Help	Click to view pagelet help. When you click this icon, any documentation that you configure appears. Not all pagelets provide documentation.
 Personalize	Click to personalize the pagelet. When you click this icon, a page appears so that you can change any pagelet settings that you are allowed to configure. The options that you can personalize vary among users and pagelets, and not all pagelets have options that you can personalize.
 Minimize	Click to hide the body of the pagelet so that only the header is visible. When you click this icon, in its place you will see the Show icon.
 Refresh	Click to force an application server trip so that the pagelet data and settings are current.
 Remove	Click to remove the pagelet from the page. You will be prompted to confirm the action and must accept or dismiss the confirmation.

Pagelet Example

The Tasks pagelet demonstrates many characteristics of a well-designed pagelet. In addition, notice the pagelet header icons:



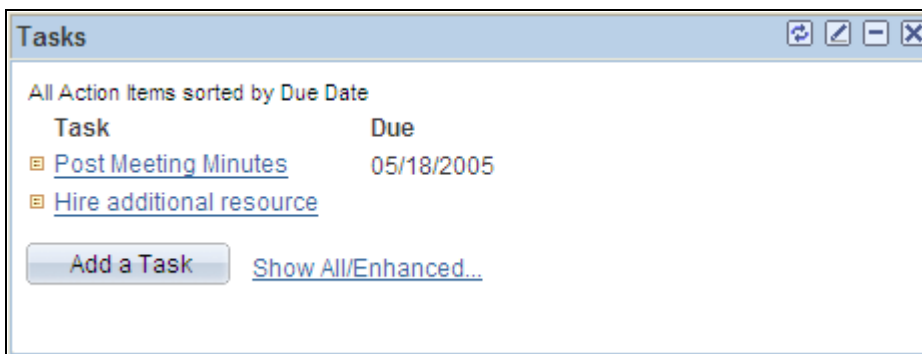
Tasks pagelet

Pagelet Extensions

Some pagelets use pagelet extensions, which are the supporting or supplementary components for a pagelet. They are not displayed on your homepage; you access pagelet extensions by associating them with 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.

You could configure a homepage pagelet so that each item is a link to the worklist item in another database, or to the personal task defined in the portal, or to a variety of resources.

The Tasks pagelet is a homepage pagelet that is designed to display a short list of relevant tasks. If dozens of tasks were open, which is likely over time, the pagelet would grow to be too long and possibly crowd out relevant, actionable data on other pagelets. The Tasks pagelet is configured with active links and buttons that access pagelet extensions, which enhance the functionality of the pagelet.



Tasks pagelet


Here you see two extension pages that you access from the Tasks pagelet:


- The Task Details page.


Click a link for a task name to access the Task Details page, which you use to view or modify personal task information. The Task Details page enhances the pagelet by showing more detail and enabling you to perform further actions, such as attaching a file or deleting the task.

Task Details

***Task:**

Due Date: 

Status: 

Priority: 

Task Description:

Review personnel report from Oscar's organization. He's requesting additional resources.

Attachment

File Name:

File Description:

Added By: Vice President of Finance

Date Added: 05/14/10 1:52:04PM

[Return to Tasks](#)

Task Details extension page

- The Tasks page.

Click the Show All/Enhanced link to access the Tasks page, which provides the user with more options such as the ability to display all or certain types of tasks. Because it is a standard page, the extension page can use a grid to retrieve and display a longer list of tasks. In addition, the extension page is also better suited to display more details because it is wider than the pagelet and can show more columns.

Tasks

5 Task(s)

View: All Tasks

Tasks					
	Task	Due Date	From	Status	Priority
	Post Meeting Minutes	05/18/2005	Vice President of Finance	Assigned	High
	Review EMEA Personnel Report	05/21/2010	Vice President of Finance	In Process	Medium
	Delegate email to admin	05/28/2010	Vice President of Finance	New	High
	PTO Starts	05/29/2010	Vice President of Finance	New	High
	Hire additional resource		PeopleSoft Demo Role User	On Hold	Medium

[Add a Task](#) [Routed Task Details](#)

Tasks page

Pagelet Personalizations

Like homepages, pagelets can be personalized in different ways. The data displayed can be filtered automatically to show only data that is 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:

- You do not click a button or link on the pagelet to access personalization pages. Instead, clicking the Personalize icon on the pagelet header accesses a personalization page where you can define user options that are specific to that pagelet.
- Setting values on the personalization page will change the way data appears on the pagelet for the user. Detail pages usually display data in read-only mode or allow you to change application data.

Note. Any text that the pagelet developer enters during Step 3 in Pagelet Wizard appears as instructions on the Personalize page.

Personalize Tasks

Use this page to specify the default values used in displaying your tasks.

Display Preferences

Tasks Displayed: All Tasks

Max Number of Displayed Rows: 05

Save [Return to Home](#)

Personalize Tasks page

You can use the Personalize Tasks personalization page for the Tasks pagelet 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

All Tasks sorted by Due Date

Task	Due
Post Meeting Minutes	05/18/2005
Review EMEA Personnel Report	05/21/2010
Delegate email to admin	05/28/2010
PTO Starts	05/29/2010
Hire additional resource	

Add a Task [Show All/Enhanced...](#)

Tasks pagelet after making personalizations

Understanding PeopleSoft Portal Architecture

This section discusses:

- PeopleSoft portal architecture.
- Client access.
- The web server.
- The application server.
- The database server.
- The portal registry.
- Portal components.

- Page assembly.

PeopleSoft Portal Architecture

When you develop pagelets, you need 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.

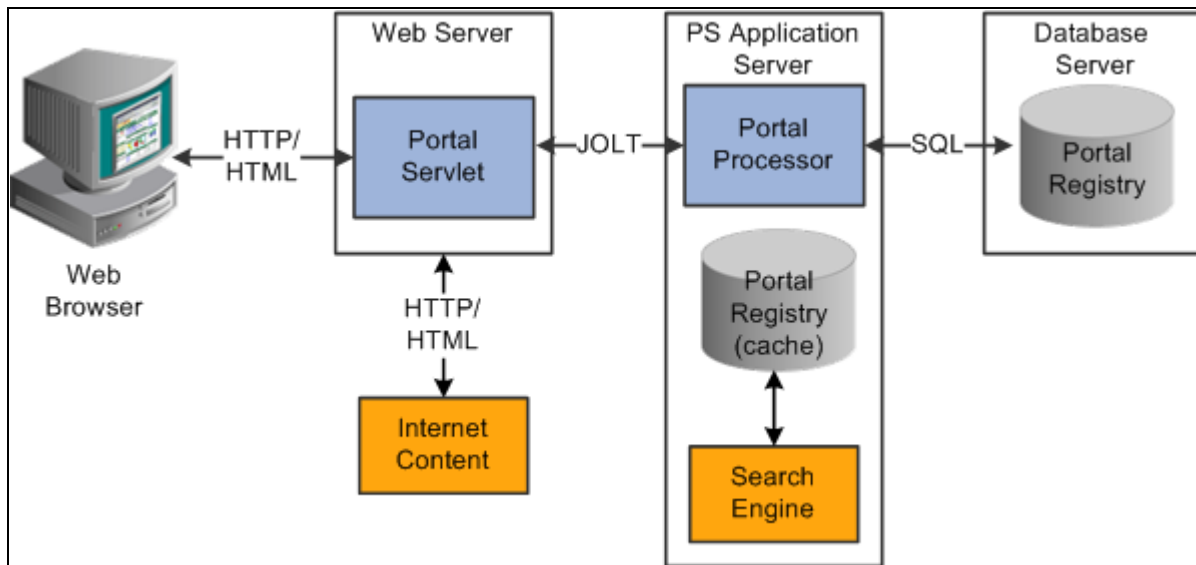
The PeopleSoft Pure Internet Architecture is the server-centric architecture used to deploy PeopleSoft internet applications to end users who access the applications through a web browser. This next generation architecture leverages a number of internet technologies and concepts to deliver simple, ubiquitous access to PeopleSoft applications and to enable the open flow of information between systems.

Using PeopleSoft Pure Internet Architecture as the foundation, you can 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.

The following diagram highlights these 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

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, such as:

- A web browser running on supported operating system.
- A wireless device or cell phone.
- An external or third-party system with extensible markup language (XML)/hypertext transfer protocol (HTTP) protocols.

No PeopleSoft executables are 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 he or she were navigating any other website.

A key concept of PeopleSoft Pure Internet Architecture is that no complex client software installation is required. The internet client device accessing the internet architecture already has all of the software and configuration that 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.

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 sign-in screen sent to the browser in HTML. The web server manages communication with the browser.

The following web server products can be configured to deploy your PeopleSoft applications:

- Oracle WebLogic server.
- IBM WebSphere server.

Two key PeopleSoft servlets run 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 as 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.

The Application Server

The application server is the core of PeopleSoft Pure Internet Architecture; it carries out 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.
- Run 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 runs as an application service of the PeopleSoft application server. It runs with the other application services, such as 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).

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 runtime architecture. The application server carries out business logic based on the PeopleSoft metadata.

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 carries out the business rules based on the definition.

Note. In general, PeopleSoft Application 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.

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. 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: Homepage, Navigation Collections, Pagelets, Template Pagelets, and Templates.

In addition to these standard folders, several folders typically are 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 that 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—that is, 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. PeopleTools provides portal administration pages 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.

Portal Components

The portal technology framework includes the following services:

- Personalization.

Pagelets optionally can 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 means of clicking the Personalize 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. PeopleTools search capabilities were built to assume multilanguage support, including double-byte languages.

- Navigation.

PeopleTools 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 on the role of the user.

- Menu navigation.

PeopleTools 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. Menu navigation is available through the Menu pagelet or from the Main Menu drop-down navigation depending on your portal settings.

- Favorites.

The PeopleTools portal includes a My Favorites folder that you use to store frequently accessed pages. In the Menu pagelet, this folder is at the root level. In drop-down navigation, My Favorites appears under the Favorites drop-down menu.

- Recently Used.

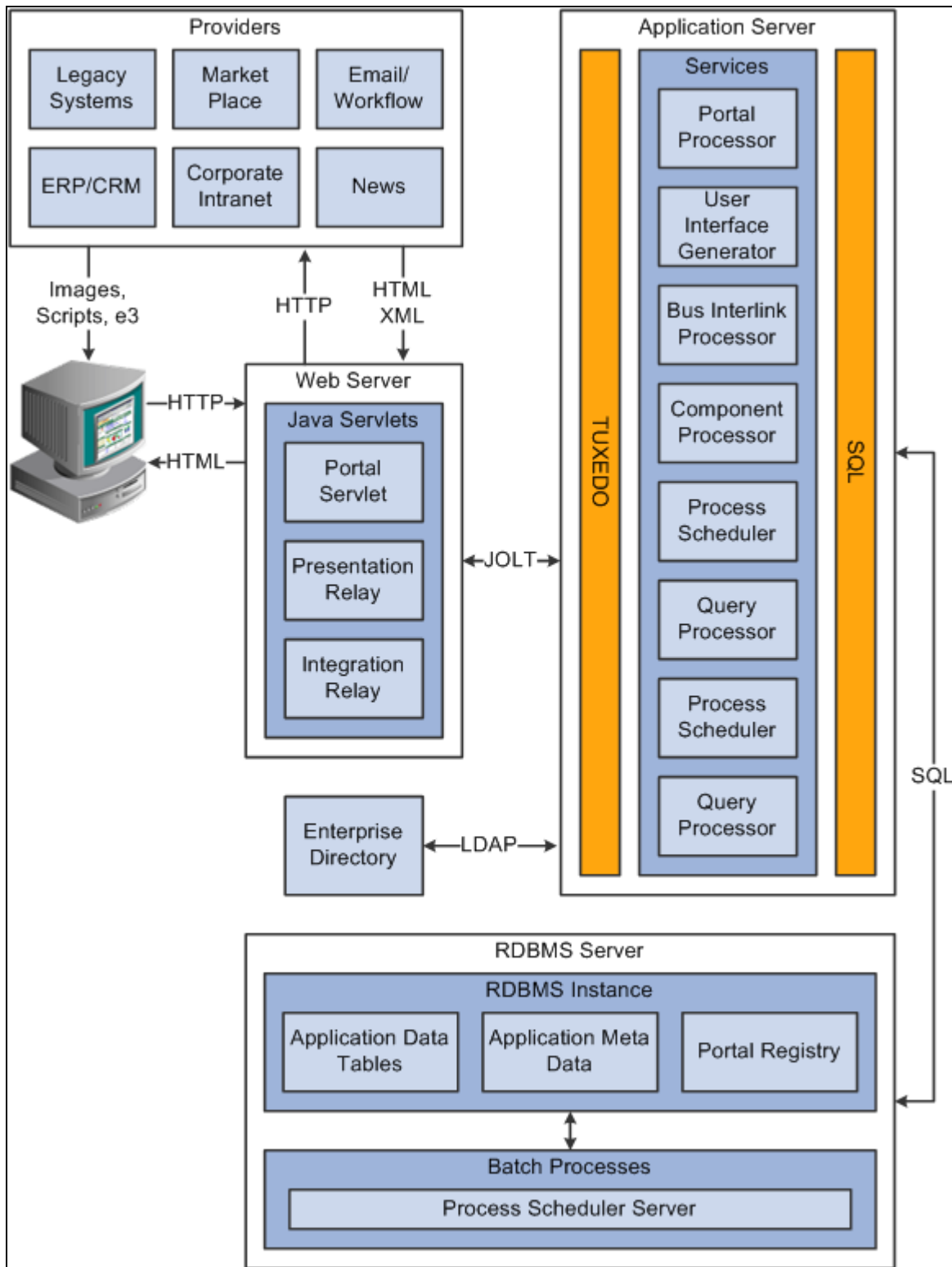
The PeopleTools portal include a Recently Used list that includes up to five of your most recently accessed components. This list appears under the Favorites drop-down menu.

See Also

[Chapter 5, "Administering Portals," Managing General Settings for Portals, page 172](#)

Page Assembly

Previous sections in this PeopleBook have described the individual components of PeopleSoft Pure Internet Architecture. This section provides a high-level overview.



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. Assume 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 link 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 discusses:

- Pagelet development guidelines.
- Pagelet development considerations.
- Pagelet development options.
- Size considerations.
- HTML design considerations.
- Branding considerations.

Pagelet Development Guidelines

Follow these guidelines when developing new pagelets:

- Adhere to object naming standards.
- Do not provide any filter, sort, or refresh buttons because they also require trips to the application server.
- If you enable the cache homepage option in the web profile, the pagelet must be display-only and cannot be interactive.

Because the portal servlet performs page assembly and proxying only on blocks of HTML, a pagelet must:

- Be URL-accessible.

The portal servlet will issue an HTTP request to the registered URL.

- Be HTML 4.0 compliant.

The HTML returned will be combined with HTML from other pagelets to form the entire web page.

- Be 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, "Developing and Registering Pagelets," Pagelet Sizes, page 469.](#)

- Use JavaScript namespacing.

If you write custom JavaScript code, ensure that JavaScript from multiple portlets can coexist on the same page. Only one namespace is shared by all portlets on a portal page. For example, if portletA and portletB both use the GETURL function, only one definition will exist 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, for example, 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 because 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.

Pagelet Development Considerations

When developing pagelets, you should also consider the following topics.

Single Signon

Oracle delivers a component interface named PRTL_SS_CI that enables external applications to support a single signon solution with PeopleSoft portal applications. This enables users to sign in once when accessing the portal and not have additional sign-ins for each system that is referenced in your portal.

See [Chapter 12, "Developing and Registering Pagelets," Understanding Single Signon and Pagelets, page 474.](#)

Layout

Observe 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 pagelet 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.
- Deselect the Show Borders option for group boxes and scroll areas.

- Select View, Internet Options in PeopleSoft Application Designer to ensure that you can access 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.

Appearance and Character

The appearance of objects on pagelets, such as text, links, tables, and so forth, should be modified by means of the PeopleSoft style classes to retain a consistent appearance and character between your pagelet and the rest of the portal. PeopleSoft uses 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:

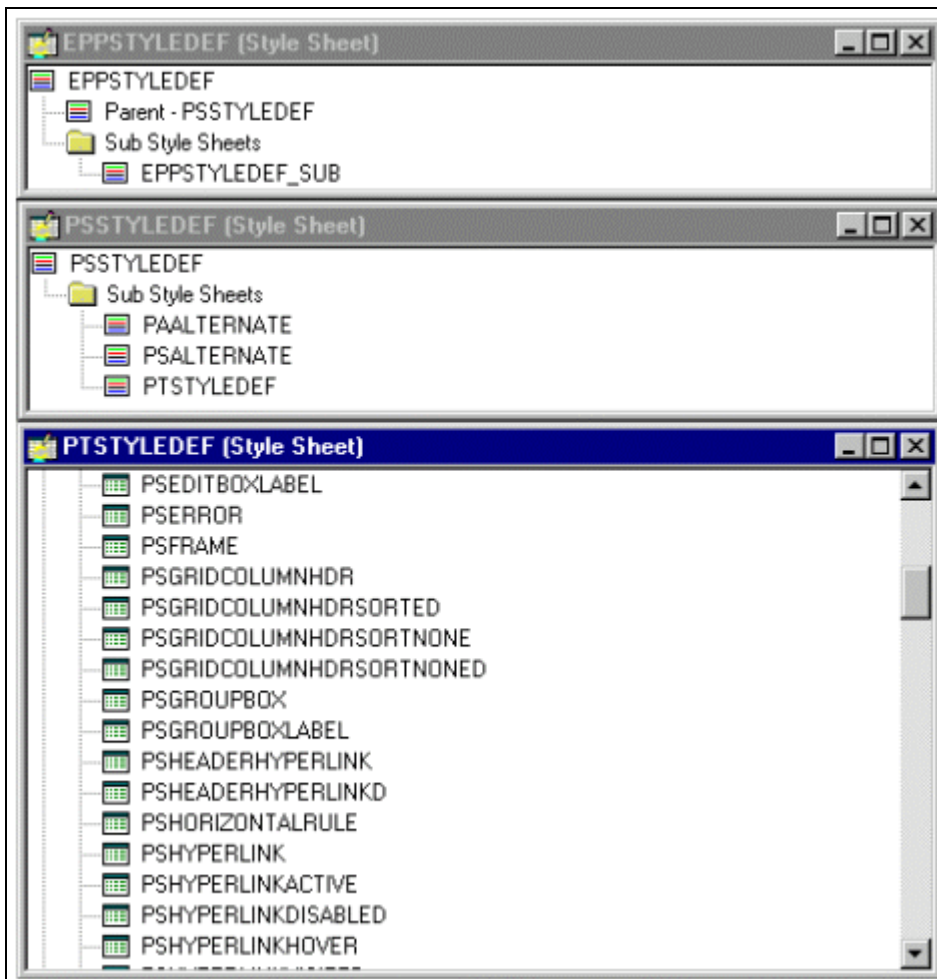
1. Template.
2. Target component.
3. Pagelets – in the order that they are assigned.

Note. In the same way that PeopleTools supports subrecords to enable shared definitions between records, you can define substyle sheets to share a common set of classes. A substyle sheet has all of the properties of a style sheet.

Style sheet definitions are stored in the database. They are accessed and modified by means of PeopleSoft Application Designer. The default style sheet used by PeopleTools depends on your application release, but is either PSSTYLEDEF or PSSTYLEDEF_SWAN. The PeopleSoft Application Portal uses a PSSTYLEDEF or PSSTYLEDEF_SWAN derivative.

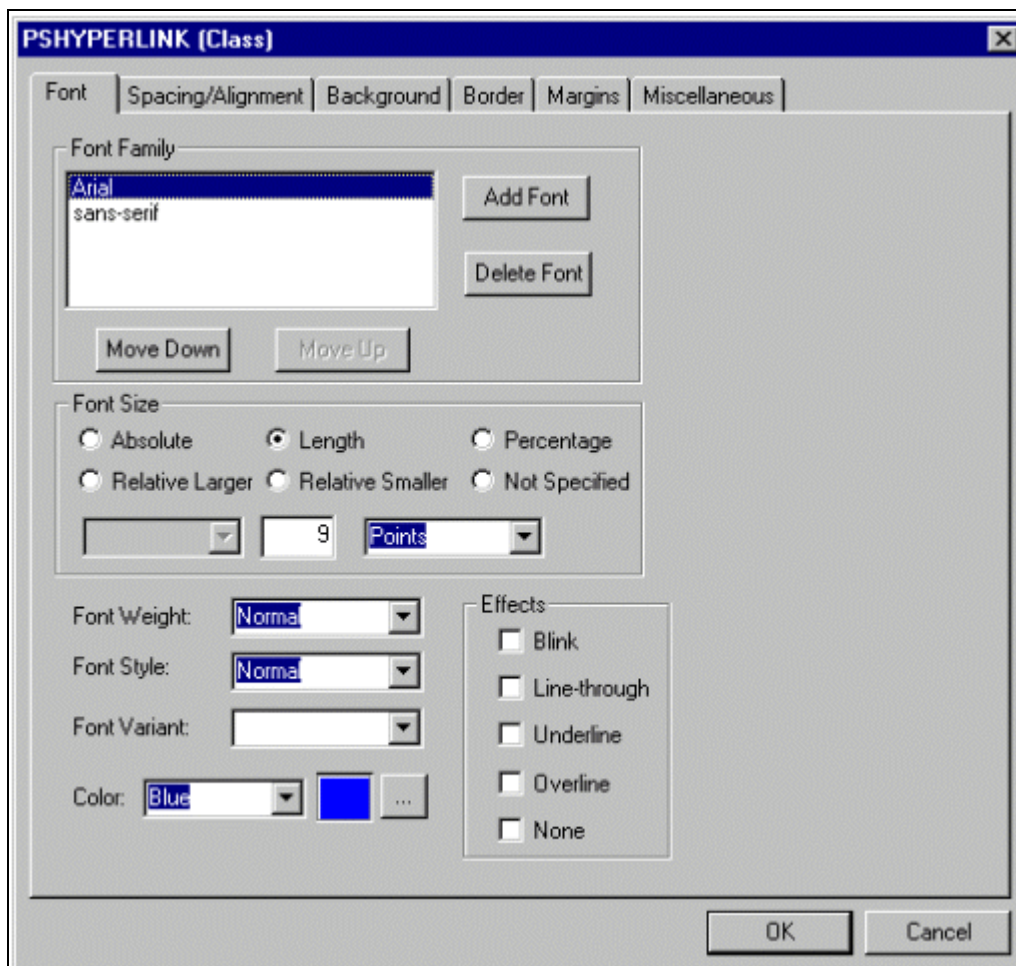
You can determine which style sheet is being used by selecting PeopleTools, Utilities, Administration, PeopleTools Options. The value in the Style Sheet Name field designates the style sheet used by the database.

Two ways are available by which you can see the style classes defined in a style sheet. If you access PeopleSoft Application Designer, you can open the designated style sheet and view its definitions in a graphical interface.



Style sheet definitions

Double-click 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\<site name>\cache.

The file is generally named PSSTYLEDEF_1.css or PSSTYLEDEF_SWAN_1; 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.

- PSPSMALLTEXT
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

The following examples show 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		
Language Code:	English	<input type="checkbox"/> Translations Change Last Updat
*Sort Order Option:	Binary Sorting	

Language Settings on the PeopleTools Options page

This HTML generates the following interface:

```
<input type='text' name='STYLESHEET' id=>
'SYLESHEET' tabindex='28' value="" class='PSEDITBOX' =>
style="width:23px; text-align:RIGHT; " maxlength='2' />
<label for='PSOPTIONS_TEMPTBLINSTANCES' class='PSEDITBOXLABEL'>=>
  Style Sheet Name:
</label>
```

Style Sheet Name:	EPPSTYLEDEF	
-------------------	-------------	---

Style Sheet Name field on the PeopleTools Options page

Links 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 signs in to a PeopleSoft portal, he or she signs 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, assume 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 Application 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 Application 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 enable or disable this feature, 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 Refresh button, the homepage is accessed from the web server again, overwriting the homepage cached on the browser.

Important! If any homepage pagelet in your implementation uses interactive mode, you must disable browser homepage caching in the web profile.

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 in 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 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 disable homepage caching for selected browser versions.

This can be useful if you have one or two supported browsers and want to disable cache for nonstandard 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 by 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, because the users' browsers already have cache control set by the previous value of the option, you must delete the browser cache for the new Time page held in cache value to take effect.

Oracle enables you 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 in which 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 completely disabling caching 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 enable 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 August 24, then the header information in the HTML is dated like this:

Wed 24 Aug 2011 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 no caching occurs 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 Page Expired message in .
- The setting overrides any date and time header settings.

The following table helps illustrate the way in which the EnableBrowserCache option works with the METAXP option.

<i>EnableBrowserCache</i>	<i>METAXP</i>	<i>Caching Behavior</i>
<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).

Multilanguage Support

PeopleSoft applications currently support the following 33 languages:

- Arabic.
- Bahasa Malay.
- Bulgarian.

- Canadian French.
- Chinese (Simplified).
- Chinese (Traditional).
- Croatian.
- Czech.
- Danish.
- Dutch.
- English.
- Finnish.
- French.
- German.
- Greek.
- Hebrew.
- Hungarian.
- Italian.
- Japanese.
- Korean.
- Norwegian.
- Polish.
- Portuguese.
- Romanian.
- Russian.
- Serbian.
- Slovak.
- Slovenian.
- Spanish.
- Swedish.
- Thai.
- Turkish.
- UK English.

Pagelet Wizard can access the PeopleSoft runtime environment to determine details of a user's profile, such as his or her language. For more information about how you can use this Pagelet Wizard feature to facilitate passing runtime parameters, such as language, see Pagelet Wizard documentation in this PeopleBook.

See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," page 323.](#)

In addition, Java classes delivered with PeopleTools enable you to call PeopleCode from your Java program and access contextual information from the runtime system. If needed, language can be retrieved through a PeopleCode function that is accessible from Java. Further information about this feature can be found in this document.

See [Chapter 12, "Developing and Registering Pagelets," Developing Java-Based Pagelets, page 486.](#)

Pagelet Development Tools

You can create a pagelet using several ways. 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 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 enables 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 multilanguage and browser support.

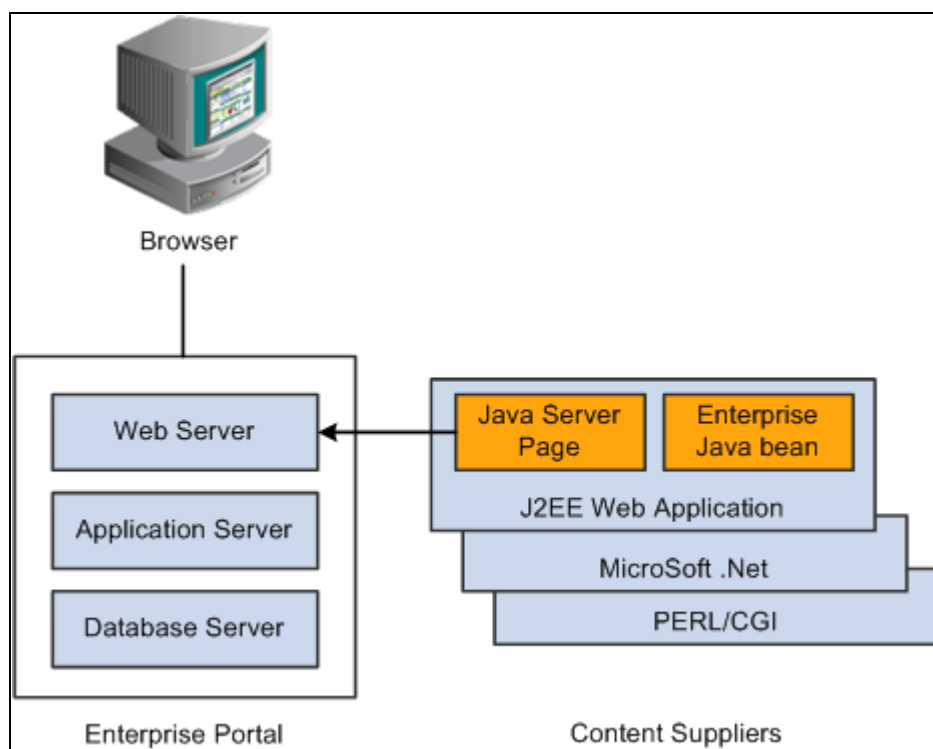
The focus of this document is a set of options that can be used to create a pagelet application with one of the many technologies that generate HTML. This may be the approach you take if you're integrating a non-PeopleSoft system with the PeopleSoft Application Portal. For example, you're applying business 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:

- Adobe ColdFusion.
- AJAX.
- ASP.
- CGI.
- HTML.
- Internet server API (ISAPI).

- JavaScript.
- JSP.
- Perl.
- Servlets.
- Tool Command Language (Tcl).

Here you see ways that you develop in a PeopleSoft environment using technology other than PeopleTools:



Developing in a PeopleSoft environment with non-PeopleTools technologies

You can convert your own HTML into a pagelet in several ways:

- Pagelet Wizard.

Pagelet Wizard walks you through a series of steps involved in creating and publishing a pagelet. Portal administrators and nontechnical users can 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 need 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, such as a JSP or ASP, directly in the portal registry.

Pagelet Sizes

The homepage layout can vary based on end-user personalization. These two layout choices are available for homepages and dashboard pages:

- The three-column layout divides the homepage into three equally-sized columns. In PeopleSoft applications, these are called narrow columns.
- The two-column layout divides the homepage into two differently-sized columns: one narrow column and one wide column, which is twice the width of the narrow column.

Because pagelets must function in two possible homepage layouts, it is column size that drives pagelet size. This fact is evident in the three categories of pagelet widths that are listed here:

- Narrow pagelets.
- Wide pagelets.
- Banner pagelets.

Narrow Pagelets

With three-column layout, all pagelets are rendered as narrow pagelets. Narrow pagelets are designed to be approximately 300 pixels wide. Subtracting the border and the internal margin leaves 285 pixels for the content. A narrow pagelet appears in a column that is one-third the width of the portal homepage. Because the pagelet is narrow, you should provide a succinct list of values that users can quickly traverse. Select the minimum set of data that best encapsulate the pagelet content. A narrow pagelet typically accommodates a single 30-character field.

A pagelet that has been designed to fit in a narrow column can be rendered in a wide column as well. However, user interface issues can occur 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.

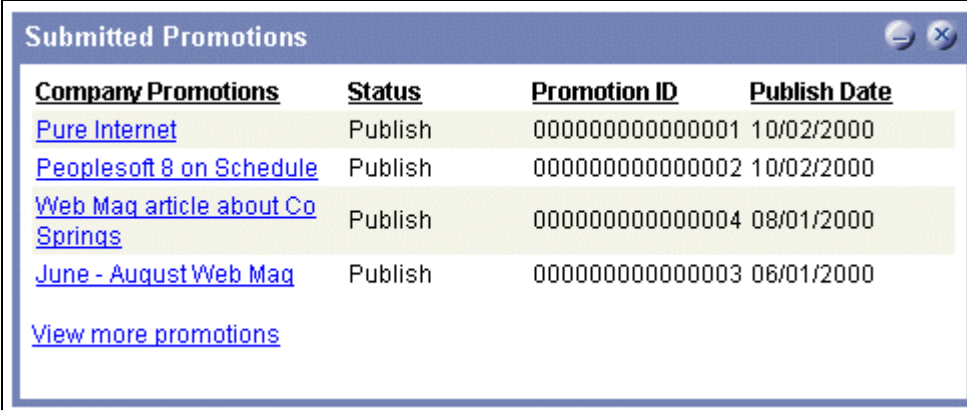


Sample narrow pagelet

Generally, all pagelets should be developed with the narrow column size as the default. However, a pagelet can be designed to take advantage of the additional space when it is rendered in a wide column.

Wide Pagelets

With two-column layout, the first column allows spacing for a narrow pagelet and the second column allows spacing for a wide pagelet. Wide pagelets are designed to span two-thirds of the width of the homepage. Subtracting the border and the internal margin leaves 650 pixels for the content. Although you can fit more data on the pagelet, the data must remain meaningful.



<u>Company Promotions</u>	<u>Status</u>	<u>Promotion ID</u>	<u>Publish Date</u>
Pure Internet	Publish	0000000000000001	10/02/2000
Peoplesoft 8 on Schedule	Publish	0000000000000002	10/02/2000
Web Mag article about Co Springs	Publish	0000000000000004	08/01/2000
June - August Web Mag	Publish	0000000000000003	06/01/2000
View more promotions			

Sample wide pagelet

When the layout contains the wide column, the **PORTALPARAM_COMPWIDTH** query string parameter is set to wide and passed to the HTML denote the wide version. Developing a wide version of a pagelet is optional.

The **PORTALPARAM_COMPWIDTH** Parameter

The parameter **PORTALPARAM_COMPWIDTH** can contain the value *Wide* or *Narrow* and can be used to specify whether a pagelet is being rendered in a narrow or wide column. The function **pgltAction** does not check or use this parameter, the narrow pagelet will just be rendered in a wider area.

This sample PeopleCode references the query string parameter **PORTALPARAM_COMPWIDTH**:

```
Component string &CompWidth;

&CompWidth = %Request.GetParameter("PORTALPARAM_COMPWIDTH");

If &CompWidth = "Wide" Then;
    TransferPage(Page.PT_CONTRK_PGLT_W);
End-If;
```

Banner Pagelets

Banner pagelets are the widest of all pagelets. Banner pagelets extend across the entire width of the homepage or dashboard page and span all columns regardless of layout. Both the two-column and three-column layouts enable you to add one banner pagelet that is positioned at the top of the page. Banner pagelets are designed to be 1240 pixels wide.

Note. The actual number of pixels that are available for content varies based on the resolution of your display and the rendering engine of your browser.

The portal uses the content reference attribute, *PORTAL_BANNER_PGLT* to appropriately position banner pagelets. You must manually enter this attribute name and its accompanying value.

This example shows a *PORTAL_BANNER_PGLT* attribute entry:

The screenshot shows a web form titled "Content Reference Attributes". It has a blue header bar. Below the header, there are three input fields: "Name" with the value "PORTAL_BANNER_PGLT", "Label" which is empty, and "Attribute value" with the value "1". To the right of the "Label" field is a checkbox labeled "Translate" which is unchecked. A "Delete" button is located to the right of the "Name" field. At the bottom left of the form is an "Add" button.

PORTAL_BANNER_PGLT attribute entry

Important! Homepages and dashboard pages can display one banner pagelet only. If more than one pagelet on a homepage includes the *PORTAL_BANNER_PGLT* attribute, the system recognizes the attribute for the first pagelet and ignores the attribute for any other pagelets.

See [Chapter 12, "Developing and Registering Pagelets," Creating Banner Pagelets, page 511.](#)

Pagelet Extensions

Pagelet extensions are standard pages. They can be registered with a template that allows it to use the entire width of the browser with no side frame for navigation, or with a template that includes a side frame.

In the former case, no inherent sizing requirements are imposed. PeopleSoft pages are designed for browsers running with a resolution of at least 1280 x 1024. Therefore, when a side frame is not included, a pagelet extension should not be wider than 1190 pixels, accounting for borders and so forth.

In the latter case, a pagelet extension should not be wider than 375 pixels.

HTML-Related Design Considerations

Consider these HTML-related guidelines when you are designing pagelets:

- Ensure that the pagelet encapsulates data to provide at-a-glance summary information.
- Ensure that the pagelet provides links to detailed application pages.
- Avoid large borders or a design that creates extraneous white space. Screen area 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 you should design your HTML to be vertically oriented. For instance, radio buttons should be arranged vertically, not side-by-side. When you create a form, buttons should appear below a text box, not next to it.
- Because this is a business-oriented portal, avoid extravagant or extraneous graphics. If you use graphics, they should be purposeful 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 that is greater than 218 pixels.
- In general, avoid explicit sizing. Let the browser render content in the space that is available, which should allow your text and graphics to fit appropriately. For example, place text in an HTML table to enable the table to wrap any long strings.
- When appropriate, use the PeopleSoft style sheet to ensure that the appearance and character of your pagelet is consistent with the rest of the PeopleTools portal content.
- When your pagelet does not use the entire width of the column, center the pagelet content.
- If a link 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 link 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, the pagelet should have a default mode. In default mode, ensure that the pagelet includes a standard set of data and a message conveying that default data can be personalized.

Application Portal Guidelines

This section lists some HTML-related development guidelines for pagelets that you use in the PeopleSoft Application Portal.

- Within PeopleTools, attributes from all content are merged into the single <BODY> tag; be aware of the order of attribute precedence.

The order of attribute precedence is:

1. Template.
2. Target content.
3. 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 area 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 you create a form, any buttons should appear under a text box, not next to it.

- Avoid using graphics that do not have thoughtful purpose. This is a business-oriented portal, so extravagant or extraneous images should not be used. If any are used, they should be purposeful 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, 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.
- When appropriate, use the PeopleSoft style sheet to ensure that the appearance and character of your pagelet is consistent with the rest of the Application Portal content.
- When your pagelet does not use the entire width of the column, center the pagelet content.
- If a link takes the user to another website and you want it to display the content in a new window, rather than using `target=_blank` within the link tag, use: ``

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, for a user to operate within the PeopleSoft Application 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 by means of text only and should be placed at the bottom of the pagelet. No organization logos can be placed on the pagelet without permission from Oracle. You can use "Provided by XXX," where XXX is the name of your company. Also, XXX can be a link 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.

If you wanted to create a pagelet that an associate could use to search for keywords on another website, it might look like the following example.



Sample pagelet

The HTML used is shown here:

```
<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>
```

Note how this sample pagelet adheres to the standards mentioned previously:

- It uses a table to help wrap words, rather than allowing long strings to dictate the width of the pagelet.
- All text uses the PeopleSoft styles, such as PSEDITBOXLABEL, PAADDITIONALINSTRUCTIONS, and PSPSMALLTEXT.
- The branding is small and no graphics are used, thus adhering to the branding standards mentioned previously.

Understanding Single Signon and Pagelets

This section discusses:

- Single signon.
- The PeopleSoft authentication process.
- The PeopleSoft authentication API.

Single Signon

The examples shown thus far have used publicly-available URLs. Even if the examples represent third-party applications, the discussion has concentrated on retrieving data and rendering a pagelet. The chapter has not yet discussed the possible need to sign in to a non-PeopleSoft system.

When two or more systems need separate authentication, automating the sign in process is preferable. Needing to manually sign in to several different systems each day is inconvenient and annoying. Users often expect a business portal to be similar to accessing a variety of internet websites. Once signed in to the portal, a user should rarely (if ever) need to sign in to another system.

Accomplishing single signon between PeopleSoft and other systems can be done in several ways. First, you need to determine the primary (or master) and secondary (or slave) authentication systems.

- PeopleSoft system as primary.

Once a PeopleSoft user has signed in, an authentication cookie is sent to the browser's memory. Other applications can choose to authenticate using this cookie. Oracle provides an API that other applications can leverage. This is the option that is discussed in detail in this section.

- PeopleSoft system as secondary.

The PeopleSoft authentication process is flexible enough to enable accessing another system.

See *PeopleTools 8.52: Security Administration*, "Employing Signon PeopleCode and User Exits."

- PeopleSoft system and other applications as secondary to third-party authentication system.

A variant of the previous option would be for all applications (including PeopleSoft) to leverage, or trust, a third-party authentication system such as Netegrity, Oblix, or Securant.

If you are writing a pagelet for the PeopleSoft portal, you have no guarantee that all possible customers for the pagelet will have access to a third-party authentication system. Thus, this option is not discussed in this PeopleBook.

PeopleSoft Authentication Process

Before discussing how your pagelet could leverage PeopleSoft authentication, you need to understand the authentication process.

After the first application server or node authenticates a user, the PeopleSoft application delivers a web browser cookie containing an authentication token. PeopleSoft Pure Internet Architecture uses web browser cookies to store a unique access token for all users 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 reauthenticate the user in the background so that they do not have to complete the sign-in 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 in through the portal, they always take advantage of single signon. Users need to sign in once and be able to navigate freely without encountering numerous sign-in screens.

The following table presents the fields that appear in the PeopleSoft authentication token.

Field	Description
UserID	Contains the user ID of the user to which the server issued the token. When the browser submits this token for single sign-on, this is the user that the application server signs in to the system.
Language Code	Specifies the language code of the user. When the system uses this token for single sign-on, it sets the language code for the session based on this value.
Date and Time Issued	<p>Specifies the date and time that the token was first issued. The system uses this field to enforce a time-out interval for the single sign-on token. Any application server that accepts tokens for sign-in has a <i>time-out minutes</i> 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 sign-on is not related to PeopleSoft Integration Broker messaging, except that single sign-on functionality leverages the messaging concept of nodes and local nodes.</p> <p>You configure only a node to trust single sign-on 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 determine whether it trusts the issued token.</p>

Field	Description
Signature	<p>Contains a digital signature that enables the application server using a token for single sign-on 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 for example,</p> <pre>signature = SHA1_Hash (UserID + Lang + Date Time issued + Issuing System + Local Node Pswd)</pre> <p>Note. The "+" indicates concatenation.</p> <p>Only one way is available to derive the 160 bits of data that make up the signature, and that is by hashing exactly the same user ID, language, date time, issuing system, and node password.</p> <p>Note. If you are using digital certificate authentication, the signature of the digital certificate occupies this space. The preceding description applies only to using password authentication.</p>

Note. Single signon does not depend on the use of an LDAP (Lightweight Directory Access Protocol) directory. You can store user credentials in an LDAP directory if desired, but it is not required.

These are the key security features of the cookie authentication token:

- The cookie exists in memory; it is not written to disk.
- No password is 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.
- The cookie is encrypted and digitally signed using a checksum to prevent tampering.

The PeopleSoft Authentication API

Oracle delivers a component interface named PRTL_SS_CI, which enables external applications to seamlessly integrate a single sign-on solution with the PeopleSoft portal applications. This component interface helps ensure that users who have already signed in to the portal don't have to sign in again for every system that 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 the portal. 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 *Enterprise PeopleTools 8.52 PeopleBook: PeopleSoft Component Interfaces*.

See *PeopleTools 8.52: 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, Oracle does 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 run 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 whether 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 reading about the development requirements of your API, 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 sign-in page.
2. If the sign-in to the portal application server is successful, the server generates a single signon token. The web server receives the single sign-on token from the application server and issues a cookie to the browser.
3. The user navigates in the portal and encounters a link 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 sign-in page.
6. After the system determines 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 Sign

Developers of external applications need to alter the sign-on process to conform to the following requirements:

1. Check for the PS_TOKEN cookie. If the cookie doesn't exist, continue with the normal sign-in process. Otherwise, bypass the sign-in page.
2. Retrieve the authentication token from the PS_TOKEN cookie.
3. Connect to the PeopleSoft portal 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 shows the process of validating your authentication token and retrieving the user ID. This example is designed to provide a general idea of the process involved and to help you incorporate the PRTL_SS_CI API into your sign-in 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 in to the portal, and it is not stored as data anywhere in the system. Therefore, no key fields are needed 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.CompIntfc.*;

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 PeopleSoft Component-Based Pagelets

To develop pagelets that are based on PeopleSoft components:

1. Open PeopleSoft 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 the page size.

For narrow pagelets, select *240xVar portal home page comp.*

For wide pagelets, select *490xVar portal home page comp.*

For banner pagelets, select *Custom size* then enter these width and height dimensions: *1193 x 259*.

Note. You can adjust the pagelet dimensions at any time in the process.

7. Design the new pagelet using the guidelines described previously and similar design techniques that are used to design a page.
8. Save the page.

Note. When the page is added to a component, the search record is specified. As noted in the preceding 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 the page.

See [Chapter 12, "Developing and Registering Pagelets," Administering Pagelets in the Portal Registry, page 500.](#)

Developing iScript-Based Pagelets

Use an iScript only for pagelets that reference non-PeopleSoft data or if a PeopleSoft page does not provide the necessary functionality. When you develop pagelets that are based on iScripts, you should be aware of the advantages and disadvantages.

Advantages

These are some advantages of iScripts:

- You may be familiar with the technique because it's similar to creating Active Server Pages (ASP) or Java Server Pages (JSP). Basically, you are using a script to produce HTML and JavaScript.
- Greater flexibility is available in pagelet appearance because you have direct control of the HTML or JavaScript.
- Greater flexibility is available in pagelet sizing. PeopleSoft Internet Architecture pages tend to be fairly static in their object placement and sizing.
- Many PeopleSoft Application Designer definitions and their attributes are available for you to take advantage of programmatically. For example, almost all PeopleSoft pages share a consistent appearance through the use of style sheets. Your pagelet, too, can use these styles and thus maintain uniformity of appearance with your PeopleSoft application because iScript-based pagelets can access the same style sheet definitions and their properties,

Disadvantages

These are some disadvantages of iScripts:

- You must manually program all aspects of the transaction. You must handle and account for support for multiple browsers and international considerations, such as language, date and currency formatting, and so on. PeopleTools handles these details automatically on transaction pages and component-based pagelets.
- You must be extremely meticulous and thoughtful. The poor design and implementation of many complex and interrelated programs can lead to loss of functionality and degradation of performance.

- When you develop 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 you want any such functionality, you must develop it.

Template Pagelets

To create an iScript-based template pagelet:

Note. Use steps 1 and 2 when you are getting HTML from another website or if you 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 begin with step 3 and create any necessary or complex PeopleCode to conditionally render the HTML or 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. Find 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, "Developing and Registering Pagelets," Administering Pagelets in the Portal Registry, page 500.](#)

Modifying Security for the Web Library

Security for the web library can be set up initially by means of 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 web library 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

PeopleTools 8.52: Security Administration, "Setting Up Permission Lists," Setting Web Library Permissions

PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide, "Using the Registration Wizard," Understanding the Registration Wizard

PeopleTools 8.52: PeopleCode API Reference, "Internet Script Classes (iScript)," iScript Security

Developing Java-Based Pagelets

This section provides overviews of how to develop pagelets in Java 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 Java Development

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 Application Portal. An example is development that is being done to integrate your non-PeopleSoft application with the PeopleSoft Application 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 Application Portal.

Two primary areas are available in which you can program in Java within the PeopleTools environment:

1. Writing business logic in Java.

You can use Java used to write business logic for PeopleSoft or non-PeopleSoft applications that are to be rendered in the PeopleSoft Application Portal. Java can be called from PeopleCode and the Java program can then reference the PeopleSoft runtime environment as needed.

In addition, Pagelet Wizard facilitates Java development in that it enables the Java programmer to concentrate on business logic, while Pagelet Wizard handles 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 by means of Java that is URL-addressable and produces HTML can be easily integrated into the PeopleSoft Application Portal through the PeopleSoft Application Portal registry functionality. This enables you to distribute your development and runtime environments making for better scalability for both.

For more information about leveraging the PeopleSoft Application Portal registry functionality to integrate your Java application with the PeopleSoft Application Portal, see the following documentation.

See [Chapter 12, "Developing and Registering Pagelets," Administering Pagelets in the Portal Registry, page 500.](#)

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 elements:

- Business rules.
- User interface.
- User personalization.
- Navigation.
- Navigation on extended pages.
- Portal integration.

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 runtime 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 to Create and Manage Pagelets," Understanding System Variables Supported as Data Source Parameters, page 373.](#)

Pagelet Wizard will also manage the user interface and storage of user personalization data. This topic is discussed in detail in the subsequent User Personalization section. 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 runtime 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 topic is discussed in the following User Interface section.

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 to Create and Manage Pagelets," Defining Pagelet Wizard XSL Prototypes, page 346.](#)

User Personalization

Pagelet Wizard supports the persistence of user personalization selections for a pagelet application and handles all aspects of the user interface that enable users to enter their preferences and selections. Personalization parameters are defined within Pagelet Wizard 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 want to include on their homepage. After they've selected a pagelet application and have decided where they want the pagelets to appear 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 enables an end user to select a pagelet application to appear on his or her 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 is presented within the category (folder) that is defined when they register the pagelet application through Pagelet Wizard.

See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Step 6: Specifying Pagelet Publishing Options, page 419.](#)

Pagelet Wizard also supports functionality that enables an end user to select values that allow a specific pagelet application to be personalized. The functionality may dictate that some input parameters that are 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 inspects the Java source code for input parameters. Pagelet Wizard administrators can then select which parameters can be accessed by an end user for personalization. The user interface to enables 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 enable a user to navigate to a related page for details and so forth, the link may need to 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 access and display 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 Application Portal, links that are included in a pagelet will be altered automatically 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

Clicking a link on an extended page displays content within the target frame. The PeopleSoft Application Portal, header, and left-hand menu area will remain. Therefore, the new content will be rendered in the context of the PeopleSoft Application 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 can access PeopleCode built-in functions, you can invoke the **`GenerateHomepagePortalURL()`** function from your Java program.

See [Chapter 12, "Developing and Registering Pagelets," Developing Java-Based Pagelets, page 486.](#)

See *PeopleTools 8.52: PeopleCode Language Reference*, "PeopleCode Built-in Functions."

Portal Integration

The pagelet application needs to be integrated with the portal registry so that the pagelet application is managed, secured, and presented through the PeopleTools portal. Registration makes the portal aware of a pagelet. 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

Many reasons exist why you would want to write business logic for your non-PeopleSoft applications (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

Three primary Java PeopleCode functions are used to invoke Java from PeopleCode:

- `GetJavaClass`
- `CreateJavaObject`
- `CreateJavaArray`

Java PeopleCode Function	Usage	Example
<code>GetJavaClass</code>	Finds a Java class that you can manipulate in PeopleCode. This is used for those classes that have static members for which instantiating an object of the class isn't appropriate. You can call only 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>In PeopleCode, you use the following code:</p> <pre>&Result = GetJavaClass("java.class.name").SomeStaticMethod();</pre> <p>The following example is simple PeopleCode that uses <code>GetJavaClass</code> to get a system class:</p> <pre>&Sys = GetJavaClass("java.lang.System"); &Sys.setProperty("java.security.policy", "C:\java\policy"); WinMessage("The security property is: " &Sys.getProperty("java.security.policy")); &Props = &Sys.getProperties(); &Props.put("java.security.policy", "C:\java\policy"); &Sys.setProperties(&Props); WinMessage("The security property is: " &Sys.getProperty("java.security.policy"));</pre>

Java PeopleCode Function	Usage	Example
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, use the following code to initialize an array:</p> <pre>intArray = new int[]{1, 2, 3, 5, 8, 13};</pre> <p>In PeopleCode, you use the following code to initialize a Java array:</p> <pre>&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>The following example 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 &oGpw; /* Create an instance of the object */ &oGpw = CreateJavaObject ("com.PeopleSoft.Random.Gpw_Demo"); &Q = "1"; /* Call the method within the class */ &NEW_VALUE = &oGpw.getNewPassword(&Q, PSRNDMPSWD.LENGTH); /* This is just returning one value for now */ PSRNDMPSWD.PSWD = &NEW_VALUE;</pre>
CreateJavaArray	<p>Enables 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 (&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 &Abc, &RefArray; Local array of String &Parms; &Parms = CreateArray(); /* Populate array how ever you want to populate it */ &Abc = GetJavaObject("com.peoplesoft.def.Abc"); /* Create the java array object. */ &JavaParms = CreateJavaArray("java.lang.String[]", &Parms.Len); /* Populate the java array from the PeopleCode array. */ &RefArray = GetJavaClass("java.lang.reflect.Array"); For &I = 1 to &Parms.Len &RefArray.set(&JavaParms, &I - 1, &Parms[&I]); End-For; /* Call the method. */ &Abc.xyz(&JavaParms);</pre>

Accessing the PeopleSoft Runtime System from Java

After a Java class has been invoked, the class can access the PeopleSoft runtime system. Java classes delivered with PeopleTools enable you to call PeopleCode from your Java program and access contextual information from the runtime 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. Hundreds of PeopleSoft system variables, constants, and built-in functions are available for use with this approach. Discussions of the various methods follow.

Accessing the runtime 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>PeopleTools 8.52: 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>PeopleTools 8.52: PeopleCode Developer's Guide</i>, "Understanding the PeopleCode Language," Constants.</p>
Name	<p>Enables you to use the PeopleSoft-reserved item references. This enables you to reference pages, components, records, fields, and so forth.</p> <p>For example, you can refer to a record field using the following PeopleCode:</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, you can refer to a PeopleSoft page using the following PeopleCode:</p> <pre>PAGE.pagename</pre> <p>In Java, it would be:</p> <pre>new PeopleSoft.PeopleCode.Name("PAGE", "PAGENAME");</pre> <p>See <i>PeopleTools 8.52: 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 that 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 prepended with either <i>get</i> or <i>set</i>, 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 &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"), => new Object[]{}); String menuName = "UTILITIES"; int nRecs = rs.Fill(new Object[]{"WHERE FILL.MENUNAME = :1", => menuName}); int i; int nFillChars = 0; for (i = 1; i <= 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=> menu item record. */ int nSQLChars = 0; Record menuRec = Func.CreateRecord(new Name("RECORD", => "PSMENUITEM")); SQL menuSQL = Func.CreateSQL("%SelectAll(:1) WHERE MENUNAME = :> 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 in the following way:</p> <pre> Local JavaObject &Test; Local number &chars; &Test = GetJavaClass("Test"); &chars = &Test.Test(); &Test = Null; WinMessage("The character counts found are: " &chars, 0); </pre> <p>See <i>PeopleTools 8.52: 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 for which the intention is 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 Application 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 Application Portal.

Some of the information in this section is covered in other Pagelet 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 runtime environment from Java is discussed in this section.

The following development scenario addresses the following elements of portal application development, integration, and implementation:

- Business rules.

Apply business logic and rules to data 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 to be placed on his or her homepage. 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 enable 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. Extended pages might show details as well as have links to enable a user to navigate to other related pages.

- Links.

Create links from your extended page to enable a user to navigate to a related page for further details and so forth.

- Return links.

Create links that enable the user to return to the portal's homepage.

- Portal integration and implementation.

Integrate the pagelet application with the portal registry so that the pagelet application is managed, secured, and presented through the PeopleTools portal.

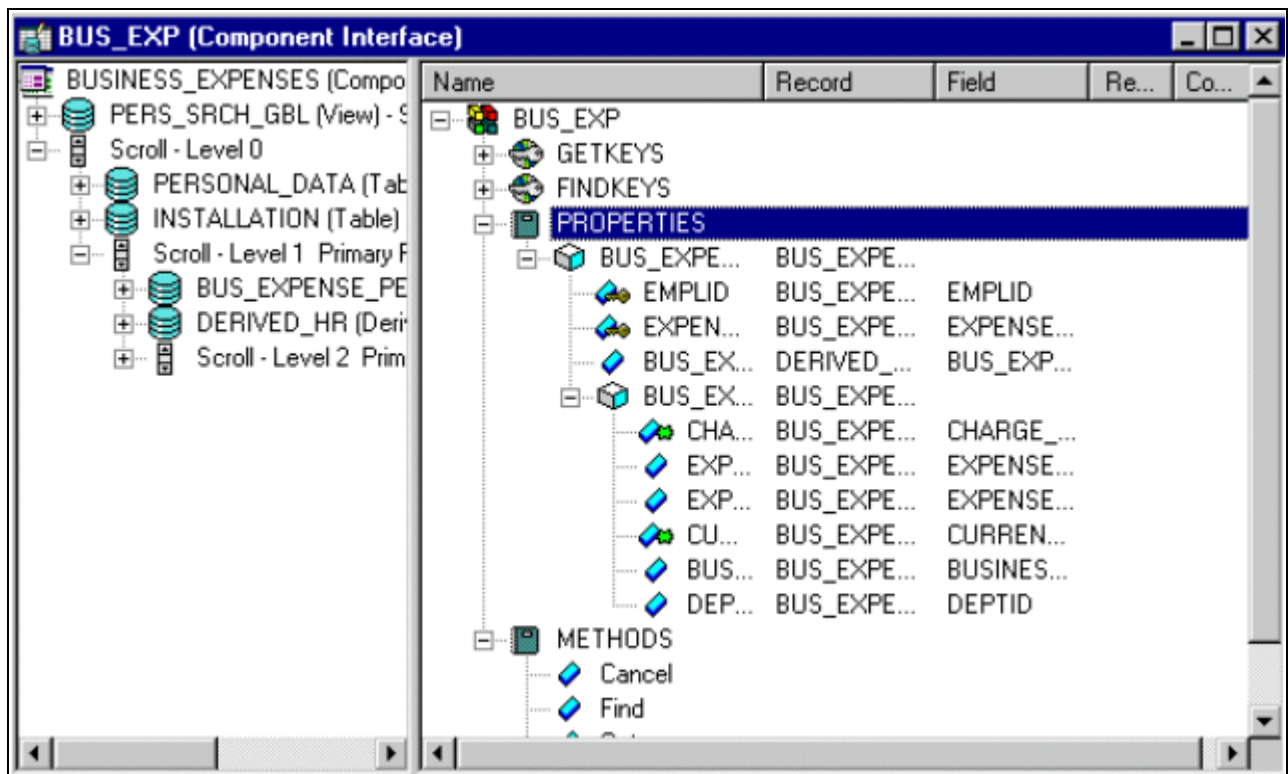
Invoking PeopleSoft Components from Java

All PeopleSoft components can be invoked from Java programs through component interface technology. This is useful for developers who want to create Java servlets, applets, JSP programs, or Enterprise JavaBeans that invoke PeopleSoft business logic. This section provides an example of how to invoke a PeopleSoft component from Java. A Business Expense component is used as the example.

Creating the Component Interface

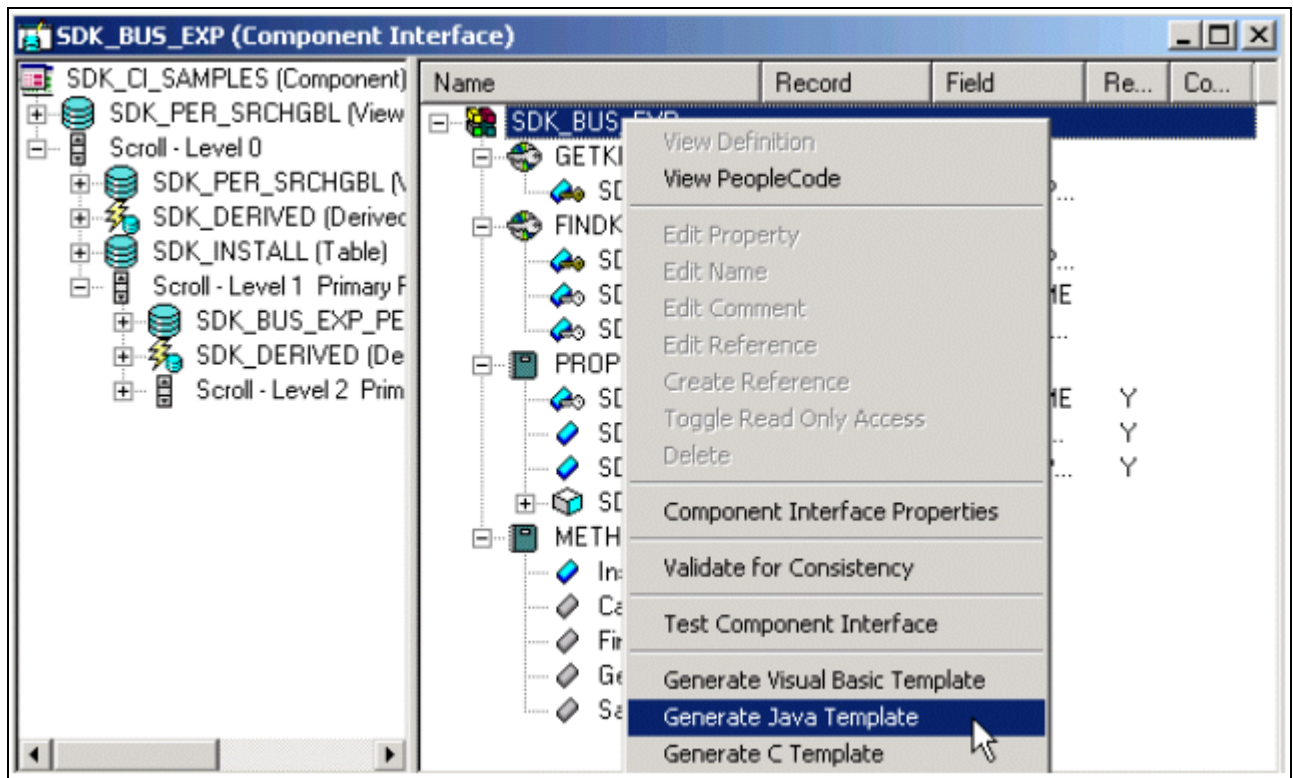
To expose a component to a third party, you must first create a component interface definition. You do this through the Component Interface Designer. Using drag-and-drop functionality, you can specify the properties and methods of the component that you want to expose. Numerous component interface definitions are delivered with the each PeopleSoft application.

The Business Expenses component definition appears in the left frame in the following example. The properties and methods that are exposed through this interface are displayed in the right frame.



BUS_EXP component interface definition

After the component interface definition is saved, you can then generate the Java classes for invoking this interface. You do this using the Component Interface Designer:



SDK_BUS_EXP component interface

See *PeopleTools 8.52: PeopleCode API Reference*, "Component Interface Classes."

Invoking the Component Interface from Java

To invoke 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 signs in to a PeopleSoft session. The **Connect()** method connects a session object to a PeopleSoft application server. Note that various options are 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 by means of simple assignment to the properties of the component interface and then the Get() method can be invoked. This populates the component interface with data based on the key values that you set; this 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, 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 by means of the **getName_OF_PROPERTY()** method of an item or by means of the generic **getPropertyByName()** method. This code illustrates the 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 by means of item objects and the **setNameOfProperty()** method of those items. Also note that you 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, a **DeleteItem()** method is available:

```
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 contextually relevant pagelets that you embed in transaction pages.

1. Create an embeddable pagelet using Pagelet Wizard.

See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," page 323.](#)

2. Create a transaction page definition using Application Designer.

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Creating Page Definitions."

3. Place an HTML area on the page definition.

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Using Page Controls," Inserting HTML Areas.

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

This PeopleCode example uses application packages to create an 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 = "EMBEDED_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 */
&DSParamColl = &myDataSource.getParameterCollection();
&CollectionParamArray = &DSParamColl.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
            &DSParameter = &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 you use Pagelet Wizard to register pagelets in the portal registry, Pagelet Wizard automates much of the process by interacting with the portal registry. You supply some key information in the pagelet definition, such as pagelet name, title, folder, and so on; Pagelet Wizard passes this information on to the portal registry.

However, the portal registry encapsulates other metadata about URLs that are 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 97

Registering Homepage and Template Pagelets

Before you can access a pagelet through the portal or associate a template pagelet with a target or WorkCenter page, you must register the pagelet in the portal registry. You can register pagelets by using the Component Registration Wizard, or by using the following procedure:

To register a new homepage pagelet or template pagelet:

1. Select PeopleTools, Portal, Structure and Content. Then:
 - For template pagelets, click these links in this order: Portal Objects, Template Pagelets.
Template pagelets must be registered in this folder.
 - For homepage pagelets, click these links in this order: Portal Objects, Pagelets. Then select any subfolder; for example, you can register a homepage pagelet in the Portal Objects, Pagelets, Organizers folder.
2. Click the Add Content Reference link.
 - For template pagelets, select *Target* as the usage type.
 - For homepage pagelets, select *Pagelet* as the usage type.
3. Select *Always use Local* as the node name.

4. Select the URL type based on the location of the pagelet.
 - Select *PeopleSoft Component* for component-based template pagelets.
 - Select *PeopleSoft Script* for iScript-based template pagelets.
 - Select *PeopleSoft Generic URL* for Pagelet Wizard-based pagelets.
5. Set pagelet attributes as necessary.
6. Set the content reference attributes as necessary.
7. Set additional parameters, if needed.
8. Save the content reference.

Modifying Pagelet Attributes and Security

Access the Content Ref Administration page. (Select PeopleTools, Portal, Structure and Content. Click the Edit link for a content reference.)

GeneralSecurity

Root > Portal Objects > Pagelets >

Content Ref Administration

*Name

*Label

Long Description
(254 Characters)

Product

Sequence number

Owner ID

Usage TypePagelet

Storage TypeRemote by URL

Author QEDMO

Parent Folder Pagelets

Creation Date 08/13/2011

URL Information

*Node Name

URL TypePeopleSoft Component

Component Parameters

*Menu Name

*MarketGBL

*Component

Additional Parameters

Example: name1=value1&name2=value2

Pagelet Attributes

Default ColumnColumn 1

Help ID

Refresh Time (sec)

☐ Hide minimize image
☐ Hide refresh image
☐ Force pagelet refresh

Edit URL Information

Node Name

URL Type

☐ Hide from MSF navigation

[IWC Message Events](#)

Content Reference Attributes

Name

Label

Attribute value

☒ Translate

[Attribute Information](#)

Add

Delete

Content Ref Administration page

The portal registry stores every content reference that is 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 affect the URL that will be used to reference the content associated with a definition.

See [Chapter 5, "Administering Portals," page 97.](#)

Access the Content Reference Security page. (Select PeopleTools, Portal, Structure and Content. Click the Edit link for a content reference. Select 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	View Definition
2	Permission List	PTPT1000	PeopleSoft User	View Definition

Inherited Security Authorizations				Customize Find  First 1 of 1 Last
	Type	Name	Description	View Definition
				View Definition

Content Reference Security page

You can grant and review the permission lists and roles that provide access to homepage tabs and template pagelets on the Content Reference Security page. The page allows for any combination of permission lists and roles. You can also make the object publicly available or available to the author only, which can be useful for testing purposes. Permission lists and roles can be used to grant access to the content item associated with this definition.

See [Chapter 5, "Administering Portals," page 97](#).

Registering URL-Based Pagelets

This sample content reference displays the properties and attributes for a URL-based pagelet:

General

Security

[Root](#) > [Portal Objects](#) > [Pagelets](#) > [Finance](#) >

Content Ref Administration

Name

ADMIN_PGDOC2_HMPG

Author

QEDMO

*Label

Sales by Region

Parent Folder

Finance

Copy object

Select New Parent Folder

Long Description

(254 Characters)

Product

ADMIN

Sequence number

Owner ID

PPT

PeopleTools

Usage Type

Pagelet

Storage Type

Remote by URL

Creation Date

08/11/2011

WSRP Producible

[Add Content Reference](#)

URL Information

Node Name

LOCAL_NODE

URL Type

PeopleSoft Generic URL

*Portal URL

s/WEBLIB_PTPPB.ISCRIPT1.FieldFormula.IScript_PageletBuilder?PTPPB_PAGELET_ID=PGDOC2

Pagelet Attributes

Default Column

Column 1

Help ID

Refresh Time (sec)

Hide minimize image

Hide refresh image

Edit URL Information

Node Name

URL Type

[IWC Message Events](#)

Content Reference Attributes

Name

PSCACHECONTROL

Delete

Label

PSCache-Control

Translate

Attribute value

private,max-age=60

Name

PTPPB_PAGELET_HELP_LINK

Delete

Label

Translate

Attribute value

http://salessupport.oracle.com

Name

PTPPB_PAGELET_ID

Delete

Label

Translate

Attribute value

PGDOC2

Add

504

Copyright © 1988, 2011, Oracle and/or its affiliates. All Rights Reserved.

Content Ref Administration 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.

Content Ref Administration

Author VP1

NamePT_PTAL_USER_PREF_GBL

Parent FolderTools - Hidden

*LabelPersonalize WorkCenter Page

Copy object

Select New Parent Folder

Long DescriptionPersonalize a WorkCenter Page.
(254 Characters)

ProductPT

Sequence number

Owner IDPPTPeopleTools

Usage TypeTarget

Storage TypeRemote by URL

Template Name

*Valid from date01/01/1900

Valid to date

Creation Date11/13/2009

☐ WSRP Producible

☐ No Template

Create Content Reference Link

Add Content Reference

Test Content Reference

URL Information

*Node NameLOCAL_NODE

URL TypePeopleSoft Component

Component Parameters

*Menu NamePTAL_ADMIN

*MarketGBL

*ComponentPTAL_USER_PREF

Additional Parameters

Example: name1=value1&name2=value2

☒ Hide from portal navigation

☐ Hide from MSF navigation

IWC Message Events

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 component (PTAL_USER_PREFS) is referenced in the Component Parameters group box. This is where the personalization page that is related to the pagelet application is identified.

Content Reference Security

Label: Personalize WorkCenter Page

☐ 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			
Personalize Find			
Type	Name	Description	View Definition
1 Permission List	PTPT1000	PeopleSoft User	View Definition
2 Permission List	PTPT1300	Portal Administrator	View Definition

Inherited Security Authorizations			
Personalize Find			
Type	Name	Description	View Definition
			View Definition

Content Reference Security page: Security tab

Using Attributes to Enhance Pagelets

This section provides an overview of pagelet and content reference attributes and discusses how to:

- Manage pagelet attributes.
- Configure pagelet Help links.
- Configure pagelet time-outs.
- Configure automatic pagelet refresh attributes.
- Create banner pagelets.
- Add images to pagelet headers.

Understanding Pagelet and Content Reference Attributes

Pagelet and content reference attributes provide a framework to specify and store free-form information about a content reference or pagelet. Attributes also enable you to manage pagelet behavior and appearance.

When you click a content reference, the portal retrieves the URL for the pagelet or content reference, it reads and processes the attribute and its value, and renders the page or pagelet incorporating the attribute.

You add attributes by entering Name-Value pairs. For example, an attribute name is `PORTAL_BANNER_PGLT`. You use this attribute to define a pagelet as a banner pagelet, which is the only pagelet that can span the enter width of the transaction page that appears in the browser in your PeopleSoft applications. The possible attribute values for this attribute are 1, 2, and 3. Each number represents different options regarding the appearance of the banner pagelet.

Page Used to Manage Pagelet Attributes

Page Name	Definition Name	Navigation	Usage
Content Ref Administration	PORTAL_CREF_ADM	PeopleTools, Portal, Structure and Content, Portal Objects, Homepage, Tabs. Click Add Content Reference.	Manage pagelet and content reference attributes.

Managing Pagelet Attributes

Access the Content Ref Administration page. (Select PeopleTools, Portal, Structure and Content. Click the Edit link for the appropriate content reference.)

The screenshot displays the 'Content Ref Administration' page. It features two main sections: 'Pagelet Attributes' and 'Content Reference Attributes'.

Pagelet Attributes: This section includes a 'Default Column' dropdown menu set to 'Column 1', a 'Help ID' text field, and a 'Refresh Time (sec)' text field. On the right, there are three checkboxes: 'Hide minimize image', 'Hide refresh image', and 'Force pagelet refresh'.

Edit URL Information: This section contains a 'Node Name' text field with a search icon and a 'URL Type' dropdown menu.

Content Reference Attributes: This section includes a 'Name' text field, a 'Label' text field, and a large 'Attribute value' text area. There is a 'Delete' button next to the 'Name' field. A 'Translate' checkbox is checked, and there is a link to 'Attribute Information'. At the bottom left, there is an 'Add' button.

Additional UI elements include a 'Hide from MSF navigation' checkbox and a link to 'IWC Message Events'.

Pagelet and Content Reference Attributes fields on the Content Ref Administration page

1. Select *Pagelet* in the Usage Type drop-down list box.
2. Use the Pagelet Attributes group box to select the default column for a pagelet, associate a help topic with the pagelet, set refresh time, and hide certain buttons from users.

3. Use the Content Reference Attributes group box to create banner pagelets and add images to pagelet headers

Pagelet Attributes

Default Column

Select the column in which the pagelet appears.

Help ID

Enter the help ID for the pagelet. When users click the Help button on the pagelet, the system displays help information that is specific to the pagelet.

This feature works only if you specified a help URL on the Web Profile Configuration - General page, and the pagelet documentation is part of the HTML PeopleBooks that are identified by the help URL.

You must also ensure that the documentation HTML includes a properly formatted anchor element that uses the value that you specify in this field. For example, if you specify a help ID of *MY_PAGELET_CONTENT*, the pagelet documentation in the PeopleBook must contain the following element:

```
<a NAME="F1ID_MY_PAGELET_CONTENT"></a>
```

Refresh Time (sec)

Enter the number of seconds after which the system auto-refreshes the pagelet. To activate the refresh timer, enter a value of 1 or greater.

The timer refreshes the pagelet as soon as the user accesses the homepage and then automatically refreshes the pagelet every *n* number of seconds.

Note. Homepages and dashboards use automatic refresh.

Hide minimize image

Select to hide the minimize button that normally appears in the pagelet headers so that users are prevented from minimizing the pagelet.

Hide refresh image

Select to hide the refresh button to prevent users from refreshing the pagelet.

If you implemented pagelet caching for this pagelet, a refresh button automatically appears in the pagelet header.

Force Pagelet Refresh

Some components can render themselves differently based on the column width. When a homepage has a 2-column layout and the user moves a component-based pagelet to a different sized column by using the drag-and-drop method, this pagelet needs to be refreshed so that the content displays correctly.

The drag-and-drop iScript that persists the pagelet changes forces a homepage refresh only when all of the following conditions are met:

- A pagelet changes columns.
- The homepage layout is 2-column.
- The pagelet is component-based.
- You select the Force page refresh property for the pagelet.

Note. When `document.write` or `document.writeln` scripts are detected inside a pagelet, the whole homepage will be refreshed instead of just that pagelet. This is required to properly assemble that pagelet's content.

See [Chapter 8, "Using Portal Caching Features," Implementing Pagelet Caching, page 224.](#)

Edit URL Information

Use the Node Name and URL Type fields to specify a page to be used for personalizing this pagelet and to make the personalization button appear in the pagelet header.

Node Name	Select the node name for the edit URL of the pagelet. This node name can be different from the pagelet node name.
URL Type	<p>Select the type of edit URL. The specific URL entry fields appear based on the setting of this field. Options are:</p> <p><i>Non-PeopleSoft URL:</i> The Edit URL field appears. Enter the URL of the personalization page to use.</p> <p><i>PeopleSoft Component:</i> The Component Parameters group box appears, containing the same fields as the Component Parameters group box for the content reference. Use these fields to identify the personalization page to use.</p> <p><i>PeopleSoft Script:</i> The iScript Parameters group box appears, containing the same fields as the iScript Parameters group box for the content reference. Use these fields to identify the personalization page to use.</p>

Content Reference Attributes

You can use the content reference attributes of a pagelet to change its character and appearance and to improve its usability.

Use content reference attributes to:

- Configure pagelet help links.
- Configure pagelet time-out settings.
- Configure automatic pagelet refresh.
- Create banner pagelets.
- Add icons to pagelet headers.

See Also

[Chapter 5, "Administering Portals," Administering Content References, page 143](#)

[Chapter 12, "Developing and Registering Pagelets," Configuring Pagelet Help Links, page 510](#)

[Chapter 12, "Developing and Registering Pagelets," Configuring Pagelet Time-out Settings, page 510](#)

[Chapter 12, "Developing and Registering Pagelets," Configuring Automatic Pagelet Refresh, page 511](#)

[Chapter 12, "Developing and Registering Pagelets," Creating Banner Pagelets, page 511](#)

[Chapter 12, "Developing and Registering Pagelets," Adding Images to Pagelet Headers, page 512](#)

Configuring Pagelet Help Links

To specify pagelet help:

1. Select PeopleTools, Portal, Structure and Content, Portal Objects, and navigate to the Content Ref Administration (content reference administration) page for the pagelet.

You define the pagelet help in the Content Reference Attributes region of the page.

2. Enter *PTPPB_PAGELET_HELP_LINK* in the Name field.
3. Leave the Label field blank.
4. Deselect the Translate check box.

Note. You *must* deselect the Translate check box or unexpected results might occur.

5. In the Attribute Value field, enter the URL of the location of your help document, for example *http://www.mycompany.com/help_doc1.html*
6. Save the content reference.

Note. The pagelet Help URL takes precedence over the Help ID when both attributes are set for a pagelet.

Note. The Help button appears only when the pagelet is on the homepage inside the PeopleSoft Portal; it does not appear when the pagelet is displayed as a WSRP portlet.

See Also

[Chapter 5, "Administering Portals," Administering Content References, page 143](#)

Configuring Pagelet Time-out Settings

To specify a pagelet time-out:

1. Select PeopleTools, Portal, Structure and Content, Portal Objects, and navigate to the Content Ref Administration (content reference administration) page for the pagelet.

You define the pagelet time-out in the Content Reference Attributes region of the page.

2. Enter *PSTIMEOUT* in the Name field.
3. Enter *PSTimeout* in the Label field.
4. In the Attribute Value field, enter the number of seconds before the pagelet is considered unavailable. This can be any positive integer.

Note. The portal ignores this attribute if you enter 0 or a negative value, or if the content reference isn't a pagelet.

5. Save the content reference.

When your specified time-out expires, the portal stops attempting to load the pagelet and generates an error message.

See Also

[Chapter 5, "Administering Portals," Administering Content References, page 143](#)

[Chapter 9, "Configuring the Portal Environment," Configuring Appearance and Character, page 267](#)

Configuring Automatic Pagelet Refresh

To specify a pagelet refresh time in seconds:

1. Select PeopleTools, Portal, Structure and Content, Portal Objects, and navigate to the Content Ref Administration (content reference administration) page for the pagelet.

You define the pagelet time-out in the Content Reference Attributes region of the page.

2. Enter the number of seconds for automatic refresh in the Refresh Time (sec) field.
3. (Optional) Select the Hide refresh image check box to prevent manual pagelet refresh by the user.
4. Save the content reference.

Note. Pagelets with document.write or document.writeln should not have the refresh timer set (to automatically refresh every x number of seconds) because this requires that the whole homepage be refreshed every x number of seconds.

Creating Banner Pagelets

To create banner pagelets:

1. Create and register your pagelet.

See [Chapter 12, "Developing and Registering Pagelets," Administering Pagelets in the Portal Registry, page 500.](#)

See [Chapter 12, "Developing and Registering Pagelets," Understanding Pagelet Development, page 457.](#)

2. Access the Structure and Content page (Select PeopleTools, Portal, Structure and Content).
3. Select Portal Objects, then Pagelets in the portal registry folder structure and click the Add Content Reference link.
4. If you have already registered the pagelet, navigate to the Pagelets folder and click the Edit link that is associated with the pagelet.
5. In the Content Reference Attributes group box, enter *PORTAL_BANNER_PGLT* in the Name field.
6. In the Attribute value field, enter one of these values:
 - 1 - The banner pagelet will display a pagelet header and a border around the pagelet content like a standard pagelet.
 - 2 - The banner pagelet will display a border around the pagelet content, but *no* pagelet header.
 - 3 - The banner pagelet will display *neither* a pagelet header *nor* a border around the pagelet content.
7. Add the pagelet to the homepage.

See [Chapter 10, "Administering Homepages, Dashboard Pages and Pagelets," Selecting the Pagelets That Appear on Homepages, page 308.](#)

8. Click the Save button.

Adding Images to Pagelet Headers

To add images to pagelet headers:

1. In PeopleSoft Application Designer, create an image definition and store the image in the image catalog.
2. In your browser, access the Structure and Content page (Select PeopleTools, Portal, Structure and Content.)
3. Navigate through the Portal Objects folder structure to the Pagelets folder until you find the pagelet content reference. Then, click the Edit link.
4. In the Content Reference Attributes group box, enter *PT_PORTAL_PGLT_HDR_IMG* in the Name field.
5. In the Attribute value field, enter the name of the image definition that you want to appear in the pagelet header.

Important! The image definition must exist in the image catalog of the PeopleTools portal database. Use PeopleSoft Application Designer to manage the image definitions that comprise the image catalog.

6. Click the Save button.

See Also

Creating Image Definitions, *PeopleTools 8.52 PeopleBook: PeopleSoft Application Designer Developer's Guide*

Handling Special Situations

This section discusses how to:

- Determine pagelet running location.
- Use refresh tags in pagelets.

Determining Pagelet Running Location

You can determine whether your code is running within the portal environment (that is being invoked by means of 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 identifies the environment. This variable works in both iframe templates and HTML templates.

Using Refresh Tags in Pagelets

The order of precedence of refresh tags in pagelets is:

1. Template
2. Target content
3. Pagelet

Among pagelets, the first pagelet in the HTML to include a metarefresh tag is included. Subsequently found refresh tags are not included in the HTML.

See Also

[Chapter 8, "Using Portal Caching Features," page 221](#)

[Chapter 6, "Modifying the Portal Interface," Applying Changes, page 201](#)

Chapter 13

Configuring OBIEE Report Pagelets

This chapter provides an overview of Oracle Business Intelligence Enterprise Edition (OBIEE) integration 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.
- Create OBIEE report pagelets.

Understanding OBIEE Integration Within the PeopleSoft Framework

The OBIEE analytics servlet enables 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 Service Oriented Architecture Protocol (SOAP) requests to the OBIEE server.

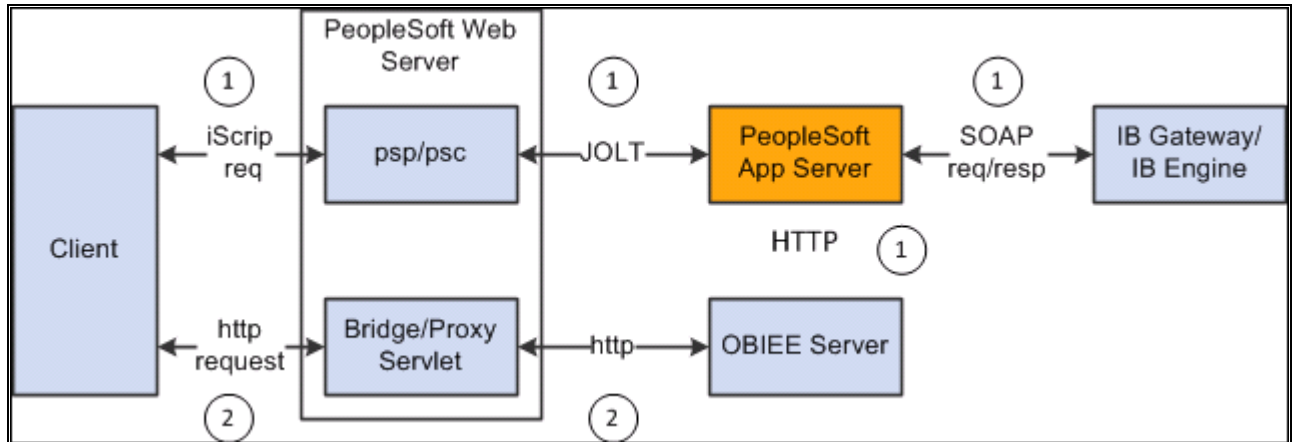
PeopleSoft applications store the credentials of an OBIEE administrator—a user with adequate privileges—in the PeopleSoft database and use these credentials to sign in to the OBIEE server, impersonate a real OBIEE user, and establish a session. The OBIEE server returns a session ID for this user session, 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 sign-inscreen 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 generates the basic HTML markup for the PeopleSoft application to display. The application uses this basic HTML to send HTTP requests to the OBIEE server from the browser. Before reaching the OBIEE server, the HTTP requests pass through the PeopleSoft bridge servlet. The bridge servlet:

1. Acts as a reverse proxy server to avoid any cross domain issues.
2. Checks the validity of the OBIEE server for each request
3. Forwards each request to the OBIEE server by copying the PeopleSoft headers—including PeopleSoft cookies.

4. Receives and unpacks the OBIEE server response.
5. Loads the unpacked server response into the client.
6. Renders the OBIEE report in the PeopleSoft system

This diagram shows the architecture and the flow of information among the PeopleSoft web server, the PeopleSoft application server, the OBIEE server, and the Integration Broker gateway.



PeopleSoft Architecture with OBIEE Integration

Note. OBIEE developers design and create reports in the OBIEE system. The OBIEE server generates the complete HTML markup for the report. The PeopleSoft system does not format the report; rather the HTML passes through the PeopleSoft system directly to the pagelet to be displayed as formatted by the OBIEE developer.

You should include only one OBIEE report on any PeopleSoft page or homepage. PeopleSoft applications do not currently support the appearance of multiple OBIEE reports on the same page, whether from the same OBIEE server or from multiple OBIEE servers.

Configuring OBIEE Server Setup

Access the OBIEE Server Setup page. (Select PeopleTools, Portal, OBIEE Setup, Server Setup.)

Server Setup

***Analytics URL:**
 (Example: http://<machine>:<port>/<Analytic Server Name>)

***OBIEE software ver.** ☐ **Active Server**

***OBIEE Server:**

Description:

***Admin User ID:**

Password:

Confirm Password:

Server Setup page

Use the OBIEE Server Setup page to define the location and administrative information that is necessary to access the OBIEE reports.

Analytics URL	Enter the URL of the OBIEE server, for example, <code>http://ptsft12.us.oracle.com:8739/AnalyticServer1</code>
OBIEE software ver.	Specify the OBIEE software version. PeopleSoft applications support OBIEE 10g and 11g. The default value is <i>OBIEE 11g</i> .
OBIEE Server	This value is supplied by default 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 you are 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 in when the system displays the OBIEE pagelet.
<hr/> Note. This user should already exist in the OBIEE server. <hr/>	
Password	Enter the OBIEE administrator's password.
Confirm Password	Reenter the OBIEE administrator's password for confirmation.

Configure the Web Server Files

You must include the OBIEE server URL in the configuration.properties file.

To configure the web server files:

1. Using a text editor, open the configuration.properties file in the web server deployment folder located here: `<Web_server_deployment_home>\applications\peoplesoft\PORTAL.war\WEB-INF\psftdocs\ps\configuration.properties`.
2. Find `OBIEEServers=` and enter the server URL. If you have multiple OBIEE servers, enter the URLs as a comma separated list.
3. Save the file.
4. Stop and restart the web server.

Loading OBIEE Report Definitions

Access the Load Report page. (Select PeopleTools, Portal, OBIEE Setup, Load Report Definition.)

Load Report

OBIEE Server: OBIEE Server1

Report Path:

[Root](#) > [USERS](#) > PTDMO

Folders		
Name	Owner	Last Modified Date
briefingbook	PTDMO	12/11/07 08:39:14Z
delivers	PTDMO	12/11/07 08:39:14Z
filters	PTDMO	12/11/07 08:39:14Z
ibots	PTDMO	12/11/07 08:39:14Z
portal	PTDMO	12/11/07 08:39:14Z

Reports					
Select	Loaded in PeopleSoft	Name	Owner	Last Modified Date	Last Loaded Date
<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	--

Report List for Deletion				
Select	Report Exist In OBIEE	Name	Report Path	Last Loaded Date
<input type="checkbox"/>	<input checked="" type="checkbox"/>	MARKET_REGION	/USERS/PTDMO	01/25/08 2:23AM

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 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 user ID of the owner as indicated in the OBIEE server.

Last Modified Date

This display-only field displays the date that the OBIEE report folder was last modified.

Reports

Select	Select to choose the OBIEE report to load into the PeopleSoft application.
Loaded in PeopleSoft	This display-only field indicates whether the report has been loaded into the PeopleSoft application.
Name	This display-only field indicates the name of the OBIEE report as entered in the OBIEE system.
Owner	This display-only field displays the user ID of the owner as indicated in the OBIEE server.
Last Modified Date	<p>This display-only field displays the date that the OBIEE report was last modified.</p> <p>This information is helpful in determining whether the version of the report loaded into the PeopleSoft application is the most current.</p>
Last Loaded Date	This display-only field displays the date that the OBIEE report was last imported into the PeopleSoft application.

Report List for Deletion

Select	Select to choose the OBIEE report to delete from the PeopleSoft application.
Report Exist In OBIEE	<p>This display-only field indicates whether the report has been deleted from the OBIEE server.</p> <p>Pagelets based on reports that you remove from the OBIEE server will <i>not</i> produce results in the pagelet.</p>
Name	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. Select the Select check box for the reports that you want to load into the PeopleSoft application.

Note. You can 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. Select the Select check box for the reports that you want to delete from the PeopleSoft application.

Note. You can 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. (Select 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

Version

Click the prompt to select the version of the OBIEE report to view.

Note. 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. (Select the Report Expression tab.)

OBIEE Server:	OBIEE Server1
Report Name:	MARKET_REGION
Report Path:	/USERS/PTDMO
Version:	2 *Most Current Version
Filter Expression:	Markets.Region is prompted

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 Oracle Business Intelligence documentation for explanations of the available types of views.

<http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/index.html>.

Access the Report References page. (Select 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

Save

Report References page

PeopleSoft Pagelet ID

This value is supplied by default from the *PTPPB_PAGELET_ID* content reference attribute value that you enter on the pagelet Content Ref Administration page.

User Developed Pagelet ID

Enter the pagelet ID of any user-developed pagelets that you want to track. This field is informational only and has no validation behind it.

Access the Xml Definition page. (Select 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

Report XML Definition

This field displays the XML tags and embedded data for the version of the report as it is currently loaded. Move 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:

Using Pagelet Wizard to Create OBIEE Report Pagelets

To create OBIEE report pagelets:

1. Load the OBIEE report into the PeopleSoft application.
2. Select PeopleTools, Portal, Pagelet Wizard, Pagelet Wizard.

Pagelet Wizard guides you through the process, by displaying a sequence of pages. Each page corresponds to a step in the process. A guide appears in a header above the page to indicate your progress.

3. Specify the pagelet information in step 1.

See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Step 1: Entering Pagelet Identifying Information, page 356.](#)

4. Specify *OBIEE* as the data type in step 2. 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 to Create and Manage Pagelets," Selecting an OBIEE Report Data Source, page 365.](#)

5. Specify the data source parameters.

See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Specifying OBIEE Data Source Parameters, page 381.](#)

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 to Create and Manage Pagelets," Step 5: Specifying Pagelet Display Options, page 395.](#)

8. Specify the publishing options.

See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Specifying Pagelet Publication Options, page 422.](#)

Part 4

Working with Related Information

Chapter 14

Configuring Interwindow Communication

Chapter 15

Developing and Configuring Related Content Services

Chapter 16

Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services

Chapter 17

Creating and Configuring WorkCenter Pages

Chapter 14

Configuring Interwindow Communication

This chapter presents an overview of interwindow communication (IWC) and discusses how to:

- Configure fields to publish.
- Configure fields to subscribe.

Understanding Interwindow Communication

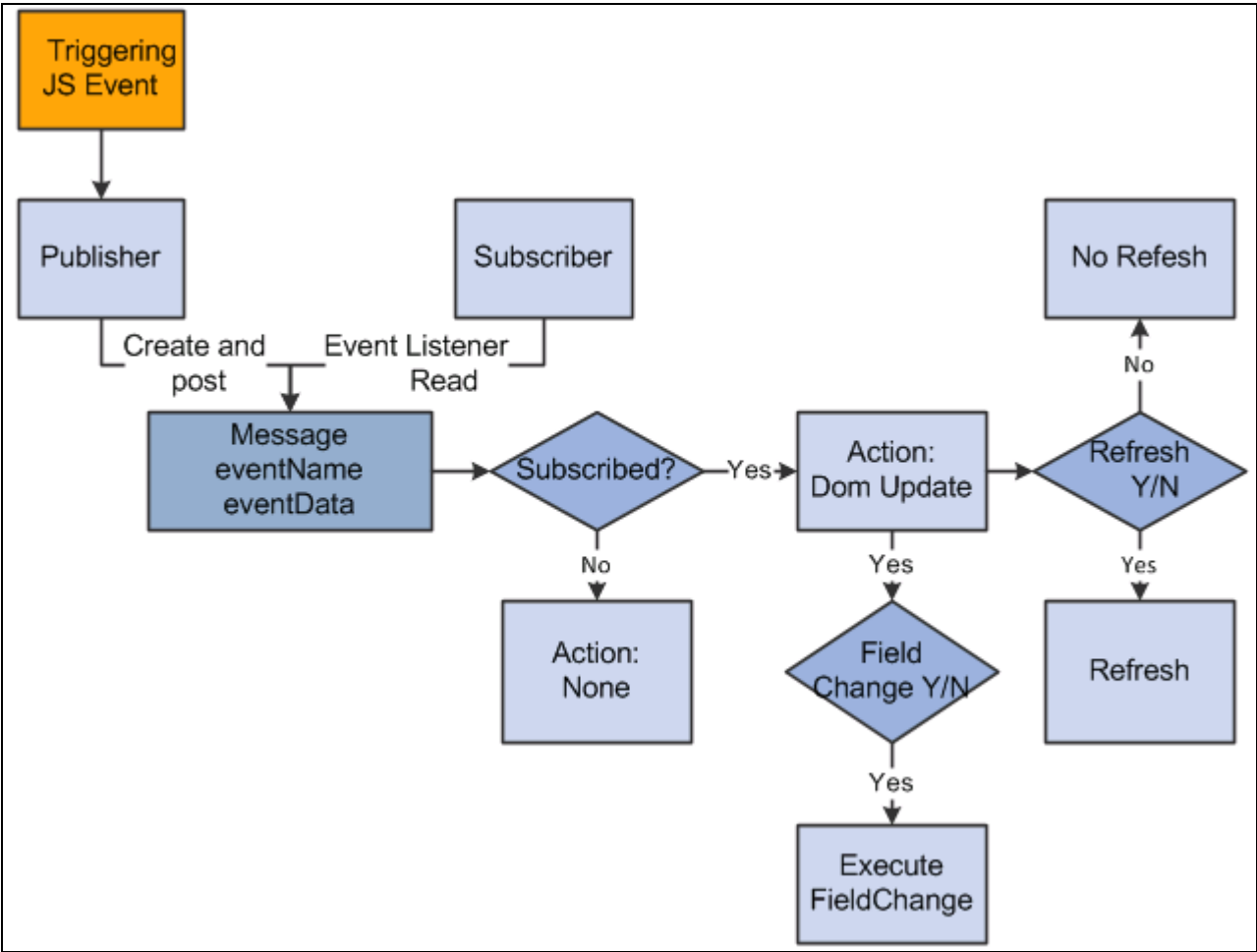
Interwindow communication (IWC) is a frame-to-frame messaging mechanism that enables browser window objects—such as pages or pagelets—to be aware of and react to field level data changes that occur in other browser window objects that share the same browser window. Interwindow communication enables you to publish field-level changes as they occur so that any subscribing page or pagelet becomes instantly aware of the change and then, can act on it in real-time.

Here is a simple example. On a dashboard you have the Employees pagelet, which contains a list of names. Also on the dashboard you have an Employee Projects pagelet, which contains project listings for an employee. Interwindow communication enables you to click a name in the Employee pagelet and have the Employee Projects pagelet retrieve the project listing for the corresponding name.

Note. Interwindow communication works at level-0 only. Pages or pagelets must be completely loaded or the process fails.

Process Flow

This diagram presents the flow of events and actions that comprise the publish and subscribe process of interwindow communication.



IWC process flow

Step	Description
1. Triggering event occurs.	User clicks a button or link or changes the value of a field that is configured to publish.
2. IWC generates message.	Simple string formatted message temporarily stored to IWC-related table.
3. Pagelet reads message.	Listener checks to see if loaded pagelets subscribe to the message.
4. Determine whether update action is necessary.	Check the document object model of the pagelet and run a field change event if necessary.
5. Determine whether refresh is necessary.	If configured to refresh, page refreshes partially or with new parameters.

Interwindow communication does not depend on any template, frame, or iframe and works seamlessly throughout PeopleSoft applications in almost any scenario for which you require pagelet to pagelet interaction. You can configure IWC to operate between pagelets and also within the same pagelet, if necessary. Although IWC is ideal for use with related content and WorkCenter pages, IWC is independent of both technologies and IWC does not depend on PeopleSoft Integration Broker.

Prerequisites

This table lists the prerequisite tasks that you must complete before setting up interwindow communication.

<i>Prerequisite</i>	<i>Cross-Reference</i>
Create pagelets	See Chapter 12, "Developing and Registering Pagelets," page 443.
Register pagelet in the current portal system.	See Chapter 12, "Developing and Registering Pagelets," Registering Homepage and Template Pagelets, page 500.

Common Elements Used in This Chapter

Event Name	Enter any unique value that identifies the event for this page or pagelet.
HTML Field Name	Enter the name of the field as it appears in the HTML source code.
Message Data	Enter the names of the fields that you want to publish. Separate multiple field names with commas.
Partial page refresh	Select to update a section of the page or pagelet bypassing a full server trip and limiting code and markup changes. This option eliminates the page flicker associated with refreshing.
Refresh with parameters	Select to update a page or pagelet while passing the necessary parameters to complete the refresh and perform any required actions.

Page and Pagelet Types

This table describes the types of pages and pagelets that IWC supports and the type of refresh that is available to each page and pagelet:

<i>Page or Pagelet Type</i>	<i>Description</i>	<i>Refresh Support</i>
Transaction page	The standard PeopleSoft page that appears in the target content area.	<ul style="list-style-type: none"> Partial page refresh Refresh with new parameters

Page or Pagelet Type	Description	Refresh Support
Component pagelet	The more frequently used pagelet built by application developers using PeopleSoft Application Designer.	<ul style="list-style-type: none"> • Partial page refresh • Refresh with new parameters
iScript pagelet	The PeopleCode-based pagelets built by using Pagelet Wizard or by writing HTML.	Refresh with new parameters

See Also

[Working with Homepages, Dashboard Pages, and Pagelets](#)

[Chapter 15, "Developing and Configuring Related Content Services," page 533](#)

Events and Messages

Interwindow communication involves publishing and subscribing to messages.

Publishing

The first part of interwindow communication is to trigger the publication of a message. Publishing can be initiated by these JavaScript events that occur in the browser:

- **OnChange**

The user changes a value in a field on a transaction page or pagelet.

- **OnClick**

The user clicks a button or link on a transaction page or pagelet.

Note. The link must have a unique identifier and be searchable in the document object model (DOM).

Subscribing

The second part of interwindow communication is to subscribe to the message that other window sends. The application calls an event listener in the **OnLoad** event of every subscribing page in the template. This event listener runs the callback function from the subscribing page or pagelet using the data that is received from the publishing page. If the page or pagelet subscribes to the event, then the receiving window acts on the message.

After a message is published, each pagelet checks to determine whether it subscribes to this message. If the page or pagelet subscribes to the message, then it can refresh the page as configuration determines.

Configuring Interwindow Communication Publish and Subscribe Options

This section discusses how to:

- Configure fields to publish.
- Configure fields to subscribe.

Pages Used to Configure Interwindow Communication Publish and Subscribe Options

Page Name	Definition Name	Navigation	Usage
Content Reference Message Events	IWC_CREF_EVENTS	PeopleTools, Portal, InterWindow Communication	Configure fields to publish. Configure fields to subscribe.

Configuring Fields to Publish

Access the Content Reference Message Events page. (Select PeopleTools, Portal, InterWindow Communication.)

Content Reference Message Events

Portal Name **EMPLOYEE**

Portal Object Name **ADMN_SD_USER_HMPG**

Event Name	Message Event Type	Field Event Type	HTML Field Name	Message Data
1	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Personalize | Find | First 1 of 1 Last

Content Reference Message Events page

1. Enter a value in the Event Name field.
2. Select *Pub* as the message event type.
3. Select either *Change* or *Click* as the field event type.
4. Enter the field name as it appears in the HTML source.
5. Enter the names of the fields that you want to publish.
6. Save the page.

Configuring Fields to Subscribe

Access the Content Reference Message Events page. (Select PeopleTools, Portal, InterWindow Communication.)

Content Reference Message Events

Portal Name EMPLOYEE

Portal Object Name ADMIN_SD_USER_HMPG

Personalize | Find | First 1 of 1 Last

	Event Name	Message Event Type	Field Event Type	HTML Field Name	Message Data		
1	<input type="text"/>	<div></div>	<div></div>	<input type="text"/>	<input type="text"/>	<div></div>	<div></div>

Content Reference Message Events page

1. Enter a value in the Event Name field.
2. Select *Sub* as the message event type.
3. Select either *Full Refresh* or *Partial Refresh* as the field event type.
4. Enter the name of the subscribing field as it appears in the HTML source.
5. Enter the names of the fields that you want to publish.
6. Save the page.

Chapter 15

Developing and Configuring Related Content Services

This chapter provides an overview of the PeopleSoft Related Content Framework and discusses how to:

- Define related content services.
- Assign and manage related content services.
- Configure related content service layout.

Understanding the PeopleSoft Related Content Framework

This section discusses:

- Framework flexibility.
- The related content development process.
- Delivered related content services for PeopleSoft Application Portal.
- Delivered related content services for Oracle WebCenter.

The PeopleSoft Related Content Framework enables developers and subject matter experts to link application pages with contextually relevant collaborative content.

The primary goal of related content services is to create opportunities for the user to view or quickly access information that supports, extends, and augments any PeopleSoft transaction. 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, and any relevant non-PeopleSoft data into a single tool—PeopleSoft application pages.

Related content services can be any collaborative, analytical, or informational content that is useful for completing business tasks. Related content services include any content such as related discussion forums, wiki pages, or links; PeopleSoft content references; or pagelets that are offered as services to be consumed by the applications. Related content services can be configured within the same database, among multiple PeopleSoft databases, and between a PeopleSoft database and external sources.

Prerequisites

This table lists the necessary and optional prerequisites to defining and configuring related content services:

Task	Reference
**Enter an authentication domain in the web profile.	See Chapter 9, "Configuring the Portal Environment," Configuring General Portal Properties , page 242.
Configure single signon.	See <i>PeopleTools 8.52: Security Administration</i> , "Implementing Single Signon."
Configure Security Authorization service.	See <i>PeopleTools 8.52: Security Administration</i> , "Using Web Services for Object and Row-Level Data Authorization."

**Indicates the prerequisite is required.

Common Elements Used in This Chapter

Assign Related Content to Application Pages

Click to access the Select a Content Reference page. Use this link to create a new related content assignment.

Configure

Click to access the Configure Service page, where you map field values to the service parameters.

Content Reference

This is the content reference to which you are assigning the related content. You select the content reference on the Select a Content Reference page. This field is display-only.

Create a New Related Content Service

Click to access the Define Related Content Service page.

Description

Enter a description for the related content service.

Display in field menu

Select to enable access to a related content field-level contextual pop-up menu in the field to which you are mapping the service parameter. This field is enabled only when the mapping type is *Page Field*.

Selecting this check box causes the red related content glyph to appear in the field at runtime, even if the field is display-only.

Important! When mapping a service parameter to a page field, always examine the page to determine whether the selected page field lies within a collapsible page control such as a group box, grid, and scroll area. Services that access page fields that lie within a collapsible control that is set to the collapsed state by default must be configured to use the Display in field menu contextual menu option only.

Note. If you select a hidden field, the menu does not appear at runtime.

Edit Configuration

Click to review or change the existing related content assignment. This button accesses the Assign Related Content page.

Is Value Required?	Some service might accept valid null values being passed to its parameters. If you select the Refresh Service On Change and Is Value Required? check boxes for this parameter, then when the mapped field value of that parameter on the transaction page changes to a null value, the system does not refresh the related content frame.
Mapping Data	Enter the constant if the mapping type is <i>Fixed Value</i> . For all other mapping types, this field is disabled and displays the name of the field or variable that you map to the service parameter.
Mapping Type	Select from four possible mapping types to map the service parameter to: <i>Fixed Value:</i> Select to map the service parameter to a constant. Selecting this option enables the Mapping Data field. <i>Key Field:</i> Select to map the service parameter to one of the level-0 search keys of the component assigned to the content service. <hr/> Note. For related actions, the key name used for the related actions component is the field name that you enter here. The key value comes from the corresponding key field from the detail component key list. <hr/> <i>Page Field:</i> Select to map the service parameter to any level-0, level-1, or level-2 page field, including hidden fields. This option is available for page-level services only. <i>System Variable:</i> select to map the service parameter to one of the available system variables.
Page	Select the specific page in the component to which you are assigning related content. This field appears in the grid when you select the Services at Page level option.
Page Name	Displays the page label. This field is display-only. This field appears in the grid when you select the Services at Page level option.
Portal Name	This is the name of the current portal where the related content will be assigned. This field is display-only.
Read help text	Click to display the help text as defined on the Related Content Service Definition page. <u>See Chapter 15, "Developing and Configuring Related Content Services," Writing Help Text for Related Content Service Definitions, page 566.</u>

Refresh Service On Change	<p>Select to have the service data refresh automatically when the key values on the transaction page changes. The default value is deselected.</p> <hr/> <p>Important! Use this option judiciously. If the service is linked to a level 1 or level 2 key field and the application user scrolls through data rapidly, performance can be affected.</p> <hr/> <p>Note. If the page or the page field to which a service is mapped is in <i>deferred processing mode</i>, a trip to the server is required to refresh related content.</p> <hr/>
Remove Configuration	<p>Click to delete the existing related content assignment from the content reference.</p>
Required	<p>Specifies whether the service parameter is defined as <i>Required</i> in the service definition on the Related Content Service Definition page in the Service URL Parameter grid.</p> <p>See Chapter 15, "Developing and Configuring Related Content Services," Service URL Parameters, page 554.</p>
Select	<p>Click to access the lookup page associated with the corresponding service type. The lookup pages enable you to choose the content reference, pagelet, or service that you want to assign as related content. This button is enabled only after you select the service type.</p>
Service Parameter	<p>Displays the list of parameters that the service requires to form the contextual relationship to the primary transaction page.</p> <p>See <i>PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide</i>, "Creating Component Definitions," Understanding Search Records.</p> <p>See Chapter 15, "Developing and Configuring Related Content Services," Content Reference Attributes as Service URL Parameters, page 549.</p>
Service ID	<p>Specifies the unique identifier of the service definition. For content references and pagelets, the service ID corresponds to the Name field on the Content Reference Administration page.</p>
Service Label	<p>Enter the label that appears in the contextual menu, on the tab when the service appears in the bottom Related Content frame, and in the pagelet header when the service appears in the right Related Content frame. The content reference name or service definition name appears as the service label by default when you select the service; you can change the default value.</p>

Service Target

Select the target location where the related action appears. Select from these options:

Modal Window - select to have the related content appear in a modal window that hovers in front of the transaction page.

New Window - select to have the related content appear in a new browser window.

Replace Window - select to have the related content refresh the entire browser window, including navigation.

Target Content - select to have the related content appear in the target area, replacing the current page.

Service Type

Select from these three options.

Content Reference - select this option to assign an existing content reference as related content. Because content references are anonymous services, you do not have to configure the content reference as a service first *unless you want to specify a particular page to open when the related content is displayed*. The system creates the service definition for you automatically when you save the assignment.

Pagelet - select this option to assign an existing locally registered pagelet as related content. Because pagelets are anonymous services, you do not have to configure the pagelet as a service first. The system creates the service definition for you automatically when you save the assignment.

Service - select this option to assign an existing service that you configured using the Define Related Content Service page.

Note. After you select a service type, the system enables the Select button.

Framework Flexibility

The PeopleSoft Related Content Framework is designed to be flexible. You can assign, configure, and maintain related content services without access to developer tools such as PeopleSoft Application Designer. The related content service pages provide a variety of configuration options, which enable you to define related content services, associate them with components and pages, and grant access to these services based on your business rules.

PeopleSoft applications divide related content services into two major categories—related content and related actions—based on where the content appears when the user accesses the service.

- Related content only appears in one of two Related Content frames, which appear at the bottom or to the right of the primary transaction page.
- Related actions appear outside of the Related Content frames. Target locations include modal windows, in new browser windows, or in the existing browser window replacing the current transaction page.

The Bottom Related Content Frame

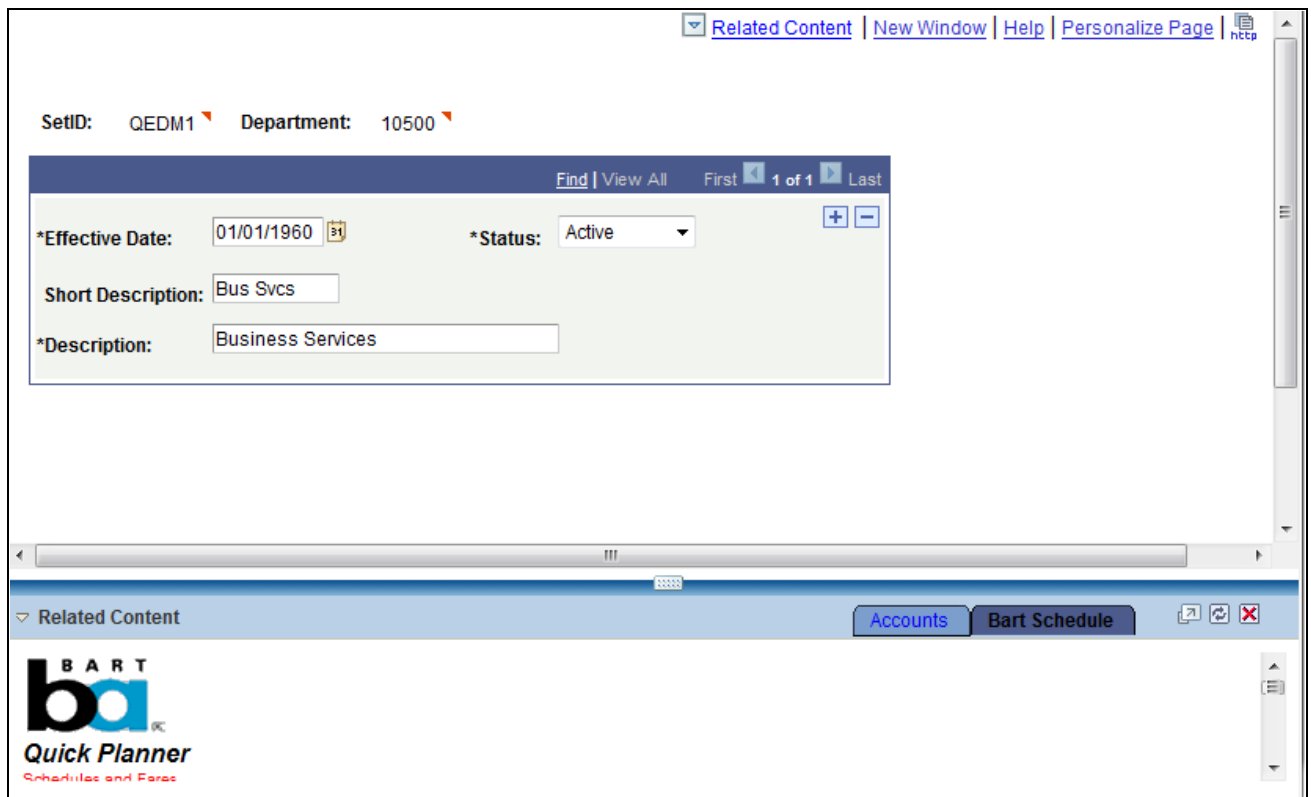
Note these points about the bottom Related Content frame:

- The frame extends across the width of the transaction page.
- The visible area increases and decreases when you drag the handle on the upper edge of the frame in a vertical direction.
- The frame appears and disappears when you click the Expand/Collapse button on the left side of the frame header.
- The frame reloads when you click the Refresh button in the frame header .
- The frame loads the current active service in a new window when you click the Zoom icon.
- The frame closes when you click the Close button.
- Open the active service in another window by clicking the zoom button in the frame header.

Note these points about the *services* that appear in the bottom frame:

- Only one service can be active and have focus at a time.
- Multiple services appear as tabs across the top of the frame.
- Users click the tab to bring the service into focus.

This is a sample transaction page showing related content in the bottom Related Content frame:



Transaction page showing the bottom Related Content frame

The Right Related Content Frame

Note these points about the right Related Content frame:

- The frame extends down the length of the transaction page.
- The visible area increases and decreases when you drag the left edge of the frame in a horizontal direction.
- The frame appears and disappears when you click the Maximize and Minimize buttons on the left border of the frame.
- The frame divides the visible area equally among all open pagelets.
- The bottom of the frame displays a More menu, which you use to access minimized pagelets.

Note these points about the *services* that appear in the right Related Content frame:

- Each service exists as a self-contained pagelet inside the Related Content frame.
- Any number of services can open be active and in the expanded state simultaneously.
- All maximized service pagelets divide the available space equally and always share the same vertical and horizontal dimensions. As users maximize and minimize other service pagelets in the frame, any service pagelets that remain open resize automatically so that the space appears equally divided.

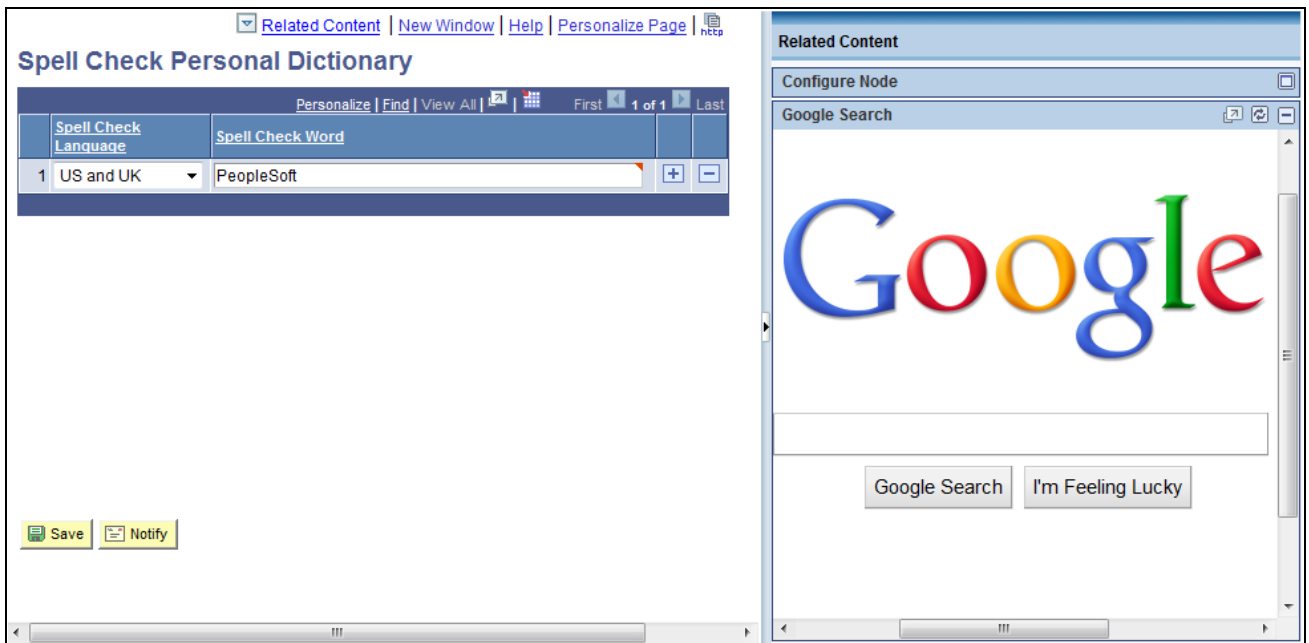
Note. Users cannot resize individual service pagelets.

- Each pagelet header can contain:
 - A Zoom button that users click to display the service in a new window.
 - A Refresh button that users click to reload the service.
 - A Maximize/Minimize button that users click to toggle the service between the expanded and collapsed states.

Note. The state of each pagelet persists based on the state from the user's last session.

- Scroll bars appear on the right and bottom of a service pagelet if it is too small to fully display the content.

This is a sample transaction page showing related content in the right Related Content frame:

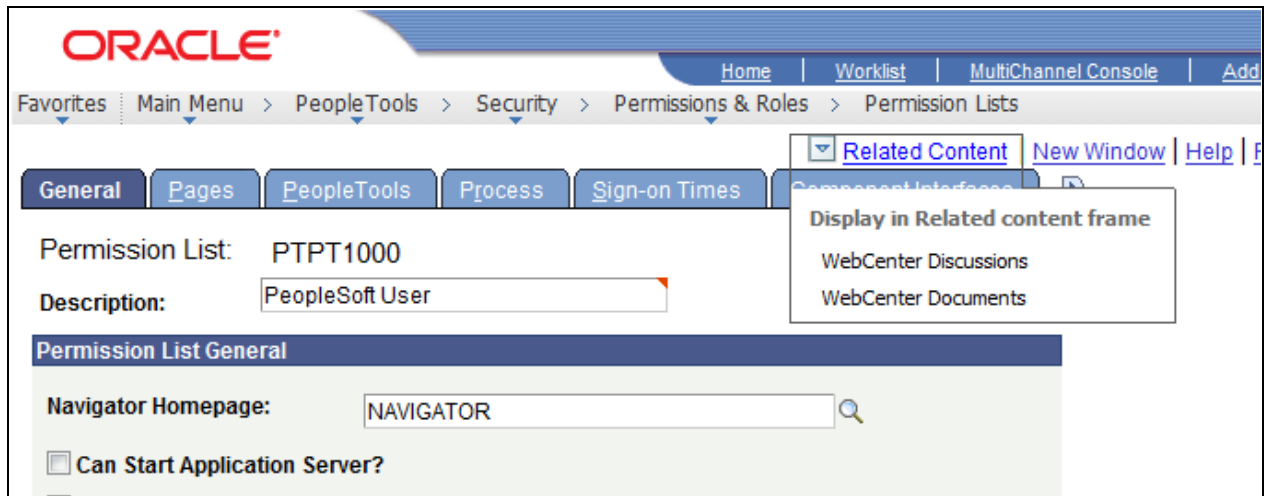


Transaction page showing the right Related Content frame with two services

Related Content Service Menus

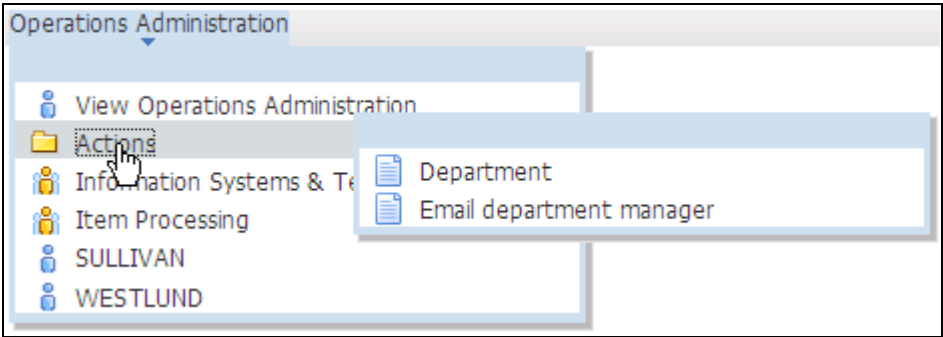
Menus provide access to related content services. Three types of menus exist:

- The Related Content menu, which appears in the page bar at the top of the transaction page.



Example: Related Content menu showing two services

- Related actions menus, which can appear:
 - As Actions folders in drop-down SmartNavigation menus,



Related actions in a sample drop-down SmartNavigation menu

- As Actions links on chart nodes of SmartNavigation pages



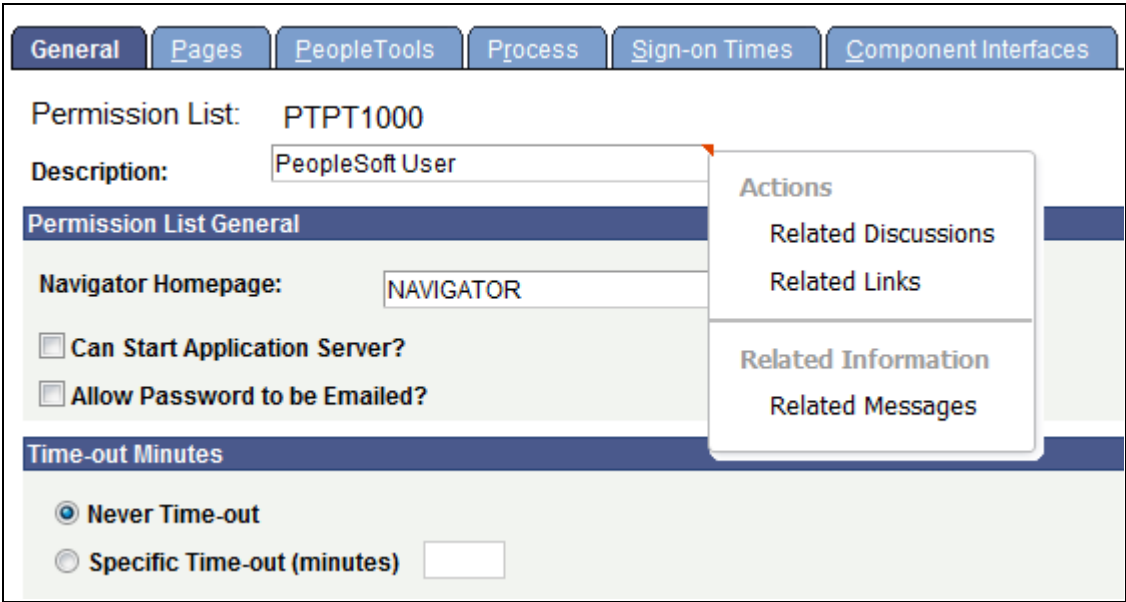
Related actions on a sample SmartNavigation page

- As Related Actions links in the search results of a Keyword search.



Related actions on a sample Keyword Search Results page

- Page field contextual menus, which appear when the user right-clicks the red page-field glyph.



Example: Page field contextual menu showing related content services in the Actions and Related Information menus

The Related Content Development Process

This table lists the steps in the process flow, the roles involved during each step, and provides cross-references to supporting PeopleBooks documentation:

Step	Role	Reference
Design related content integration.	Subject matter expert Application user	See Chapter 14, "Configuring Interwindow Communication," page 527. See Chapter 12, "Developing and Registering Pagelets," page 443.
Define related content service	Developer	See Chapter 15, "Developing and Configuring Related Content Services," Defining Related Content Services, page 546.
Test related content service	Developer	See Chapter 15, "Developing and Configuring Related Content Services," Testing Related Content Service Definitions, page 566.
Enter help text	Developer	See Chapter 15, "Developing and Configuring Related Content Services," Writing Help Text for Related Content Service Definitions, page 566.
Assign related content service to content reference	Subject matter expert	See Chapter 15, "Developing and Configuring Related Content Services," Selecting a Content Reference, page 578.
Configure service.	Subject matter expert	See Chapter 15, "Developing and Configuring Related Content Services," Configuring Service Parameters, page 588.
Configure layout.	Subject matter expert	See Chapter 15, "Developing and Configuring Related Content Services," Configuring Menu and Frame Layout, page 589.
Access transaction and view related information	Application user	See <i>PeopleTools 8.52: PeopleSoft Applications User's Guide</i> , "Using PeopleSoft Application Pages," Accessing Related Content and Actions.

Delivered Related Content Services for PeopleSoft Application Portal

PeopleTools provides the service definitions for three related content services for PeopleSoft Application Portal:

- Related Discussion service
- Related Links service
- Related Tags service

These related content services provide features of Oracle's PeopleSoft Application 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 Application Portal database is required to provide the services and to store the related data.

Setup procedures for these services is covered in the PeopleSoft Application Portal PeopleBooks.

See *PeopleSoft Application 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 Application 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 Application Portal PeopleBooks.

See *PeopleSoft Application Portal 9.1 PeopleBook: Using Portal Features*, "Working With the Related Discussion Service."

Related Links Service

The Related Links related content service enables 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 Application Portal—specifically, blogs, discussions, or collaborative workspaces—or external URLs. In addition, the Related Links service enables 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 Application Portal PeopleBooks.

See *PeopleSoft Application Portal 9.1 PeopleBook: Using Portal Features*, "Working With the Related Links Service."

Related Tags Service

The Related Tags related content service enables 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 Application Portal PeopleBooks.

See *PeopleSoft Application Portal 9.1 PeopleBook: Using Portal Features*, "Working With the Related Tags Service."

Delivered Related Content Services for Oracle WebCenter

PeopleTools provides these service definitions for use with Oracle WebCenter:

- WebCenter Blogs service
- WebCenter Discussions service
- WebCenter Documents service
- WebCenter Links service
- WebCenter Wikis service

These related content services provide features of Oracle WebCenter that can be added to PeopleSoft application pages through the PeopleTools Related Content Framework. While the service definitions are provided in PeopleTools, an active Oracle WebCenter database is required to provide the services and to store the related data.

See *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter*, http://download.oracle.com/docs/cd/E21764_01/webcenter.1111/e12405/wcadm_getstarted_app.htm.

Note. Oracle recommends that you have a WebCenter that you dedicate solely to your PeopleSoft applications integration.

WebCenter Blogs Service

Blogs provide a useful tool for discussing or evangelizing any type of idea, strategy, or point of view. Blogs may be projected out to a select group of people or to a wider audience. Typically, blogs invite readers to comment on the overall concepts.

The Blogs service enables users to create instance specific or shared WebCenter blogs, which appear for all transaction instances of the transaction page.

See Chapter 16, "Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services," Oracle WebCenter Blogs Service, page 601.

WebCenter Discussion Service

The Discussions service lets you expose discussion forums on your application pages, so users can create forums, post questions, and search for answers. For example, customers can share product reviews, or a customer service department can answer questions online. Discussion forums additionally provide the means to preserve and revisit discussions.

The Discussion service enables users to create instance specific or shared WebCenter discussions, which appears for all transaction instances of the transaction page.

See Chapter 16, "Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services," Oracle WebCenter Discussion Service, page 601.

WebCenter Documents Service

The Documents service provides features for accessing, adding, and managing folders and files; configuring file and folder properties; and searching file and folder content in the connected content repositories. When you set up a content repository connection to integrate the Documents service, you can add content to the transaction pages of the PeopleSoft application.

The Documents service enables users to create instance specific or shared WebCenter documents, which appears for all transaction instances of the transaction page.

See [Chapter 16, "Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services," Oracle WebCenter Documents Service, page 601.](#)

WebCenter Links Service

The Links service provides a way to view, access, and associate related information. For example, in a list of project assignments, you can link to the specifications relevant to each assignment. In a discussion thread about a problem with a particular task, you can link to a document that provides a detailed description of how to perform that task.

The Links service enables users to create instance specific links or shared links, which appears for all transaction instances of the transaction page. When the transaction page or instance also uses other WebCenter related content services, related links to the other WebCenter group spaces appear by default and cannot be removed.

Unlike other WebCenter related content services, the Links service is a PeopleSoft component-based page. When you click the links in the service, the target opens in a new browser window.

See [Chapter 16, "Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services," Oracle WebCenter Links Service, page 602.](#)

WebCenter Wiki Service

A wiki is a collection of useful content or information that users can browse, update, and remove, sometimes without the need for registration. This ease of interaction and the variety of operations make wiki an effective tool for collaborative authoring, where multiple people create written content together.

The wiki service enables users to create instance specific wiki pages or shared WebCenter wiki pages, which appears for all transaction instances of the transaction page.

See [Chapter 16, "Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services," Oracle WebCenter Wiki Service, page 602.](#)

See Also

[Chapter 16, "Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services," Understanding the Oracle WebCenter Integration, page 599](#)

Defining Related Content Services

This section provides overviews of service types, service URL parameters, service security, and discusses how to:

- Define application class-related content services.
- Define non-PeopleSoft-related content services.
- Define PeopleSoft component related content services.

- Define Embeddable Pagelet related content services.
- Define OBIEE Pagelet related content services.
- Define PeopleSoft Query related content services.
- Define PeopleSoft Script related content services.
- Test related content service definitions.
- Write help text for related content service definitions.
- Copy related content service definitions.

Understanding Service Types

The PeopleSoft Related Content Framework supports seven explicitly definable service types. Each type is defined by its URL construction and each type requires different information to form the complete URL that locates and requests the contextual data.

When you select a service URL type, various service parameters appear for you to configure. The service parameters differ based on the URL type that you select. For example, when you select *Embeddable Pagelet*, the Node and Pagelet Name fields appear as service parameters because the framework must know the name of the node where the pagelet resides and the name of the pagelet to construct the URL and retrieve the pagelet and its data.

In addition to system-defined service parameters, you can enter user-defined parameters, which offers maximum flexibility in configuring the service to meet business requirements.

Definable Service Types

You can explicitly define the following related content service types:

Application Class

This service type accesses any service that is defined by means of PeopleCode application classes. Application class services can:

- Access data in the component buffer.
- Run any PeopleCode program that is valid in the FieldChange event, for example, transferring the user to another page, opening a modal window, or modifying data in the component buffer.
- Be associated only with the Related Actions menu for use on SmartNavigation pages and cannot be targeted to the related content frame because the target frame and the related content frame maintain separate sessions and separate buffer structures.

Non-PeopleSoft

This service type accesses any non-PeopleSoft pages, external content, or services.

Note. Use this service type for PeopleSoft User Productivity Kit (UPK) related content.

See [Appendix F, "Configuring UPK Topics as Related Content," page 725](#).

PeopleSoft Component

This service type accesses any PeopleSoft components. The component can be from the local PeopleSoft system or from a remote PeopleSoft system. The node name specifies whether the service resides in the local or remote PeopleSoft system.

The framework enables you to assign most PeopleSoft components as anonymous services without first creating a service definition. However, you must create a PeopleSoft component service definition when:

- You want to specify the page that opens when the service is accessed.
- You want to pass parameters to the component other than the key structure.
- You want to assign a component as a related action on SmartNavigation pages; only components that you specifically define as services appear in the Actions drop-down menu for SmartNavigation.

Embeddable Pagelet

This service type accesses Pagelet Wizard pagelets. The pagelet can be of any pagelet data source but it should be published as an *embeddable* pagelet.

OBIEE (Oracle Business Intelligence Enterprise Edition) Pagelet

This service type accesses analytics-based Oracle Business Intelligence Enterprise Edition (OBIEE) pagelets. You use Pagelet Wizard to create and publish embeddable OBIEE pagelets that use OBIEE reports as the data source. OBIEE reports are analytics reports that are consumed in the PeopleSoft system.

PeopleSoft Query

This service type accesses PeopleSoft Query definitions. The query results can also be filtered based on the query definition and the parameters that it requires.

PeopleSoft Script

This service type accesses iScript-based services. 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.

The information that the system uses to construct the seven URL types differs for each type and is described in the following sections.

Anonymous Service Types

Services that you do not have to define explicitly are called anonymous services. Two service types exist that you do not have to explicitly define:

- Content references.
- Pagelets.

When you configure an anonymous related content service, the system automatically defines the service and the service parameters. No explicit service definition configuration is required.

Understanding Service URL Parameters

A service might need additional parameters to obtain the context for the data that the service provides. You define these parameters in the Service URL parameter grid.

When you access the service from the transaction page, contextual values from fields in the buffer, system variables, or even fixed values that you map to the parameters pass as arguments to the service.

Content Reference Attributes as Service URL Parameters

When you assign a content reference as an anonymous service, the system automatically defines service parameters for the components that provide either a valid search record or contextual record. For other types of content references, such as iScripts, PSQuery, and so on, the system checks for the RCF_SERVICE_PARAMETER attribute, which you must define manually on the Content Reference Administration page. If this attribute exists, the system automatically uses the attribute values as the service parameters for the content reference.

Note these points about the RCF_SERVICE_PARAMETER attribute.

- The attribute overrides the default behavior of component-based URLs. The system will not retrieve the service parameters from a valid search record or contextual record, rather the system supplants any search record parameters with the parameters included in the Attribute Value field.

- The attribute value format should follow this pattern:

NAME,Label;NAME,Label for example, *EMPLID,Employee ID;EXPENSEID,Expense ID*.

- The NAME attribute value should *not* include spaces.

This is a sample content reference that uses the RCF_SERVICE_PARAMETER attribute:

The screenshot shows a web-based form titled "Content Reference Attributes". It contains three input fields: "Name" (containing "RCF_SERVICE_PARAMETER"), "Label" (empty), and "Attribute value" (containing "EMPLID,Employee ID;EXPENSEID,Expense ID"). A "Delete" button is positioned to the right of the "Name" field. A "Translate" checkbox is located to the right of the "Label" field. An "Add" button is located at the bottom left of the form area.

Content Reference Administration page: Content Reference Attributes group box

See Also

Chapter 5, "Administering Portals," Content Reference Attributes, page 150

Understanding Service Security

PeopleSoft Related Content Framework uses the basic PeopleTools security model, which secures local and remote PeopleSoft transaction pages and the application data that underlies these pages based on the user's assigned permission lists and roles.

For services that provide access to non-PeopleSoft content, the framework enables you to use the security model of the content provider. You should consult the content provider and your security administrator to determine the best course of action.

PeopleSoft Security Authorization Web Service

The security authorization web service runs on PeopleSoft and certain third party systems such as Oracle WebCenter. The service examines the basic PeopleTools security on the providing system and can respond to a web service call from another PeopleSoft system.

PeopleSoft Related Content Framework uses the authorization service in these two ways

- To show and hide related content links that access PeopleSoft content.
- To grant access to third party systems, such as Oracle WebCenter that contain PeopleSoft content.

The authorization service can check for basic PeopleTools security at the definition level and at the data, or row, level.

At the definition level, the authorization service response determines whether related content or related actions links to PeopleSoft content appear for the user. This level of authorization is automatically engaged for any PeopleSoft content. Beyond basic PeopleTools security setup, no other configuration is required.

If the user has the appropriate permissions to view the content, then the authorization service determines whether data, or row-level security has been configured. If any such configurations exist, then the appropriate data is returned when the user clicks the related content or related action link to access the data.

Important! Row-level, or data permission, security is *not* automatic. Use the Authorization Configuration component to configure the authorization service to implement data-level security on remote systems. Setup procedures for Authorization service configuration is covered in another PeopleBook.

See *PeopleTools 8.52: Security Administration*, "Using Web Services for Object and Row-Level Data Authorization," Understanding Developing and Invoking the Security Authorization Service.

Service Definition Security Options

You specify the security options for a service definition at design time. The selection of a security options does not enforce security, rather the selection indicates to the consumer where to look for the security associated with the service definition. The security options that appear on the service definition are not mutually exclusive. You can select one or more options to make available to the person who assigns the service to a content reference. During the service configuration process, he or she selects the security option that the service implements. Before you select a security option for a service, you should understand its security model. If you select a security option for a service definition that isn't applicable to the service definition type, unexpected results can occur.

Important! The security authorization web service is called for all PeopleSoft content regardless of the option that you select; however, the service is not called for nonPeopleSoft content.

PeopleTools offers three security options.

- Public access.
- Related content provider security.
- Related content consumer security.

Public Access

The Public Access security option provides access to the service based on the permissions of the public user that you specify in the web profile. The Public Access option is intended for use with services that require few parameters, such as a generic help policy or an internet search engine.

Components that use basic PeopleTools security, which provides authorization and access through permission lists and roles, are not public. If you select the Public Access option for a PeopleSoft Component service definition, users must have access to any PeopleSoft component associated with the service.

See [Chapter 9, "Configuring the Portal Environment," Public Users, page 252.](#)

See *PeopleTools 8.52: Security Administration*, "Setting Up Permission Lists," Setting Page Permissions.

Related Content Provider Security

Some content providers offer complex and industry-specific security models and mechanisms that are native to the service. The Related Content Provider Security option is intended for use with services that implement security that is specific to their data model or the content that they provide.

As an example, consider the Employee component in an HCM environment as the consumer and the Compensation Details component in Payroll as the producer. If you select Related Content Provider Security, then the provider system (Employee/HCM) must provide the security.

Related Content Consumer Security

PeopleSoft applications use basic PeopleTools security, which provides authorization and access to the many elements of your PeopleSoft application through the assignment of permission lists and roles to users. The Related Content Consumer Security option is intended for use with services that rely on the basic PeopleTools security or row-level security for authorizing access to the service content, for example:

- PeopleSoft Component, Embeddable Pagelet, OBIEE Pagelet, PeopleSoft Query, and PeopleSoft iScript services that access remote PeopleSoft nodes.
- Application Portal services that create related discussions, tags, and links workspaces
- Oracle WebCenter services that create group spaces linked to PeopleSoft transaction pages. The WebCenter maintains the group space security, but authorization and access are based on the permissions of the PeopleSoft transaction page.

For example, consider the Employee component in HCM as the consumer and the Compensation Details component in Payroll as the producer. If you select Related Content Consumer Security, then the consumer system (Employee/HCM) must provide the security. Therefore, the provider system (Compensation/Payroll) creates an authorization request to the consumer (Employee/HCM), which responds with the authorization details for that user.

See Also

PeopleTools 8.52: Security Administration, "Setting Up Permission Lists"

PeopleTools 8.52: Security Administration, "Setting Up Roles"

PeopleTools 8.52: Security Administration, "Administering User Profiles"

[Chapter 16, "Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services," Understanding Security Synchronization, page 603](#)

PeopleTools 8.52: Security Administration, "Using Web Services for Object and Row-Level Data Authorization," Understanding Developing and Invoking the Security Authorization Service

Common Elements Used in This Section

This section describes the common elements used in defining related content services. The elements appear in the lists in the order that they appear on the respective pages

Related Content Service

This table describes the fields on the Define Related Content Service page that are common to all service URL types. .

URL Type

Select from seven service types that build the URL that defines the location and parameters of the service:

Application Class

Non-PeopleSoft URL

PeopleSoft Component

Embeddable Pagelet

OBIEE Pagelet

PeopleSoft Query

PeopleSoft Script

See [Chapter 15, "Developing and Configuring Related Content Services," Understanding Service Types, page 547.](#)

Write help text

Click to open a help text page where developers can enter instructions that are specific to assigning this related content service and mapping service parameters. A subject matter expert can access the help text for assistance when configuring a related content assignment.

See [Chapter 15, "Developing and Configuring Related Content Services," Selecting a Content Reference, page 578.](#)

Copy Service Definition

Click to clone the service definition and its parameters.

See [Chapter 15, "Developing and Configuring Related Content Services," Writing Help Text for Related Content Service Definitions, page 566.](#)

URL Information

This table describes URL information fields. These fields are common to several or all service URL types.

Node Name

Enter the node name of the PeopleSoft system where the service resides.

If the service resides in the same PeopleSoft system, the node name will be either *LOCAL_NODE* or any default local node of the PeopleSoft system.

Populate parameters

Supplies the service parameters in the Service URL parameter grid based on the URL type. This button appears for embeddable pagelet, PeopleSoft OBIEE pagelet, and PeopleSoft Query service types only.

Note. For embeddable and OBIEE pagelets, only context-sensitive parameters are supplied. For Query pagelets, only prompt fields are populated.

See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Step 3: Specifying Pagelet Data Source Parameters, page 372.](#)

Post mapping definition data

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 deselected by default.

Escape URL Parameters	Select to have the parameters defined in the Service URL Parameter grid escaped when the application forms the service URL. The default is selected.
Show Formed URL	Click to display the entire formed URL with test parameter values in a display-only edit box beneath the Service URL Parameter grid. <hr/> Note. Application class URLs cannot be formed by means of this link because the URL is dynamically generated by PeopleCode using data in the buffer. <hr/>
Formed URL	This field appears only after you click the Show Formed URL link. The formed URL appears within a display-only edit box.
Test Related Content Service	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 Parameters

This table describes service URL parameter fields. These fields are common to all service URL types.

Parameter Name	<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 Chapter 15, "Developing and Configuring Related Content Services," Service Parameter Mapping, page 574.</p>
Required	<p>Select this check box when the additional parameter is necessary for a service to return a valid output. This check box is deselected 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>Important! 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/> <p>See Chapter 15, "Developing and Configuring Related Content Services," Service Parameter Mapping, page 574.</p>

Description	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 option fields. Display options apply only when you configure the to appear in the related content frame. These fields are common to all service URL types.

Refresh	Select to enable the administrator to include the refresh icon in the related content frame; this icon enables the user to force a server trip to retrieve and refresh the data.
New Window	<p>Select to enable the administrator to include the new window icon in the related content frame; this icon enables the user to send the service to a new browser window.</p> <p>When the service opens a new browser window, the service data in the new window is not actively linked to the parent transaction page. If the user searches for a different key value on the parent page, the content of the new window does <i>not</i> change; however, you can continue to view and use it. This is important in situations in which you are viewing content and might want to continue using it after completing work in the transaction page.</p>

Note. Any browser setting that enables the browser to determine whether to open pages in a new window or a tab can affect the expected functionality of the `New Window` property. Configure the *BNEWWIN* navigation personalization option to override any user-specific browser settings so that the browser never uses tabbed browsing for PeopleSoft applications.

See *PeopleTools 8.52: Security Administration*, "Managing PeopleSoft Personalizations," Understanding Navigation Personalizations.

Pages Used to Define Related Content Services

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Define Related Content Service	PTCSSERVICES	PeopleTools, Portal, Related Content Services, Define Related Content Service	<ul style="list-style-type: none"> • Create and maintain application class related content service definitions. • Create and maintain non-PeopleSoft URL related content service definitions. • Create and maintain PeopleSoft component related service definitions. • Create and maintain embeddable pagelet related content service definitions. • Create and maintain OBIEE related content service definitions. • Create and maintain PeopleSoft Query related content service definitions. • Create and maintain PeopleSoft script related content service definitions.
Test Related Content Service	PTCS_TESTURL	On the Define Related Content Service page, click the Test Related Content Service link.	Test related content service definitions.
Related Content Service Definition - Help information	PTCS_HELPTEXT_SEC	On the Define Related Content Service page, click the Write Help Text button.	Write Help Text for related content service definitions.
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.

Creating and Maintaining Application Class Related Content Service Definitions

Access the Define Related Content Service page. (Select PeopleTools, Portal, Related Content Service, Define Related Content Service) Select the *Application Class URL* URL type.

The screenshot shows the 'Define Related Content Service' page for the 'Application Class' URL type. At the top, there's a dropdown menu for '*URL Type:' set to 'Application Class', with buttons for 'Write help text' and 'Copy Service Definition'. Below this is a section titled 'URL Information' containing 'Application Class Parameters' with input fields for 'Package', 'Path', and 'Class ID', each with a search icon. A note states 'parameter names are case-sensitive.' There are checkboxes for 'Post mapping definition data.' and 'Escape URL Parameters' (which is checked). Below this is a table for 'Service URL Parameter' with columns for '#Parameter Name', 'Required', and 'Description'. The table has one row with index '1'. At the bottom, there are links for 'Show Formed URL' and 'Test Related Content Service'.

Define Related Content Service page - Application Class URL type

Application Class Parameters

- Package** Enter the name of the application class package that you want to use to define the related content service.
- Path** Enter the names—delimited by colons—of each subpackage in the hierarchy that drills down to the class that contains the method that defines the service. If the class is defined in the top-level package, the Path lookup search results display only a colon. Select the colon.
- Class ID** Enter the name of the application class that you want to use to define and build the related content service. The class must exist in the application package and subpackage that you specify.

These fields are validated when you save the definition; the application does not allow you to save the service if you enter an invalid application class.

Service URL Parameters

Application class service definitions do not require service URL parameters except when you want to map field-level data to the service and display the service in a field-level contextual menu.

Application class URLs cannot be formed by clicking the Show Formed URL link because the URL is dynamically generated by PeopleCode using data in the buffer. For the same reason, application class URLs cannot be tested by clicking the Test Related Content Service link.

Application Class PeopleCode Example

This is an example of application class PeopleCode that could be used to create an application class service for use as a related action on a SmartNavigation page:

```

import PT_RCF:ServiceInterface;

class res_det implements PT_RCF:ServiceInterface
    method execute();
end-class;

Global number &PRL_Schedule_Current_Row;

method execute
    /* Extends/implements PT_RCF:ServiceInterface.execute */

    Evaluate %Component
    When = Component.PGM_RESOURCE_LIST

    /*****
    Transfer to the Resource detail page.
    *****/
    If (%Page = Page.PC_PRL) Then
        GetPage(Page.PC_PRL_SCHED).Visible = True;
        &PRL_Schedule_Current_Row = CurrentRowNumber();

        /*AN: Hide the grids if the resource is not a resource class */
        If (GetRow().PROJECT_TEAM.TEAM_MEMBER_CLASS.Value <> "L") Then
            GetRow().GetRowset(Scroll.PROJ_TEAM_SCHED).HideAllRows();
            GetRow().GetRowset(Scroll.PC_TEAM_IP_SCHD).HideAllRows();
        Else
            Local Rowset &ipRowSet = GetRow().GetRowset(Scroll.PC_TEAM_IP_SCHD);
            If (&ipRowSet.ActiveRowCount > 1 Or
                (&ipRowSet.ActiveRowCount = 1 And
                 All(&ipRowSet(1).PC_TEAM_IP_SCHD.START_DT.Value))) Then
                GetRow().GetRowset(Scroll.PC_TEAM_IP_SCHD).ShowAllRows();
            Else
                GetRow().GetRowset(Scroll.PC_TEAM_IP_SCHD).HideAllRows();
            End-If;
        End-If;

        Local Rowset &arlResDtlVw = GetRow().GetRowset(Scroll.PC_ARL_RESDTLVW);
        &arlResDtlVw.Flush();
        &arlResDtlVw.Select(Record.PC_ARL_RESDTLVW, "where BUSINESS_UNIT =>
= :1 AND PROJECT_ID = :2 AND TEAM_MEMBER = :3 and TEAM_MEMBER_CLASS =>
= :4 and TEAM_MEMBER_IND = :5 ", PROJECT.BUSINESS_UNIT, PROJECT.PROJECT_ID, =>
GetRow().PROJECT_TEAM.TEAM_MEMBER.Value, =>
GetRow().PROJECT_TEAM.TEAM_MEMBER_CLASS.Value, =>
GetRow().PROJECT_TEAM.TEAM_MEMBER_IND.Value);

        TransferPage(Page.PC_PRL_SCHED);

    End-If;

    /*AN: Hide the grids if the resource is not a resource class */
    If (&teamRow.PROJECT_TEAM.TEAM_MEMBER_CLASS.Value <> "L") Then
        &teamRow.GetRowset(Scroll.PROJ_TEAM_SCHED).HideAllRows();
        &teamRow.GetRowset(Scroll.PC_TEAM_IP_SCHD).HideAllRows();
    Else
        Local Rowset &ipSet = &teamRow.GetRowset(Scroll.PC_TEAM_IP_SCHD);
        If (&ipSet.ActiveRowCount > 1 Or
            (&ipSet.ActiveRowCount = 1 And
             All(&ipSet(1).PC_TEAM_IP_SCHD.START_DT.Value))) Then
            &teamRow.GetRowset(Scroll.PC_TEAM_IP_SCHD).ShowAllRows();
        Else
            &teamRow.GetRowset(Scroll.PC_TEAM_IP_SCHD).HideAllRows();
        End-If;
    End-If;

```

```

End-If ;
End-If ;

TransferPage( Page.PC_PRL_SCHED ) ;

End-If ;

End-Evaluate ;

end-method ;

```

Creating and Maintaining Non-PeopleSoft Related Content Service Definitions

Access the Define Related Content Service page. (Select PeopleTools, Portal, Related Content Service, Define Related Content Service.) Select the *Non-PeopleSoft URL* URL type.

The screenshot shows the 'Define Related Content Service' page for the 'Non-PeopleSoft URL' type. At the top, there's a dropdown menu for 'URL Type' set to 'Non-PeopleSoft URL', with a 'Write help text' button and a 'Copy Service Definition' button. Below this is a section titled 'URL Information' with a collapsed arrow. Inside, there's a text field for '*External URL:' and a note: 'Note: parameter names are case-sensitive.' To the right of the note are two checkboxes: 'Post mapping definition data.' (unchecked) and 'Escape URL Parameters' (checked). Below these is a table titled 'Service URL Parameter' with columns: '#', '*Parameter Name', 'Required', and '*Description'. The table has one row with index '1', an empty parameter name field, an unchecked 'Required' checkbox, and an empty description field. To the right of the table are 'First', '1 of 1', and 'Last' navigation buttons. Below the table are two links: 'Show Formed URL' and 'Test Related Content Service'. At the bottom is a large text area labeled 'Formed URL:'.

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 and Maintaining PeopleSoft Component Related Content Service Definitions

Access the Define Related Content Service page. (Select PeopleTools, Portal, Related Content Service, Define Related Content) Select the *PeopleSoft Component URL* URL type.

Note. PeopleSoft components service definitions can be anonymous. You define this type of service explicitly when you need to include a service parameter that does not appear with the anonymous service.

*URL Type:

Peoplesoft Component

?

Write help text

Copy Service Definition

▼ URL Information

*Node Name:

Component Parameters

*Menu Name:

*Market:

*Component Name:

Page 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"/>	<div><div>+</div><div>-</div></div>

First

1 of 1

Last

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.

Page Name

Enter the page definition name as configured in Application Designer. Use this field to open a specific page in the component when it appears in the related content target area.

Component URL Example

This is an example of a fully formed component URL:

```
http://mymachine.peoplesoft.com/ps/ps/EMPLOYEE/PT_LOCAL/c/MAINTAIN_SECURITY.USERMAINT.GBL?OPRID=PTPORTAL&OPRDEFNDESC=Portal Administrator
```

Copyright © 1988, 2011, Oracle and/or its affiliates. All Rights Reserved.

561

Creating and Maintaining Embeddable Pagelet Related Content Service Definitions

Access the Define Related Content Service page. (Select PeopleTools, Portal, Related Content Service, Define Related Content) Select the *Embeddable Pagelet* URL type.

The screenshot shows the 'Define Related Content Service' page for the 'Embeddable Pagelet' URL type. At the top, there is a dropdown menu for '*URL Type:' set to 'Embeddable Pagelet', with buttons for 'Write help text' and 'Copy Service Definition'. Below this is a section titled 'URL Information' containing fields for '*Node Name:' and '*Pagelet ID:'. A note states 'parameter names are case-sensitive.' and there are checkboxes for 'Post mapping definition data.' and 'Escape URL Parameters' (which is checked). A table titled 'Service URL Parameter' has columns for 'Parameter Name', 'Required', and 'Description'. The first row shows a parameter name field, a 'Required' checkbox, and a 'Description' field. At the bottom of the table are '+', '-', and 'x' icons. Below the table are links for 'Show Formed URL' and 'Test Related Content Service'.

Define Related Content Service page - Embeddable Pagelet URL type

URL Information

Pagelet ID

Enter the unique identifier of the embeddable pagelet created in Pagelet Wizard.

See Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Step 2: Selecting a Pagelet Data Source, page 358.

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 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 supplies 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 to Create and Manage Pagelets," Step 3: Specifying Pagelet Data Source Parameters, page 372.

Embeddable Pagelet URL Example

This is an example of a fully formed embeddable pagelet URL:

`http://mymachine.peoplesoft.com/psc/ps/EMPLOYEE/PT_LOCAL/s/WEBLIB_PTEMBED.ISCRIPT1.FieldFormula.IScript_EmbeddedPageletBuilder?PAGELETID=USER_ROLES&OPRID=PTDMO&.MAXROWS=10`

Creating and Maintaining OBIEE Pagelet Related Content Service Definitions

Access the Define Related Content Service page. (Select PeopleTools, Portal, Related Content Service, Define Related Content) Select the *PeopleSoft OBIEE Pagelet* URL type.

*URL Type: 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"/>	+	-

[Show Formed URL](#) [Test Related Content Service](#)

Define Related Content Service page - 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 to Create and Manage Pagelets," Step 1: Entering Pagelet Identifying Information, page 356.

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 Pagelet Wizard.

If the pagelet resides in the same system, then 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 supplies 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 to Create and Manage Pagelets," Step 3: Specifying Pagelet Data Source Parameters, page 372.

OBIEE Pagelet URL Example

This is an example of a fully formed OBIEE pagelet URL:

`http://mymachine.peoplesoft.com/psc/ps/EMPLOYEE/PT_LOCAL/s/WEBLIB_PTEMBED.ISCRIPT1.FieldFormula.IScript_EmbeddedPageletBuilder?PAGELETID=OBIEE_REPORT&1) Markets.Region=EASTERN REGION`

Creating and Maintaining PeopleSoft Query Related Content Service Definitions

Access the Define Related Content Service page. (Select PeopleTools, Portal, Related Content Service, Define Related Content.) Select the *PeopleSoft Query* URL type.

The screenshot shows the 'Define Related Content Service' page for the 'PeopleSoft Query' URL type. At the top, there's a dropdown menu for 'URL Type' set to 'PeopleSoft Query', with buttons for 'Write help text' and 'Copy Service Definition'. Below this is a section titled 'URL Information' containing fields for '*Node Name:' and '*PS Query:', a note about case-sensitive parameter names, and checkboxes for 'Post mapping definition data.' and 'Escape URL Parameters'. At the bottom is a 'Service URL Parameter' table with columns for 'Parameter Name', 'Required', and 'Description'. The table has one row with a search icon, a plus button, and a minus button. At the very bottom are links for 'Show Formed URL' and '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 supplies 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 *PeopleTools 8.52: PeopleSoft Query*, "Defining Selection Criteria," Defining Prompts.

PeopleSoft Query URL Example

This is an example of a fully formed PeopleSoft Query URL:

```
http://mymachine.peoplesoft.com/psc/ps_newwin/EMPLOYEE/PT_LOCAL/q/?ICAction=ICQryNameURL=PUBLIC.MESSAGES_FOR_MSGSET&MESSAGE_SET_NBR=240
```

Creating and Maintaining PeopleSoft Script Related Content Service Definitions

Access the Define Related Content Service page. (Select 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:

FieldFormula

*PeopleCode Function Name:

IScript_

Note: parameter names are case-sensitive.

☐ Post mapping definition data.

☒ Escape URL Parameters

Service URL Parameter

*Parameter Name

Required

*Description

1

☐

+

-

First

1 of 1

Last

Show Formed URL

Test Related Content Service

Define Related Content Service page - PeopleSoft Script URL type

See *PeopleTools 8.52: 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://mymachine.peoplesoft.com/psc/ps/EMPLOYEE/PT_LOCAL/s/WEBLIB_STOCK.ISCRIPT1.FieldFormula.IScript_StockQuote?stockquote=ORCL
```

Copyright © 1988, 2011, Oracle and/or its affiliates. All Rights Reserved.




565

Testing Related Content Service Definitions

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

Parameter Name	Displays the parameters that you enter on the Define Related Content Service page. This field is display-only.
Parameter Value	Enter a test value for the parameter.
Test Related Content Service	Click to test the related content service with the parameter values that you enter.
Return	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.

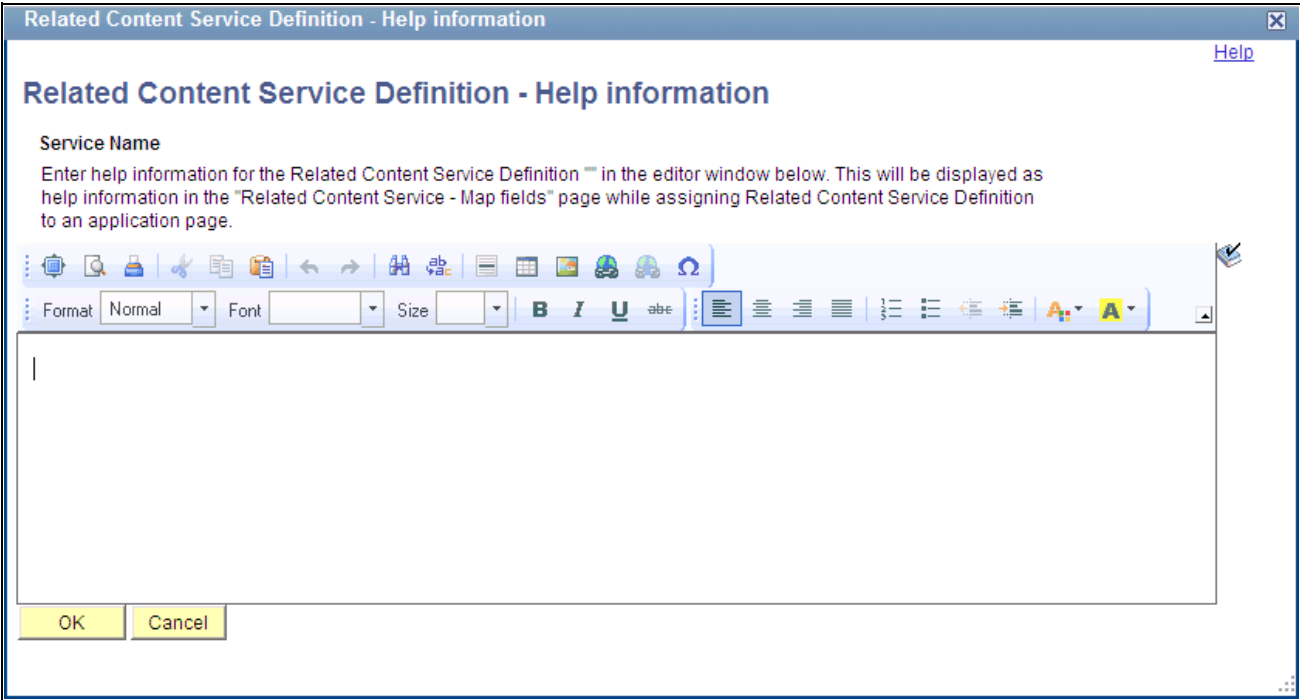
Writing Help Text for Related Content Service Definitions

When you define related content services, you may want to document the service and how the service parameters should be mapped to transaction content, especially if your services use multiple, complex service parameters. The components and pages can include numerous fields, some of which are named similarly, which can make configuring the service parameter mapping difficult.

Use the Help text to clearly communicate the values that must be assigned to the service parameters for the service. The text that you enter is accessible to the subject matter expert when she clicks the Read help text link on the Related Content Service - Configure Service page.

Important! You must use explicit, concise, clear, understandable language in your instructions.

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 page

Service Name	Displays the name of the related content service. This field is display-only.
OK	Click to save the text and return to the Related Content Service Definition page.
Note. You must save the service definition to save the help text.	
Cancel	Click to discard any changes and return to the Related Content Service Definition page.

To write Help text for a related content service definition:

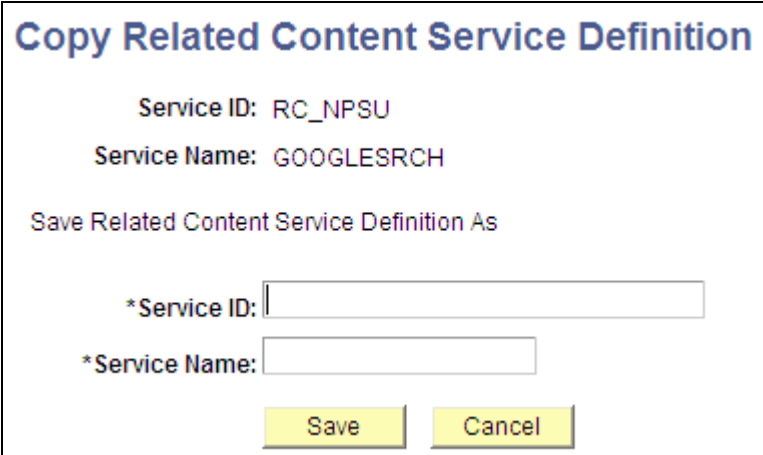
1. Enter help information for the related content service definition in the editor window.
2. Click the OK button.

This returns you 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 Define Related Content Service 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 discard the related content definition that you copied.

Note. Clicking the Cancel button does not cancel the save. The transaction is saved when you click the Save button.

PeopleSoft Application Designer Projects

You can use PeopleSoft Application Designer to create upgrade projects that contain related content definitions and related content services, which include related content configurations and menu layouts. You can save these projects to files or copy them directly to other PeopleSoft databases.

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Working With Projects."

Assigning and Managing Related Content Services

This section provides overviews of related content assignment, related actions assignment, and service configuration and discusses how to:

- Select a content reference.
- Assign component services to content references.
- Assign page level services to content references.
- Configure related content services.

Understanding Related Content Assignment

The Related Content Framework is designed so that subject matter experts or business analysts with an understanding of PeopleSoft applications and your business processes can assign related content services to components with little or no assistance from the developer after the service is defined. Almost every aspect of assigning and configuring related content services is done in the browser; no access to developer tools, such as PeopleSoft Application Designer, is necessary.

The Manage Related Content Configuration page is the starting point for creating and assigning related content services. 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.

When you design and configure related information correctly, the application user is presented with seamless access to related information, which can increase productivity and improve job performance.

Assigning Related Content Process Flow

Related content assignment follows this basic process flow:

1. Select the content reference.

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 because components on different menus might have different business requirements.

2. Include hidden content references (Optional).

Selecting this check box redraws the tree including hidden content references in the system. Deselecting it redraws the tree with only the visible content references in the system.

3. Assign component-level related content to the content reference.
4. Assign page-level related content to the content reference.
5. Configure the assigned services.
6. Save the configuration.

Managing Related Content

After you have defined related content services and assigned them to content references, the assignments appear in the Content References grid that is on the Manage Related Content Configuration page. Use the Edit Configuration buttons on the Manage Related Content page to modify existing related content assignments.

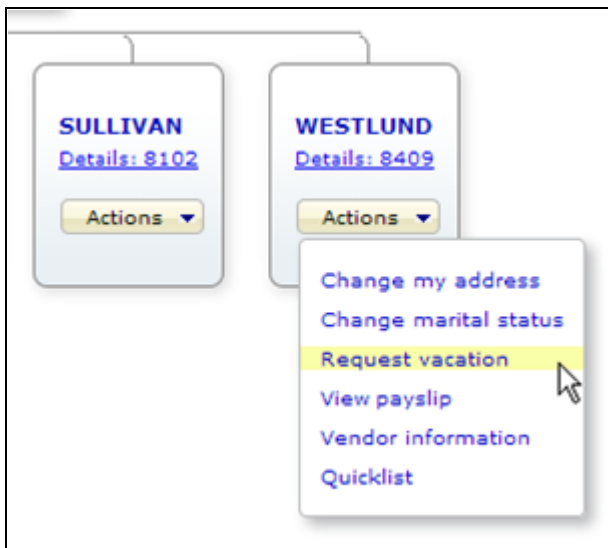
When you click the Edit Configuration button, the Assign Related Content page appears.

Understanding Related Actions Assignment

Related actions links can appear in an Actions drop-down menu, which you find on nodes that appear on SmartNavigation pages. SmartNavigation pages are dynamically generated navigation structures that combine your own hierarchical application data from trees and rowsets, along with PeopleSoft charting metadata to provide users with an alternative way to access transaction data in components.

See [Chapter 5, "Administering Portals," Defining Folders, page 111](#).

This example shows two nodes on a SmartNavigation page and the Actions drop-down menu.



SmartNavigation node showing Actions drop-down menu and related actions links

Related action links can also appear in the search results list of a keyword search.

This example shows the Related Actions drop-down menu that appears in the results of a keyword search:

Search Results

Filter by

Location

- United States (184)
- France (50)
- Canada (23)
- Germany (7)
- India (7)
- More...



Status

- Active (310)
- Template (3)

Support Team

- Enterprise Support Team (80)
- FrenchSupport Team (43)
- International Support Team (41)
- Super Support Team (32)
- Test Team (30)
- More...

Note: Keyword Search will return data updated 4 days ago (2011-06-03-16.29.23.000000)
200 of 313 results are displayed.

View as:  

[FRA05 - FRANCE 5 Bike and Equip - SHARE](#)
Last Updated Date:2009-07-24-00.00.00.000000
FRANCE 5 - FRANCE 5 Bike and Equip - Customer since: 1900-01-01
▼ **Related Actions**

- Select Conversations to Review
- Contact Information
- Google
- Messages
- Corporate Tree Messages

[erunit Custom - SHARE](#)
000
erunit Custom - Customer since: 1900-01-01
[ARE](#)
000
e: 2006-01-01

[0000050036 - We Ship Express - SHARE](#)
Last Updated Date:2009-07-24-00.00.00.000000
We Ship - We Ship Express - Customer since: 1975-04-08
▼ **Related Actions**








[1000 - Alliance Group - SHARE](#)
Last Updated Date:2009-07-27-00.00.00.000000
Alliance - Alliance Group - Customer since: 1900-01-01
▼ **Related Actions**

Search results list showing Related Actions drop-down menu and related actions links

You configure related actions similarly to the way that you configure related content. However, the component to which you assign the related content service might not seem obvious. These next sections explain how to find the component to which you want to assign a related content service that you want to appear in the Actions drop-down menu on a SmartNavigation page.

Finding the Details Component for Tree-Based SmartNavigation Pages

Trees are hierarchical collections of data that you create and maintain by using PeopleSoft Tree Manager. Tree-based SmartNavigation folders use a specific type of tree called a detail tree as the data source. When you use PeopleSoft Tree Manager to configure a detail tree structure, you define the location of the component that you use to maintain the detail values in the tree. When you assign a related content service that you want to appear as a related action, you assign the service to the detail component, which underlies the nodes and leaves of the tree structure. To find this information, access the Details page on the Tree Structure component. (Select Main Menu, Tree Manger, Tree Structure and then select the Details tab.) Use the value in the Component Name field, which is the *DEPARTMENT* component in this example:

Structure	Levels	Nodes	Details
Tree Details			
Structure ID:	DEPTID		
Record Name:	<input type="text" value="DEPT_TBL"/> 		
Field Name:	<input type="text" value="DEPTID"/> 		
Page Name:	<input type="text" value="DEPARTMENT"/> 		
Component Name:	<input type="text" value="DEPARTMENT"/> 		
Menu Name:	<input type="text" value="DESIGN_CHARTFIELDS"/> 		
Menu Bar Name:	<input type="text" value="USE"/> 		
Menu Item Name:	<input type="text" value="DEPARTMENT"/> 		

Tree Structure - Tree Details page

Note. Nodes and leaves can have different details components. The Nodes tab displays the tree node details component. When setting up related actions for a tree, you can configure services for either one or both.

See *PeopleTools 8.52: PeopleSoft Tree Manager*, "Creating Trees," Defining Tree Details.

Finding the Details Component for Rowset-Based SmartNavigation Pages

Rowsets are collections of data that are associated with buffer data. Rowset-based SmartNavigation pages use PeopleCode to populate the component buffer, generate the chart, and retrieve related actions. When the developer writes the PeopleCode programs that populate the chart, she must define a detail component from which the data is retrieved. When you assign a related content service that you want to appear as a related action, you assign the service to the detail component indicated in the PeopleCode program.

To find this information you can consult the developer or you can follow these steps:

1. Select PeopleTools, Portal, Structure & Content.
2. Navigate to the SmartNavigation folder by clicking the folder labels until you reach the appropriate folder.

When you find the folder, the label will be text, not a link.

3. Click the Edit link for that folder.
4. Use the PeopleCode Type and the iScript Parameters or Application Class Parameters fields to determine the name of the PeopleCode function or class that contains the information.
5. Open PeopleSoft Application Designer and access the record or class that you ascertained in the previous step.

6. Search for a line that sets the detail component. The exact code can vary depending on the naming conventions of the programmer, but it should set these elements:

- MenuName
- BarName
- ItemName
- Page
- Action
- Record
- KeyFieldNumber
- FieldName

This is an example: "MenuName.QE_SAMPLE_APPS, BarName.USE, ItemName.QE_EMPLOYEE, Page.QE_EMPLOYEE, Action.U, Record.QE_EMPLOYEE, KeyFieldNumber.001, FieldName.EMPLID"

Including Related Actions Links in Search Results

The appearance of related actions links in the search results list of a keyword search depends on several configuration options. Consult the PeopleSoft Search Technology documentation for detailed information.

See *PeopleTools 8.52: PeopleSoft Search Technology*, "Working with PeopleSoft Search," Working with Related Actions.

Assigning Related Actions Process Flow

Related content assignment follows this basic process flow:

1. Select the content reference.

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 because components on different menus might have different business requirements.

2. Include hidden content references (Optional).

Selecting this check box redraws the tree including hidden content references in the system. Deselecting it redraws the tree with only the visible content references in the system.

3. Assign component level services to the content reference.
4. Assign page level services to the content reference.
5. Configure the assigned services.
6. Save the configuration.

Managing Related Actions

After you have defined related content services and assigned them to content references, the assignments appear in the Services assigned to Application Pages grid, which is on the Manage Related Content Configuration page. Use the Edit Configuration buttons on the Manage Related Content page to modify existing related actions assignments.

When you click the Edit Configuration button, the Assign Related Content page appears. Click the Related Actions tab to access the Assign Related Actions page.

See Also

PeopleTools 8.52: PeopleSoft Tree Manager, "Creating Trees," Defining Tree Details

PeopleTools 8.52: PeopleCode Developer's Guide, "Using the PeopleCode Editor," Accessing Definitions and Associated PeopleCode

Understanding Service Configuration

When you assign a related content service to a page or component, you must configure the service parameters. Configuration can include mapping the field data to the service parameters, specifying which type of menu the service link is listed in, adding filters to the service, and specifying security options. The service parameter list and other configuration options vary based the service type, the service definition, and the service target that you choose.

Related content service configuration establishes the relationships between the service and the transaction data so that the service appears in context with the current transaction. To establish contextually relevant relationships, service configuration includes working with:

- Service parameter mapping.
- Service filters.
- Security options.

Service Parameter Mapping

Many services include parameters that you must map to key fields or page fields, system variables, and fixed values. You map the parameters to these elements so that they can pass the data to the service and create the relevant context for the related content.

This table discusses the source of the parameters for each of the service types.

Service Type	Parameter Source
Content reference	<p>Search key fields from the component search record</p> <p>Note. If the search record has no key fields, the systems retrieves search key fields from the component context search record.</p>

Service Type	Parameter Source
Pagelet	Values in the Name field if the RCF_SERVICE_PARAMETER content reference attribute exists on the Content Reference Administration page.
Service	Values in the URL Parameters grid for the service definition.

When you map parameters, usually you must first select the mapping type and then select the mapping data. Some services might already have the mapping defined.

This table discusses the mapping types.

Mapping Type	Description	Example
Fixed value	A hard coded value that does not change respective to context	Service that includes the service ID as a parameter for example, PTWC_WIKIS
Key field	Primary or composite keys from the component	In the department table, DEPT_ID
Page field	Any level 0, 1, 2, or 3 field in the component buffer	In the education table, SCHOOL_NAME
System variable	Any variable that the PeopleSoft system resolves at runtime	%USERID, the current user

After you select a mapping type, click the Select button to see the mapping type prompt page.

Note these points about the prompt pages:

- If the mapping type is *Key Field*, then the prompt page displays a list of search key fields from the component search record.
- If the mapping type is *Page Field*, then the prompt page constructs a grid and dynamically populates it with a row for each page field that is available to the service. For each page field row, the grid also includes a hidden field indicator, the occurs or scroll level, the record (table) name, the field definition name, and the page field ID number.

Use the page field ID number to identify a specific occurrence of a page field when the field appears multiple times in the same occurs level on the page. Each instance of a page field has a unique ID number that you find on the Order tab of the page definition in PeopleSoft Application Designer.

- If the mapping type is *System Variable*, then the prompt page displays valid system variables.

See *PeopleTools 8.52: PeopleCode Language Reference*, "System Variables."

Important! Only %UserID, %EmployeeId, and %EmailAddress are supported for services to be displayed in the Related Actions menu of SmartNavigation pages.

If the mapping type is *Fixed Value*, then the Select button is hidden.

Note. To reduce mapping effort, the PeopleSoft Related Content Framework defines a set of reserved words and service parameters.

See [Appendix E, "Related Content Reserved Words and Service Parameters," page 723.](#)

Service Filters

A service filter hides or displays a service link in a field-level menu or the Related Content menu based on the value in the field at runtime. The Application Package PeopleCode that you specify can access the component buffer, using field values to trigger data-specific logic that causes the related content service link to be visible or hidden based on the value of the field. The filter enables the context of the data to determine whether the service link appears.

Security Options

The service definition determines the choices of security options that appear.

See [Chapter 15, "Developing and Configuring Related Content Services," Understanding Service Security, page 550.](#)

Prerequisites

This table discusses the prerequisites for assigning related content.

Prerequisite	Cross Reference
For all service types other than PeopleSoft components and pagelets, which are anonymous, the service definitions must exist before you can assign them to a component or page as related content.	See Chapter 15, "Developing and Configuring Related Content Services," Defining Related Content Services, page 546.
For PeopleSoft components that require parameters other than those which the search record or contextual search record supplies, you must create a service definition with the PeopleSoft Component URL type.	See Chapter 15, "Developing and Configuring Related Content Services," Creating and Maintaining PeopleSoft Component Related Content Service Definitions, page 561.
For pagelets that require parameters, you must enter the RCF_SERVICE_PARAMETER attribute.	See Chapter 15, "Developing and Configuring Related Content Services," Content Reference Attributes as Service URL Parameters, page 549.

Common Elements Used in This Section

Package Enter the application class package name to use for the service filter.

Path

Enter the names—delimited by colons—of each subpackage in the hierarchy that drills down to the class that contains the method that filters the service. If the class is defined in the top-level package, the Path lookup search results display a colon only. Select the colon.

Note. The application class parameters are validated when you save the definition; the service does not allow you to save an invalid application class.

ClassID

Enter the class name that defines the data-specific logic.

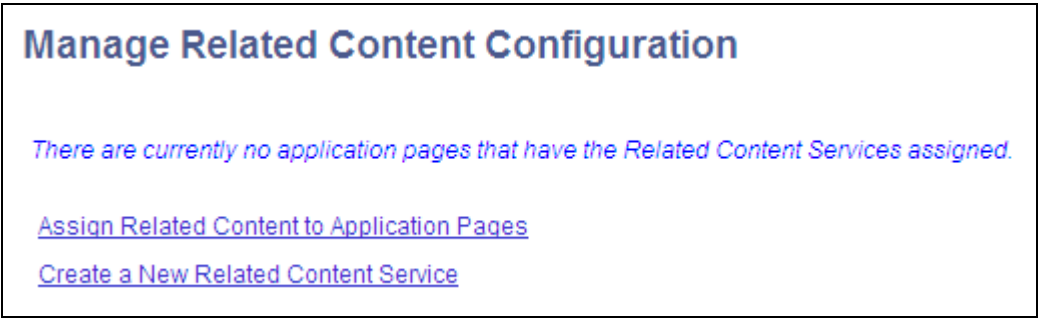
Pages Used to Assign and Manage Related Content Services

<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	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 a content reference for related content assignment.
Assign Related Content	PTCS_SERVICECFG	<p>You can access this page in the following ways:</p> <ul style="list-style-type: none"> On the Select a Content Reference page, select a content reference. On the Manage Related Content Service page, click Edit Configuration button. 	Assign related content to a component or page.
Assign Related Actions	PTCS_SERVICECFG_RA	<p>You can access this page in the following ways:</p> <ul style="list-style-type: none"> On the Select a Content Reference page, select a content reference. On the Manage Related Content Service page, click Edit Configuration button. Select the Configure Related Actions tab. 	Assign related actions to a component or page.

Page Name	Definition Name	Navigation	Usage
Lookup Service ID	PTCS_SVCPRMT_SEC	Select <i>Service</i> as the field type and then click the Select button associated with the service in the Component Level Related Content group box.	Select a service definition.
Lookup Pagelet	PTCS_PGLTPRMT_SEC	Select <i>Pagelet</i> as the field type and then click the Select button associated with the service in the Component Level Related Content group box.	Select a pagelet.
Lookup Content Reference	PTUNDATA	Select <i>Content Reference</i> as the field type and then click the Select button associated with the service in the Component Level Related Content group box.	Select a content reference.
Configure Service	PTCS_SVCCFGCRC_SEC	Click the Configure button associated with a service in the Component Level Related Content group box. Click the Configure button associated with a service in the Page Level Related Content group box.	Configure service parameters.
Page Fields List	PTCS_PGFLDS_SEC	On the Configure Service page, select <i>Page Field</i> as the mapping type. Then, click the Select button.	Select the page field to map to the service parameter.
Key Fields List	PTCS_KEYFLDS_SEC	On the Configure Service page, select <i>Key Field</i> as the mapping type. Then, click the Select button.	Select the key field to map to the service parameter.
System Variables	PTCS_SYSTEMVAR_SEC	On the Configure Service page, select <i>System Variable</i> as the mapping type. Then, click the Select button.	Select the system variable to map to the service parameter.

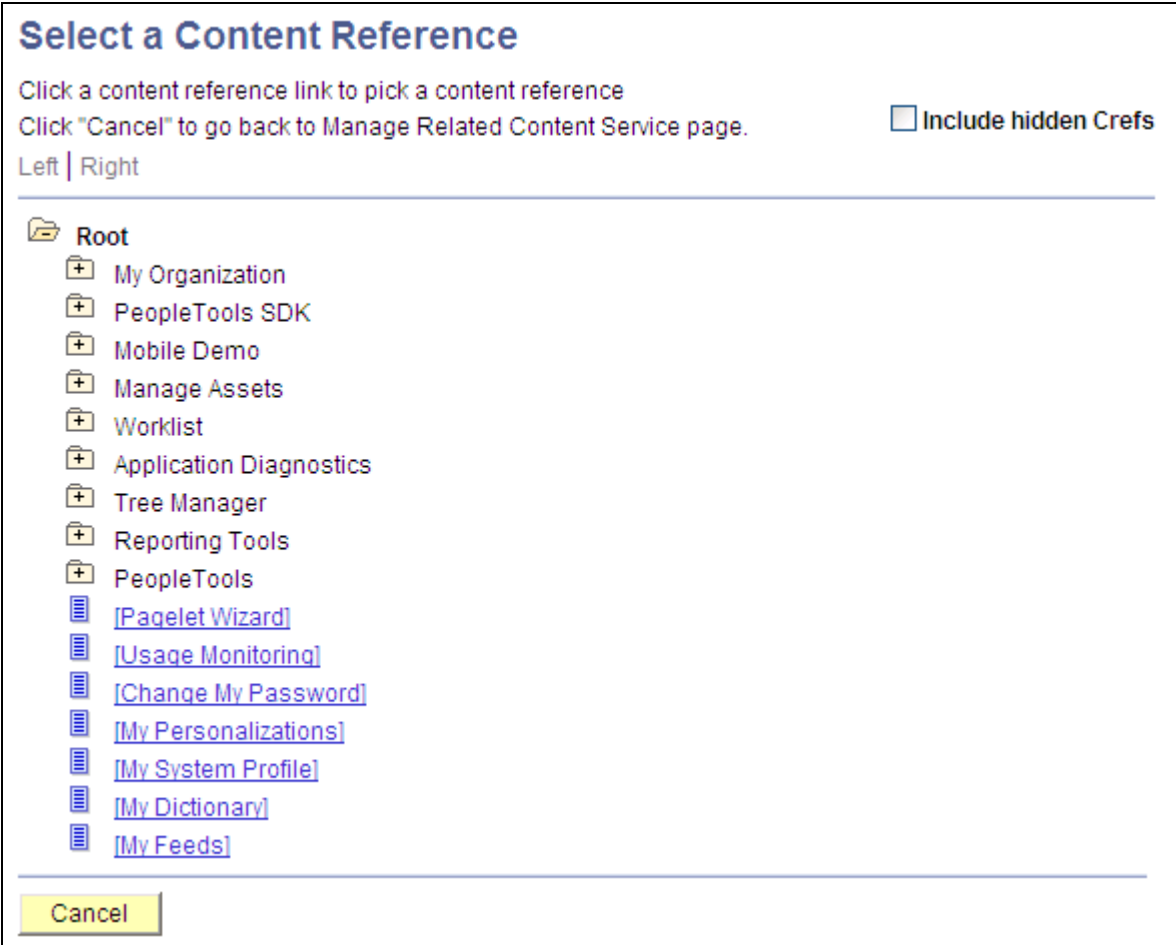
Selecting a Content Reference

Access the Manage Related Content Configuration page. (Select PeopleTools, Portal, Related Content Service, Manage Related Content Service.)



Manage Related Content Configuration page without any existing assignments

Access the Select a Content Reference page. (Click the Assign Related Content to Application Pages link.)



Select a Content Reference page

Select Include hidden Crefs to make hidden content references available for assignment.

Note. The tree shows both PeopleSoft component URLs and PeopleSoft component-based generic URLs.

Assigning Component-Level Services to Content References

At the component level, you can:

- Assign related content.
- Assign related actions.

Assigning Component-Level Related Content

Access the Assign Related Content page.

Assign Related Content

Assign content to be displayed within the Related Content Frame. Use the Configure link to define the parameter mappings and security settings for the service.

Portal Name EMPLOYEE
Content Reference PT_CHANGE_PASSWORD_GBL [Change My Password](#)

Component Level Related Content

Assign Component Level Related Content to be added to the Related Content menu.

Enable	Service Type	Select	Service ID	Service Label	Configure
<input checked="" type="checkbox"/>	Service	Select			Configure

Page Level Related Content

Assign Page Level Related Content to be added to a field level contextual menu.

Enable	*Page	Page Name	Service Type	Select	Service ID	Service Label	Configure
<input checked="" type="checkbox"/>			Service	Select			Configure

[Return to Manage Related Content Configuration page](#)

Assign Related Content page

1. In the Component Level Related Content group box, select the service type.

2. Click the Select button.

The lookup page that appears will depend on the service type you select. If you select *Service* or *Pagelet* as the service type, a lookup page appears.

If you select *Content Reference*, this page appears:

Lookup Content Reference page

If you have multiple PeopleSoft applications that have been configured as remote nodes, you can change the node and the portal, and then select a remote content reference.

3. From the lookup page, select a service.
4. If necessary, modify the label in the Service Label field.

Assigning Component-Level Related Actions

Access the Assign Related Actions page.

Assign Related Actions

Assign services to be used as Related Actions. Select the service target to determine where the service will be displayed. Use the Configure link to define the parameter mappings and options for this service.

Portal Name EMPLOYEE
Content Reference PT_CHANGE_PASSWORD_GBL [Change My Password](#)

Component Level Related Actions

Assign Component Level Related Actions to be added to the Drop-down menu and Search Actions menu.

Enable	Service Type	Select	Service ID	Service Label	*Service Target	Configure
<input checked="" type="checkbox"/>	Service	Select			Target Content	Configure

Page Level Related Actions

Assign Page Level Related Actions to be added to a field level contextual menu.

Enable	*Page	Page Name	Service Type	Select	Service ID	Service Label	Service Target	Configure
<input checked="" type="checkbox"/>			Service	Select			Target Content	Configure

[Return to Manage Related Content Configuration page](#)

Assign Related Actions page

1. In the Component Level Related Actions group box, select the service type.
2. Select the service type.

3. Click the Select button.

The lookup page that appears will depend on the service type you select. If you select *Service* or *Pagelet* as the service type, a lookup page appears.

If you select *Content Reference*, this page appears:

Lookup Content Reference page

If you have multiple PeopleSoft applications that have been configured as remote nodes, you can change the node and the portal, and then select a remote content reference.

4. From the lookup page, select a service.
5. If necessary, modify the label in the Service Label field.

6. Select a service target.

Assigning Page-Level Services to Content References

At the page level, you can:

- Assign related content.
- Assign related actions.

Assigning Page-Level Related Content

Access the Assign Related Content page.

Assign Related Content

Assign content to be displayed within the Related Content Frame. Use the Configure link to define the parameter mappings and security settings for the service.

Portal Name EMPLOYEE
Content Reference PT_CHANGE_PASSWORD_GBL [Change My Password](#)

Component Level Related Content

Assign Component Level Related Content to be added to the Related Content menu.

Enable	Service Type	Select	Service ID	Service Label	Configure
<input checked="" type="checkbox"/>	Service	Select			Configure

Page Level Related Content

Assign Page Level Related Content to be added to a field level contextual menu.

Enable	*Page	Page Name	Service Type	Select	Service ID	Service Label	Configure
<input checked="" type="checkbox"/>			Service	Select			Configure

[Return to Manage Related Content Configuration page](#)

Assign Related Content page

1. In the Page Level Related Content group box, click the Page lookup button and then, select the page to which you want to assign the service.
2. Select the service type.

3. Click the Select button.

The lookup page that appears will depend on the service type you select. If you select *Service* or *Pagelet* as the service type, a lookup page appears.

If you select *Content Reference*, this page appears:

Lookup Content Reference page

If you have multiple PeopleSoft applications that have been configured as remote nodes, you can change the node and the portal, and then select a remote content reference.

4. From the lookup page, select a service.
5. If necessary, modify the label in the Service Label field.

Assigning Page-Level Related Actions

Access the Assign Related Actions page.

Assign Related Actions

Assign services to be used as Related Actions. Select the service target to determine where the service will be displayed. Use the Configure link to define the parameter mappings and options for this service.

Portal Name EMPLOYEE
Content Reference PT_CHANGE_PASSWORD_GBL [Change My Password](#)

Component Level Related Actions

Assign Component Level Related Actions to be added to the Drop-down menu and Search Actions menu.

Enable	Service Type	Select	Service ID	Service Label	*Service Target	Configure
<input checked="" type="checkbox"/>	Service	Select			Target Content	Configure

Page Level Related Actions

Assign Page Level Related Actions to be added to a field level contextual menu.

Enable	*Page	Page Name	Service Type	Select	Service ID	Service Label	Service Target	Configure
<input checked="" type="checkbox"/>			Service	Select			Target Content	Configure

[Return to Manage Related Content Configuration page](#)

Assign Related Actions page

1. In the Page Level Related Actions group box, click the Page lookup button and select the page to which you want to assign the service.
2. Select the service type.

3. Click the Select button.

The lookup page that appears will depend on the service type you select. If you select *Service* or *Pagelet* as the service type, a lookup page appears.

If you select *Content Reference*, this page appears:

Lookup Content Reference page

If you have multiple PeopleSoft applications that have been configured as remote nodes, you can change the node and the portal, and then select a remote content reference.

4. From the lookup page, select a service.
5. If necessary, modify the label in the Service Label field.

6. Select a service target.

Configuring Service Parameters

Access the Configure Service page. (On the Assign Related Content page, click the Configure button.

Configure Service

Page Name USER_ROLES
Service ID QE_MESSAGE_SET3
Service Label Assigned Permissions

Map Parameters ?

	Service Parameter	Required Flag	Mapping Type	Select	Mapping Data	Refresh Service On Change	Is Value Required?	Display in field menu
1	OPRID	<input checked="" type="checkbox"/>	Page Field	Select	User ID	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Service Filter ?

Package Path Class ID

Select Security Options

☒ Public Access

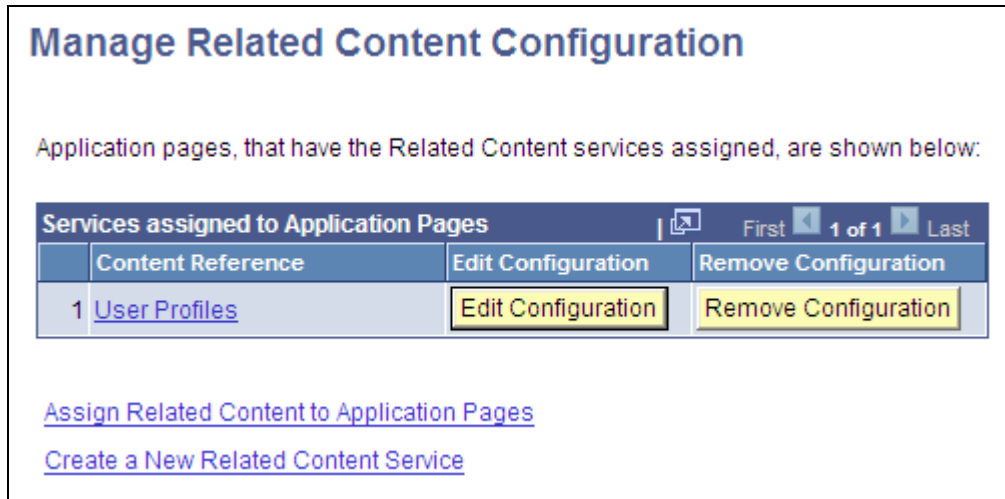
OK Cancel

Configure Service page: page level related content

1. For each service parameter, select the mapping type.
2. For each fixed value mapping type, enter the value in the Mapping Data field.
3. For each key field, page field, or system variable mapping type, click the Select button.
4. Click a link to select the field or variable on the prompt page.
5. For page-level services, select the Display in field menu check box for the field that provides access to the contextual pop-up menu.
6. (Optional) Enter the package, path, and class ID of the service filter .
7. Select the security option.
8. Click the OK button to accept the configuration and return to the assignment page.

Managing Related Content Configuration

Access the Manage Related Content Configuration page. (Select PeopleTools, Portal, Related Content Service, Manage Related Content Service.)



Manage Related Content Configuration page

To manage related content services:

1. Click the [Edit Configuration](#) button to modify the assignment or the service configuration.
2. Click the [Remove Configuration](#) button to delete the assignment from the content reference.

Configuring Menu and Frame Layout

This section provides overviews of the related content menus configuration and the Related Content frames configuration and discusses how to:

- Configure the related actions menus.
- Configure the Related Content menu.
- Configure page field contextual menus.
- Configure the related content frames.

Understanding Menu Configuration

Menus provide access to related content services. Three menu layout options exist:

- Related Actions menus configuration can contain up to 3 levels of folders. Folders appear first in the menu followed by links.

- The Related Content menu configuration is a flat hierarchy and uses no folders.
- Page field contextual menus configuration can contain two sections: related actions and related content.

Related actions appear first; this part of the menu can be hierarchical displaying folders and links.

Related content appears last; this part of the menu is flat displaying links only.

Understanding the Configuration of Related Content Frames

PeopleSoft Related Content Framework enables you to configure related content to appear in two related content frames. The Related Content frames appear beneath and to the right of the transaction page so that users can view the transaction page and the related content simultaneously. When you assign related content to a content reference, you configure the services to appear either in the bottom frame or the right frame; the two frames cannot appear simultaneously.

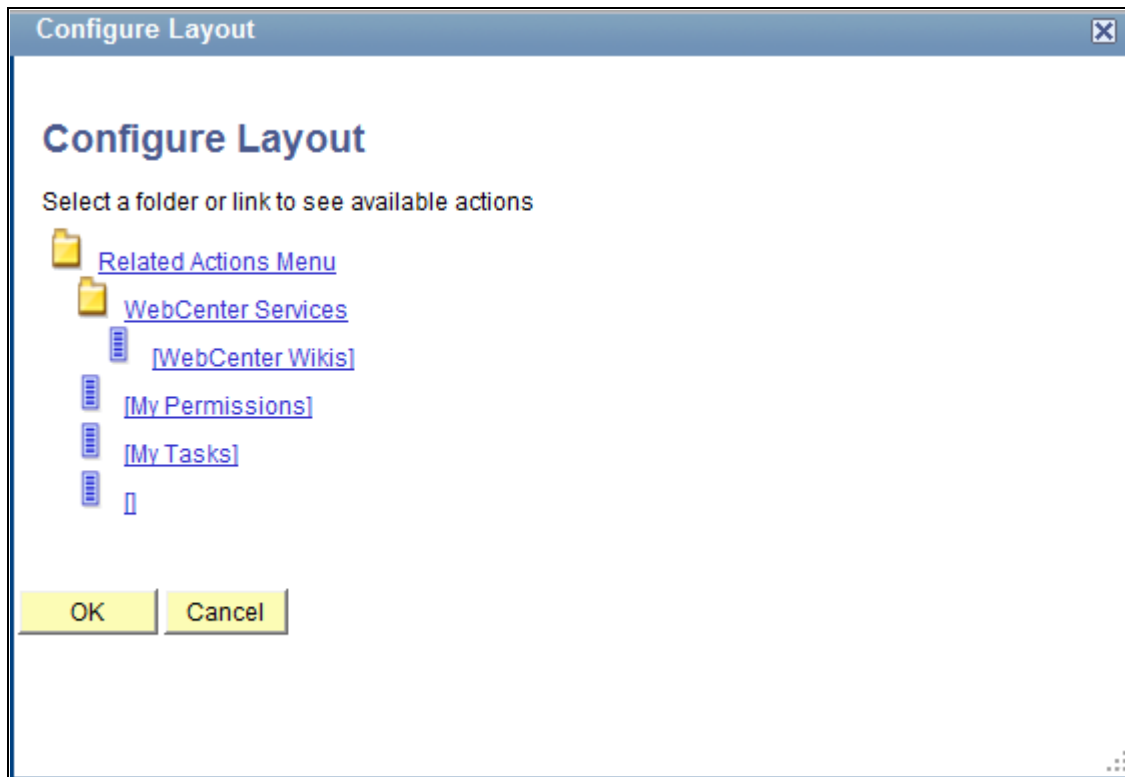
Pages Used to Configure Related Content Service Layout

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Configure Layout	PTCS_MENUCFG	Select PeopleTools, Portal, Related Content Service, Manage Related Content Service. Select the Configure Layout tab.	<ul style="list-style-type: none"> • Configure the Related Actions menu. • Configure the Related Content menu. • Configure page field contextual menus. • Configure the Related Content frames.
Configure Layout	PTCS_MENULAYOUT	You access the Configure Layout page in these ways: Click the Configure Layout button in the Related Action Menu group box. Click the Configure Layout button in the Related Content Menu group box. Click the Configure Layout button in the Page Field Contextual Menu group box.	<ul style="list-style-type: none"> • Configure the Related Actions menu. • Configure the Related Content menu. • Configure page field contextual menus.
Select a folder	PTCS_MENUFLDSEL	Select a service link. Click the Move to Folder button.	Select a folder

Page Name	Definition Name	Navigation	Usage
Configure Related Content Frame	PTCS_MENUCFGOPT	Click the Configure Layout button in the Related Content Frame group box.	Configure the Related Content frames.

Configuring the Related Actions Menu

Access the Configure Layout page (Select PeopleTools, Portal, Related Content Service, Manage Related Content Service. Select the Configure Layout tab. Click the Configure Layout button in the Related Actions Menugroup box.)



Configure Layout page: Related Actions menu

1. Adjust any necessary folders within the hierarchy.

See [Chapter 15, "Developing and Configuring Related Content Services," Working with Folders, page 592](#).

2. Move service links within the hierarchy.

See [Chapter 15, "Developing and Configuring Related Content Services," Working with Services, page 592](#).

See [Chapter 15, "Developing and Configuring Related Content Services," Selecting a Folder, page 592](#).

3. Click the OK button.

4. Click the Save button.

Working with Folders

When you select a folder, these options appear:

Note. Not all options appear for all folders.

Add Folder	Click to add a folder one level below the selected folder. If this option does not appear, you cannot add a subfolder.
Delete Folder	Click to delete the folder. If this option does not appear, you cannot delete the folder.
Move Up	Click to move the folder up within the list of folders in the same level.
Move Down	Click to move the folder down within the list of folders in the same level.

Working with Services

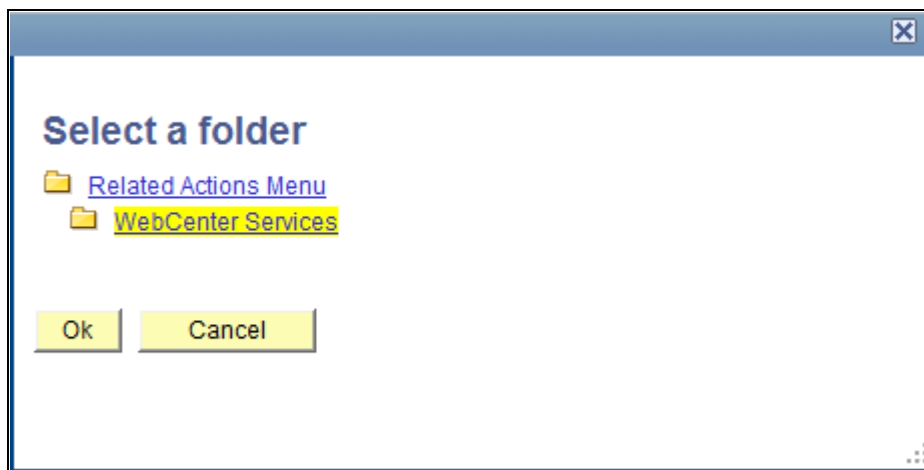
When you select a service, these options appear:

Note. Not all option appear enabled for all services.

Move to Folder	Click to access the Select a folder page, which enables you to move the service to another folder.
Move Up	Click to move the service up within the list of services in the same folder.
Move Down	Click to move the service down within the list of services in the same folder.

Selecting a Folder

Access the Select a folder page (Select a service link. Click the Move to Folder button.)

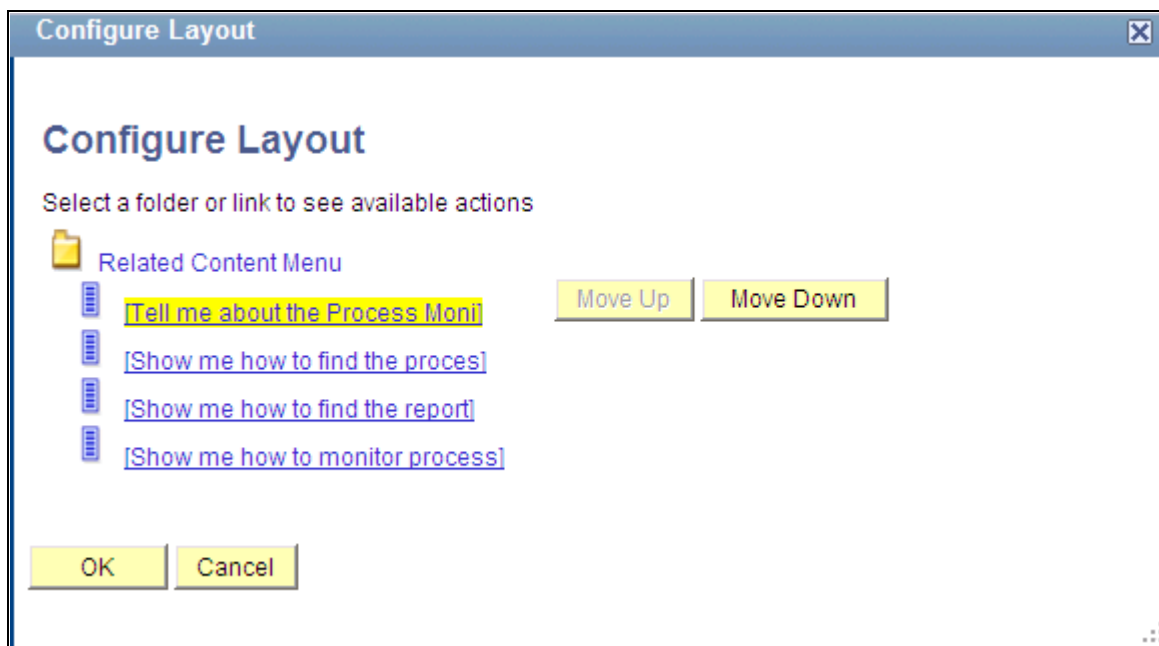


Select a folder page

Select the new folder and click the OK button.

Configuring the Related Content Menu

Access the Configure Layout page (Select PeopleTools, Portal, Related Content Service, Manage Related Content Service. Select the Configure Layout tab. Click the Configure Layout button in the Related Content Menugroup box).



Configure Layout page: Related Content menu

1. Select a service link.
2. Click the Move Up or Move Down button to move the service link.
3. Click the OK button.

4. Click the Save button.

Note. The system does not save the menu configuration until you click the Save button.

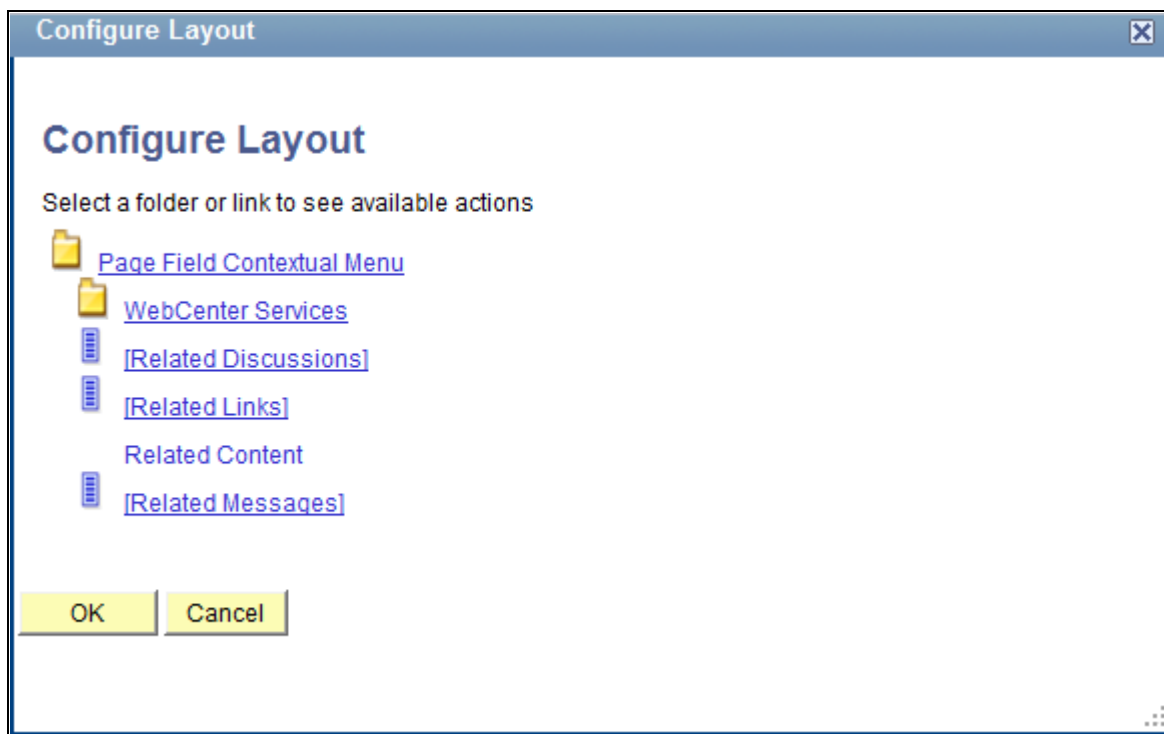
Working with Services

When you select a service, these options appear:

Move Up	Click to move the service up within the list of services.
Move Down	Click to move the service down within the list of services.

Configuring Page Field Contextual Menus

Access the Configure Layout page (Select PeopleTools, Portal, Related Content Service, Manage Related Content Service. Select the Configure Layout tab. Click the Configure Layout button in the Page Field Contextual Menugroup box).



Configure Layout page: Page field contextual menu

1. Adjust any necessary folders within the Page Field Contextual Menu hierarchy.

See [Chapter 15, "Developing and Configuring Related Content Services," Working with Folders, page 592.](#)

2. Move service links within the hierarchy.

Note. Links that appear beneath the Related Content menu can be moved up and down only, not into other folders.

See [Chapter 15, "Developing and Configuring Related Content Services," Working with Services, page 592.](#)

See [Chapter 15, "Developing and Configuring Related Content Services," Selecting a Folder, page 592.](#)

3. Click the OK button.
4. Click the Save button.

Working with Folders

When you select a folder, these options appear:

Note. Not all options appear for all folders.

Add Folder	Click to add a folder one level below the selected folder. If this option does not appear, you cannot add a subfolder.
Delete Folder	Click to delete the folder. If this option does not appear, you cannot delete the folder.
Move Up	Click to move the folder up within the list of folders in the same level.
Move Down	Click to move the folder down within the list of folders in the same level.

Working with Services

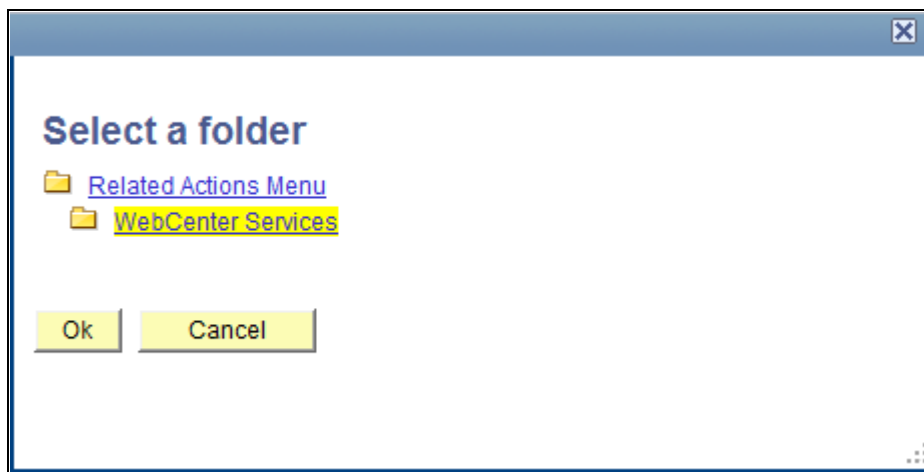
When you select a service, these options appear:

Note. Not all option appear enabled for all services.

Move to Folder	Click to access the Select a folder page, which enables you to move the service to another folder.
Move Up	Click to move the service up within the list of services in the same folder.
Move Down	Click to move the service down within the list of services in the same folder.

Selecting a Folder

Access the Select a folder page (Select a service link. Click the Move to Folder button.)



Select a folder page

Select the new folder and click the OK button.

Configuring the Related Content Frames

Access the Configure Related Content Frame page (Select PeopleTools, Portal, Related Content Service, Manage Related Content Service. Select the Configure Layout tab. Click the Configure Layout button in the Related Content Frame group box).

Configure Layout

Configure Related Content Frame

Portal Name EMPLOYEE

Content Reference PT_ACCESS_CNTRL_LISTX_GBL [Permission Lists](#)

Related Content Location

☒ Bottom Frame ☐ Right Frame

Related Content Frame options | First 1-3 of 3 Last

	Service ID	Service Label	Default Service	Sequence number
1	QE_MESSAGE_SET1	Related Messages	<input type="checkbox"/>	0
2	PTWC_DISCUSSIONS	WebCenter Discussions	<input type="checkbox"/>	0
3	PTWC_DOCUMENTS	WebCenter Documents	<input type="checkbox"/>	0

Layout Options

☐ Open in expanded view

OK Cancel

Configure Related Content Frame page

1. Select the frame location. Choose either *Bottom Frame* or *Right Frame*.

See Chapter 15, "Developing and Configuring Related Content Services," Related Content Location, page 597.

2. Configure the order and initial state of the services that appear in the Related Content frame.

See Chapter 15, "Developing and Configuring Related Content Services," Related Content Frame Options, page 598.

3. Select the frame layout and appearance options.

See Chapter 15, "Developing and Configuring Related Content Services," Layout Options, page 598.

Related Content Location

Bottom Frame

Select to have related content services appear as tabs in the frame beneath the transaction page.

Right Frame

Select to have related content services appear as pagelets in the frame to the right of the transaction page.

Related Content Frame Options**Initially Minimized**

Select so that the service appears in the closed state when the user initially accesses the content reference.

Layout Options**Open in expanded view**

Select to open the Related Content frame automatically when the user accesses the component rather than after the user selects a service.

**Max no of Minimized
Pagelets**

Enter the number pagelets that can appear minimized in the right Related Content frame. If you minimize more than this number, the pagelets move to the More drop-down list at the bottom of the frame. This field is visible only when you select *Right Frame* as the related content location.

Chapter 16

Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services

This chapter provides an overview of the Oracle WebCenter integration, and discusses how to:

- Configure Oracle WebCenter options.
- Synchronize Oracle WebCenter and PeopleSoft application security.
- Manage WebCenter links.

Understanding the Oracle WebCenter Integration

Oracle WebCenter is a collaborative portal environment that provides an infrastructure for enterprise-wide collaboration. Oracle WebCenter Spaces and Oracle WebCenter Services provide a comprehensive set of standards-based components that enrich your PeopleSoft applications with the most complete and open set of Enterprise 2.0 capabilities in the industry.

Oracle WebCenter Spaces allows users to work more effectively with project teams and work groups, including teams that span multiple geographies and include external members. Within the context of a space, Oracle WebCenter Services provide the capability to include wikis, blogs, communications, content collaboration and social networks. PeopleTools, by leveraging the PeopleSoft Related Content Framework, enables you to integrate WebCenter group spaces and many of the accompanying features with PeopleSoft transaction pages. The following WebCenter related content services are standard with your PeopleTools installation and are available for use with any PeopleSoft transaction page:

- Blogs
- Discussions
- Documents
- Links
- Wikis

Understanding WebCenter Spaces

Within any company, large groups or departments often must create smaller groups to focus on a particular goal, project, or even topic. To help these groups organize themselves, Oracle WebCenter uses the concept of spaces as the nexus for bringing content together. Spaces provide a meaningful way to manage all project details involving any group of users. The WebCenter Spaces application creates spaces to help PeopleSoft application users share information and interact in a collaborative setting.

Templates

Oracle WebCenter Spaces application provides templates for building spaces that support different types of endeavors. When a user accesses a transaction page for the first time, WebCenter creates a new group space for the page or its transaction instance. The template specifications determine the services that appear by default in the new group space. You can install and use the PeopleSoft template that your PeopleTools installation includes, you can use an out-of-the-box WebCenter template, or create and install your own custom template using an existing group space as the basis for the template.

When you configure the WebCenter options in your PeopleSoft applications, PeopleTools requires that you specify the default template for use with new PeopleSoft transaction group spaces. PeopleTools uses web services to retrieve group space template information from the WebCenter server, and then display the group space services in the bottom related content frame.

Understanding WebCenter Services

Oracle WebCenter exposes social networking and shared productivity features through a comprehensive set of services. The services are integrated so that you can use them together. For example, you can add a discussion forum to enable shared discussion, upload documents for shared review, and a blog to keep your personal notes regarding a particular PeopleSoft transaction. You can associate these Oracle WebCenter services with PeopleSoft application pages through the Manage Related Content component:

- Oracle WebCenter Blogs service
- Oracle WebCenter Discussions service
- Oracle WebCenter Documents service
- Oracle WebCenter Links service
- Oracle WebCenter Wikis service

While the related content service definitions are provided in PeopleTools, an active Oracle WebCenter database is required to provide the services and to store the related data.

Important! With the exception of the links service, each WebCenter service uses a set of corresponding WebCenter pages for managing the elements of the group space. The WebCenter application, not PeopleTools, controls the behavior of these pages.

See *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter*,
http://download.oracle.com/docs/cd/E21764_01/webcenter.1111/e12405/wcadm_getstarted_app.htm.

Oracle WebCenter Blogs Service

The Blogs service enables users to create instance specific or shared WebCenter blogs. When users access the service for the first time from a transaction page, WebCenter creates a new group space and associates it with that transaction page. Blog posts to the group space become shared blogs of that transaction page, which means the blogs appear for all transaction instances of the transaction page. When you specify one or more transaction key fields, WebCenter creates an additional group space and associates it with the transaction instance. Blog posts of the group space appear when the user selects the specific transaction instance on the transaction page.

This service supports Related Content Security only. The permissions that the service creates for the group spaces are set equivalent to the content reference for the associated transaction page. All user roles with access to the content reference become user group members and are granted the contributor privilege to the group spaces. If you specify a default moderator role on the WebCenter Options page, a moderator is also added.

When the transaction page permissions change, you should run the WebCenter security sync batch process to synchronize content reference permissions to the group spaces.

Important! Currently, the service does not support transaction instance-level data security.

Oracle WebCenter Discussion Service

The Discussions service lets you expose discussion forums on your application pages, so users can create forums, post questions, and search for answers. For example, customers can share product reviews, or a customer service department can answer questions online. Discussion forums additionally provide the means to preserve and revisit discussions.

The Discussion service enables users to create instance specific or shared WebCenter discussions. When users access the service for the first time from a transaction page, WebCenter creates a new group space and associates it with that transaction page. Discussions in the group space become shared discussions of that transaction page, which means the discussions appear for all transaction instances of the transaction page. When you specify one or more transaction key fields, WebCenter creates an additional group space and associates it with the transaction instance. Discussions posts of the group space appear when the user selects the specific transaction instance on the transaction page.

This service supports Related Content Security only. The permissions that the service creates for the group spaces are set equivalent to the content reference for the associated transaction page. All user roles with access to the content reference become user group members and are granted the contributor privilege to the group spaces. If you specify a default moderator role on the WebCenter Options page, a moderator is also added.

When the transaction page permissions change, you should run the WebCenter security sync batch process to synchronize content reference permissions to the group spaces.

Important! Currently, the service does not support transaction instance-level data security.

Oracle WebCenter Documents Service

The Documents service provides features for accessing, adding, and managing folders and files; configuring file and folder properties; and searching file and folder content in the connected content repositories. When you set up a content repository connection to integrate the Documents service, you can add content to the transaction pages of the PeopleSoft application.

The Documents service enables users to create instance specific or shared WebCenter discussions. When users access the service for the first time from a transaction page, WebCenter creates a new group space and associates it with that transaction page. Documents in the group space become shared documents of that transaction page, which means the documents appear for all transaction instances of the transaction page.

When you specify one or more transaction key fields, WebCenter creates an additional group space and associates it with the transaction instance. Documents in of the group space appear when the user selects the specific transaction instance on the transaction page.

The Documents service uses corresponding WebCenter pages for actions such as, uploading new documents, listing and searching documents in the group space, and managing documents. The WebCenter application, not PeopleTools, controls the behavior of these pages.

This service supports Related Content Security only. The permissions that the service creates for the group spaces are set equivalent to the content reference for the associated transaction page. All user roles with access to the content reference become user group members and are granted the contributor privilege to the group spaces. WebCenter also adds a moderator user group member, if you specify a default moderator role on the WebCenter Options page.

When the transaction page permissions change, you should run the WebCenter security sync batch process to synchronize content reference permissions to the group spaces.

Important! Currently, the service does not support transaction instance-level data security.

Oracle WebCenter Links Service

The Links service provides a way to view, access, and associate related information. For example, in a list of project assignments, you can link to the specifications relevant to each assignment. In a discussion thread about a problem with a particular task, you can link to a document that provides a detailed description of how to perform that task.

Unlike other WebCenter related content services, the Links service is a PeopleSoft component-based page. The Links service enables users to search and create links to existing WebCenter group spaces. The service also enables users to add links to external web pages. When you click the links that appear on the service page, the target opens in a new browser window.

The Links service enables users to create instance-specific or shared WebCenter links to resources.

This service supports Related Content Security only. When the transaction page or instance includes other WebCenter related content services such as the WebCenter Document service or the WebCenter Discussion service, the group spaces created for those services appear as related links by default; you cannot remove these links. Conversely, the service does not alter user access to a linked service. When one user does not have access to a group space that is the target of a link added by other users, the group space link does not appear to the unauthorized user.

Note. For this service, related content security governs whether links appear. The service does not control the security of the target content.

Oracle WebCenter Wiki Service

A wiki is a collection of useful content or information that users can browse, update, and remove, sometimes without the need for registration. This ease of interaction and the variety of operations make wiki an effective tool for collaborative authoring, where multiple people create written content together.

The wiki service enables users to create instance specific or shared WebCenter wiki. When users access the service for the first time from a transaction page, WebCenter creates a new group space and associates it with that transaction page. Wikis in the group space become shared wikis of that transaction page, which means the wiki page appear for all transaction instances of the transaction page.

When you specify one or more transaction key fields, WebCenter creates an additional group space and associates it with the transaction instance. Wikis in the group space appear when the user selects the specific transaction instance on the transaction page.

This service supports Related Content Security only. The permissions that the service creates for the group spaces are set equivalent to the content reference for the associated transaction page. All user roles with access to the content reference become user group members and are granted the contributor privilege to the group spaces. WebCenter also adds a moderator user group member, if you specify a default moderator role on the WebCenter Options page.

When the transaction page permissions change, you should run the WebCenter security sync batch process to synchronize content reference permissions to the group spaces.

Important! Currently, the service does not support transaction instance level data security.

Understanding Security Synchronization

To function correctly, the permissions of the group space must be the same as the associated content reference object of the transaction page. Your PeopleSoft applications use the WebCenter *SpaceWebService* to create and maintain group spaces security that is synchronized to PeopleSoft security. You must configure the web service security for the WEBCENTER node before using any of the WebCenter related content services. Currently, SAML token web service security is the only security type that the WebCenter application supports to consume WebCenter web services.

When a WebCenter service creates a groups space for a transaction page or its transaction instances, all user roles with access to the content reference object are added as user group members with the contributor privilege. WebCenter supports transaction data security at the role level only. Any user who has access to a transaction page is granted access to all group spaces that are created for that page or its transaction instances. If permissions change for a transaction page, the same changes must be reflected in the WebCenter permissions. You synchronize security by running the WebCenter Security batch process.

WebCenter Security Batch Process

The WebCenter Security sync batch process maintains security synchronization between PeopleSoft transactions and WebCenter services.

Important! You should run this process when the permissions of any transaction page that uses one of the WebCenter related content services changes.

The WebCenter Security synchronization process:

1. Retrieves a list of expected user group members of the group space.

This list consists of the roles that have access based on the permission lists that appear on the Security tab of the content reference object for the transaction page.

2. Adds the user roles that do not exist in the list.

3. Adds a moderator user group member, if you specify a default moderator role on the WebCenter Options page.
4. Removes user group members that do not appear in the list.
5. Removes individual user members that do not have one of the listed user roles.

Note these points about the security synchronization process:

- The batch process should be run only once after a transaction page permissions are changed to update permissions of existing group spaces.
- The batch process updates only those group spaces that the WebCenter related content services created.
- The batch process does *not* change the permissions of group spaces that are linked to a transaction page or its transaction instances by the WebCenter Links service.

Note. Links do not appear to users if permission to access the associated target is not granted.

Related Language Considerations

Note these points about the use of related language in the WebCenter integration:

- WebCenter and PeopleSoft applications support related language data differently.
- A related language item that you translate into the nonbase language appears to users of all languages.
- You cannot configure a translated item to appear to base language users only, instead of the base language.
- Both the base language and the nonbase item can coexist in the same system.

Important! WebCenter currently does not support Canadian French or UK English as a local language option.

Example

If you have an application that uses English as the base language and French as a related language:

- If a French-speaking user enters a related language item, then it appears to users of all languages.
- An English-speaking user can not provide an English translation of that item to show to English users only, instead of the French version.
- Both English and French items can appear simultaneously in the same system.

Prerequisites

These tables discuss the prerequisite steps to configuring Oracle WebCenter options in the PeopleSoft application.

Perform these tasks in the WebCenter application.

Prerequisite	Reference
Install Oracle WebCenter.	See <i>Oracle Fusion Middleware User's Guide for Oracle WebCenter Spaces 11g Release 1 (11.1.1.5.0)</i> Importing Space Templates, http://download.oracle.com/docs/cd/E21764_01/install.1111/e12001/toc.htm .
Import the PeopleSoft template into the WebCenter application.	See <i>Oracle Fusion Middleware User's Guide for Oracle WebCenter Spaces 11g Release 1 (11.1.1.5.0)</i> , Importing Space Templates http://download.oracle.com/docs/cd/E21764_01/webcenter.1111/e10149/maint_mng_temps.htm#BABIDFJE .
Enable locale switching.	See Appendix I, "Enabling Locale Switching Through Query Parameters in Oracle WebCenter," page 741.
Set the frame busting context parameter to <i>never</i> . Note. You set the parameter to <i>never</i> because the WebCenter pages appear as related content in the lower Related Content frame, which is an iframe.	See <i>Oracle Fusion Middleware Web User Interface Developer's Guide for Oracle Application Development Framework 11g Release 1 (11.1.1.5.0)</i> , Framebusting http://download.oracle.com/docs/cd/E21764_01/web.1111/b31973/ap_config.htm#ADFUI11154 .
Modify the configuration of the PeopleSoft template to fit your business needs. (Optional)	See <i>Oracle Fusion Middleware User's Guide for Oracle WebCenter Spaces 11g Release 1 (11.1.1.5.0)</i> , Working with Space Templates http://download.oracle.com/docs/cd/E21764_01/webcenter.1111/e10149/gspace_build.htm#WCSUG1265 .
Create new templates. (Optional) Important! When you create your own template, you must enable the Discussions and Documents services and include the default Wiki and Blog pages in your template. WebCenter provides no process or program to synchronize any template modifications; if you make more changes to the default template after your initial modifications, then you must recreate the custom template from the delivered template.	See <i>Oracle Fusion Middleware User's Guide for Oracle WebCenter Spaces 11g Release 1 (11.1.1.5.0)</i> , Working with Space Templates http://download.oracle.com/docs/cd/E21764_01/webcenter.1111/e10149/gspace_build.htm#WCSUG1265 .

Perform these tasks in the PeopleSoft application.

Configure the Integration Gateway.	See <i>PeopleTools 8.52: PeopleSoft Integration Broker Administration</i> , "Managing Integration Gateways," Administering Integration Gateways.
Configure single sign on.	See <i>PeopleTools 8.52: Security Administration</i> , "Implementing Single Signon."
Access the WEBCENTER node and enter the WebCenter server address in the Content URI Text field.	See Chapter 9, "Configuring the Portal Environment," Setting Portal Nodes , page 279.

Access the WEBCENTER node and configure WS Security so that the node can consume the WebCenter web services.

Note. Currently, only *SAML Token* is supported by WebCenter as the authentication token type.

See [Chapter 19, "Configuring WS-Security for WSRP Consumption and Production," SAML Token Security, page 654.](#)

See Also

PeopleTools 8.52: PeopleSoft Integration Broker Administration, "Using the Integration Network"

PeopleTools 8.52: Security Administration, "Employing Signon PeopleCode and User Exits"

Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter 11g Release 1 (11.1.1.5.0)
http://download.oracle.com/docs/cd/E21764_01/webcenter.1111/e12405/toc.htm

Configuring Oracle WebCenter Options

This section discusses how to configure WebCenter options:

Page Used to Configure Oracle WebCenter Options

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
WebCenter Options	PTWC_OPTIONS	PeopleTools, Portal, WebCenter Integration, WebCenter Options	Set the WebCenter integration options.


Setting the WebCenter Integration Options


Access the WebCenter Options page (Select PeopleTools, Portal, WebCenter Integration, WebCenter Options).




WebCenter Options

Specify the WebCenter integration options.

Node Name: WEBCENTER

Default Moderator: 

***Default Template:** 

Privilege Role Mappings	
Privilege Name	*Group Space Role
Moderator	<input type="text" value="Moderator"/> 
Contributor	<input type="text" value="Participant"/> 
Viewer	<input type="text" value="Viewer"/> 

WebCenter Options pages

WebCenter Options

Node Name	Displays the name for the WebCenter application default node, which is <i>WEBCENTER</i> .
Default Moderator	Specify the user role to be added as a user group member with the moderator privilege to any group spaces created by WebCenter related content services.
Default Template	Select the template that the WebCenter related content service uses to create new group spaces for PeopleSoft transaction pages. All public group space templates that you have set up in WebCenter appear in the drop-down list. The default value is <i>PeopleSoftTemplate</i> . This field requires a value.

Privilege Role Mappings

Use the fields in the Privilege Role Mappings grid to correlate the PeopleSoft privilege to the WebCenter group space role. This grid appears when you select a valid default group space template. All fields in this grid are required when you select a default group space template.

Privilege Name	Displays the PeopleSoft privilege.
-----------------------	------------------------------------

Group Space Role

Select from the list of privilege roles that exist in the specified default template. If you select a new default template, the values in drop-down list change to reflect the new default.

Select from these default values:

- *Moderator*
- *Participant*
- *Viewer*

This field requires a value.

Synchronizing Oracle WebCenter and PeopleSoft Application Security

This section discusses how to run the WebCenter Security Sync batch process.

Page Used to Synchronize Oracle WebCenter and PeopleSoft Application Security

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
WebCenter Security Synchronization	PTWC_SECURITY_SYNC	PeopleTools, Portal, WebCenter Integration, WebCenter Security Sync	Run the WebCenter Security Synchronization batch process.

Running the WebCenter Security Sync Batch Process

Access the WebCenter Security Synchronization page. (Select PeopleTools, Portal, WebCenter Integration, WebCenter Security Sync).

WebCenter Security Synchronization

This process will sync the group space permission with the transaction page it is associated with. Only group spaces created for a transaction page or its transaction instances by one of the WebCenter related content services will be updated.

Run Control ID: Daily at 12pm

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

WebCenter Security Synchronization page

To run the WebCenter Security Synchronization batch process:

1. Click the Run button.
2. Click the Process Monitor link to verify success.

Managing WebCenter Links

This section describes how to:

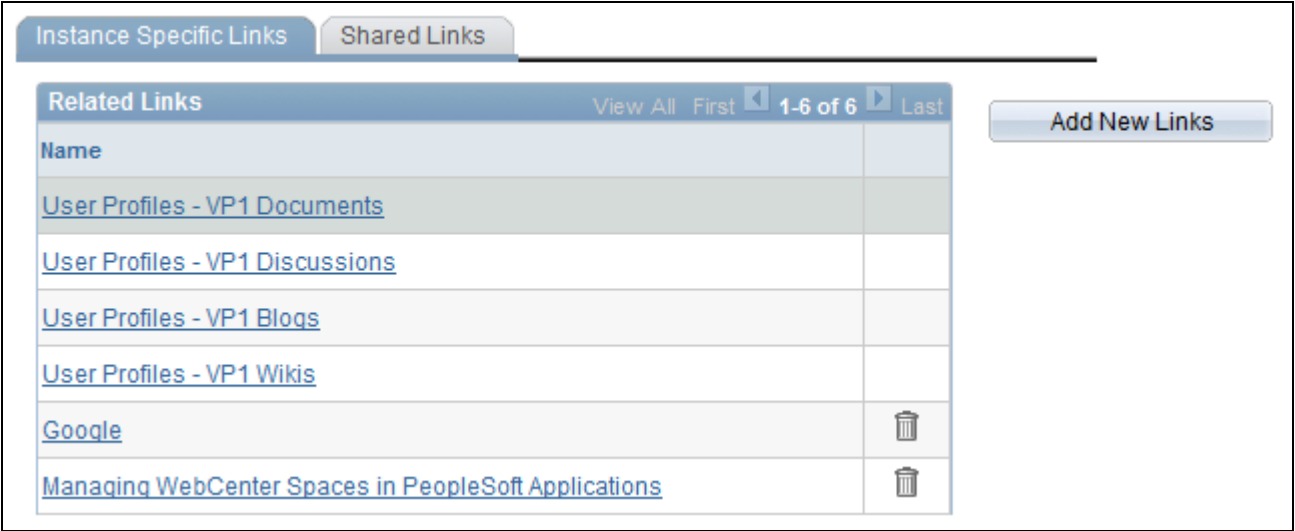
- Manage instance-specific links.
- Manage shared links.
- Add group space links.
- Add external URL links.

Pages Used to Manage WebCenter Links

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Instance Specific Links	PTWC_LINKS_INST	Click WebCenter Links in the Related Content menu.	Manage instance-specific links.
Shared Links	PTWC_LINKS_SHARED	Select the Shared Links tab.	Manage shared links.
Add New Links	PTWC_LINKS_ADD	Click the Add New Links button.	<ul style="list-style-type: none"> • Add group space links. • Add external URL links.

Managing Instance-Specific Links

Access the Instance Specific Links page (Select the Instance Specific Links tab).



Instance Specific Links page

To add new instance-specific links, click the Add New Links button. The Add New Links page appears.

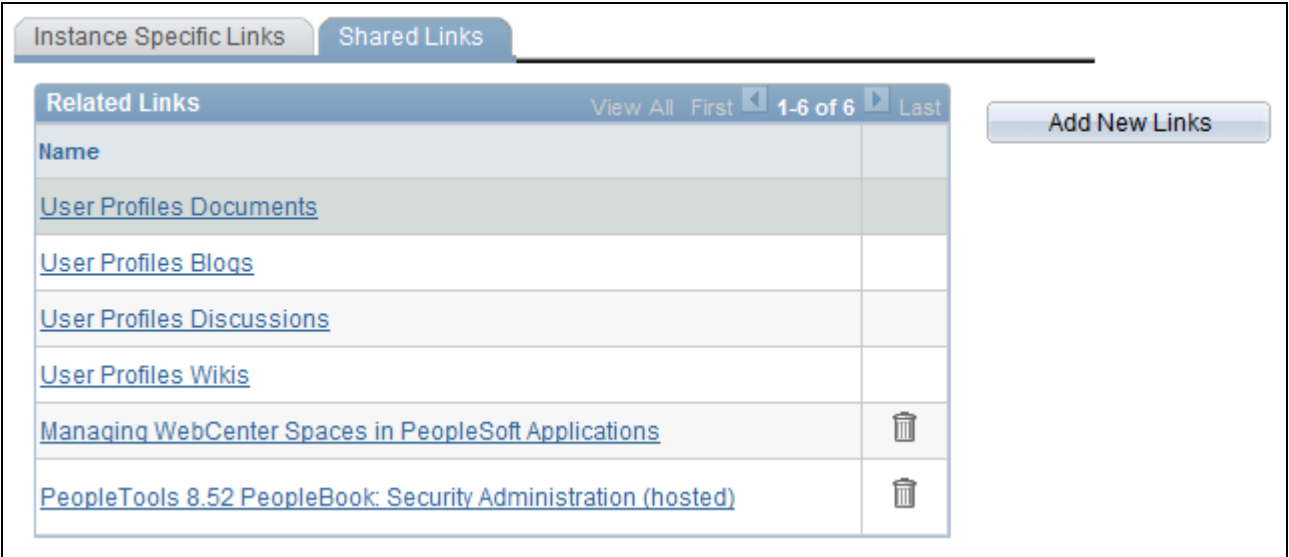
See Chapter 16, "Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services," Adding Group Space Links, page 611 and Chapter 16, "Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services," Adding External URL Links, page 612.

To delete instance-specific links, click the Delete Link(trash can) icon.

Note. Links to services within the current group space appear automatically and cannot be removed.

Managing Shared Links

Access the Shared Links page (Select the Shared Links tab).



Shared Links page

To add new shared links, click the **Add New Links** button. The Add New Links page appears.

See [Chapter 16, "Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services," Adding Group Space Links, page 611](#) and [Chapter 16, "Configuring PeopleSoft Applications to Use Oracle WebCenter Related Content Services," Adding External URL Links, page 612](#).

To delete shared links, click the Delete Link(trash can) icon.

Note. Links to services within the current group space appear automatically and cannot be removed.

Adding Group Space Links

Access the Add New Links page (Click the Add New Links button).

Add New Links page

To add new group space links:

1. Select *Group Space Link* as the content type.
2. To search the group spaces that are available to you in your WebCenter application, enter keywords and click the **Search** button.

Note. Keywords for each group space are managed by the WebCenter administrator in the WebCenter application

3. Select the group spaces for which you want to add links.

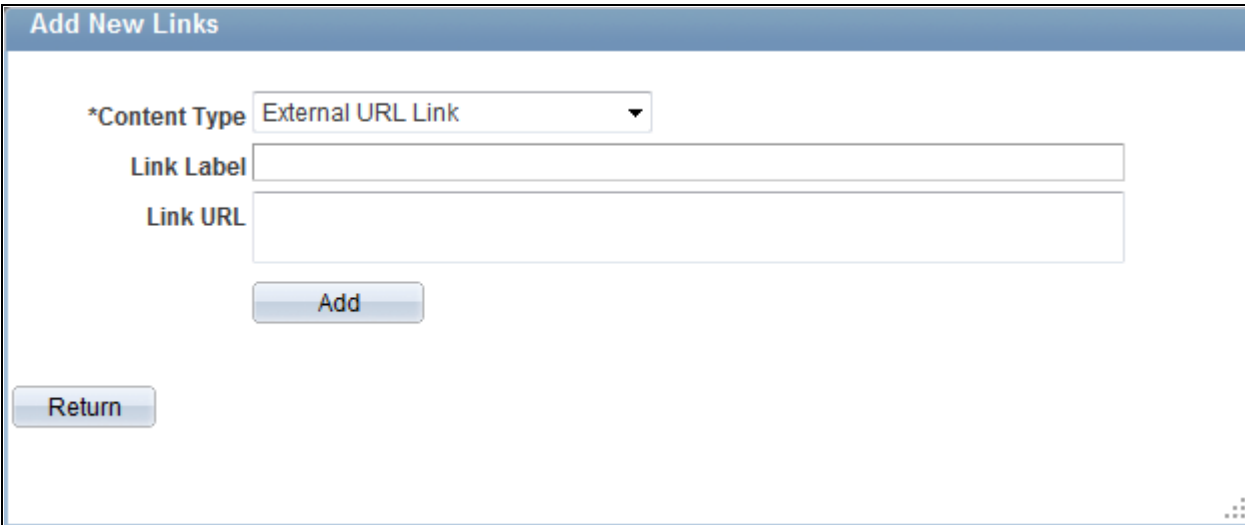
4. Click the Add Selected Group Spaces button.

The application returns you to the WebCenter Links page.

Click the Return button to go back to the previous page.

Adding External URL Links

Access the Add New Links page (Click the Add New Links button).



Add New Links page

To add new external URL links:

1. Select *External URL Link* as the content type.
2. Enter the label that you want to appear in the Related Links grid on the WebCenter Links page.
3. Enter the target address of the external link in the Link URL field. Use this example as a model:
https://support.oracle.com.
4. Click the Add button.

The application returns you to the WebCenter Links page.

Click the Return button to go back to the previous page without saving your entry.

Chapter 17

Creating and Configuring WorkCenter Pages

This chapter provides overviews of WorkCenter pages, iScript and component-based pagelets, and the WorkCenter Page Development and Implementation process and discusses how to:

- Define page layout templates.
- Manage WorkCenter page configurations.
- Set pagelet area user preferences.

Understanding WorkCenter Pages

WorkCenter pages are a type of configurable PeopleSoft page that enable you to offer access to related transactions, analytics data, query results, and other PeopleSoft resources from one central location. You can organize the elements on WorkCenter pages based on processes, procedures, roles, or other business requirements that are specific to your organization. When carefully designed, WorkCenter pages bring together the PeopleSoft elements that your users access most when performing the tasks that compose their primary responsibilities. As a result, WorkCenter pages promote productivity by minimizing the time that users spend navigating between tasks.

WorkCenter pages can bring these different elements together because WorkCenter pages use a portal template that provides flexibility in the way that you access components and pages and also in the way the application displays these elements. The WorkCenter portal template differs from the standard portal template in two primary ways:

- The WorkCenter portal template partitions the browser window into an adjustable pagelet area and target content area.
- The two areas of the WorkCenter portal template are designed to interact by passing contextual information back and forth by using an event driven model. The actions in one area influence the content in the other.

When users access WorkCenter pages, they are presented with pagelets that contain links to components, analytics, query results, and other PeopleSoft resources. The WorkCenter pagelet area is designed to extract and leverage values that are necessary to complete the transaction in the target area. When users click a link, it triggers the appropriate transaction, starts a search process, and passes values from the pagelet area to the search process so that the application can display the appropriate content in the target content area or in a new browser window.

See Also

[Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Template Pagelets, page 333](#)

[Chapter 7, "Working with Portal Templates," Developing Portal Templates, page 216](#)

The Pagelet Area

The WorkCenter pagelet area consists of three parts:

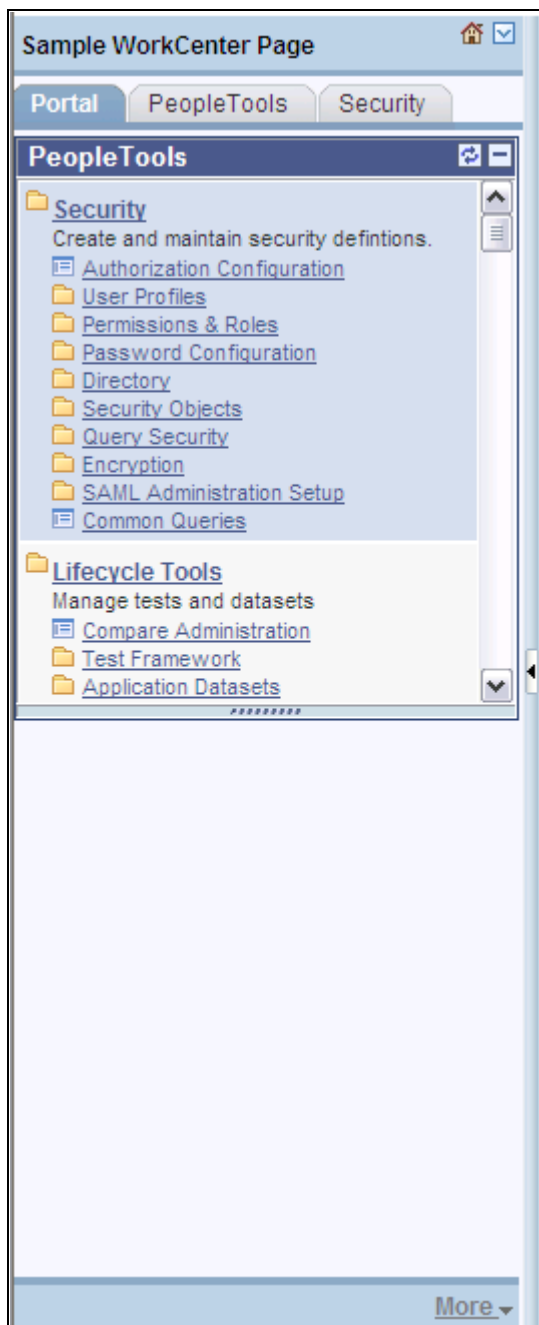
- The header.
- The pagelet area.
- The footer.

In the header, the title, the Home button, and the pagelet area controls drop-down menu are visible. You click the Home button to return to the WorkCenter starting page. You click the pagelet area controls drop-down menu to access up to three links that enable you to reload, personalize, and configure the WorkCenter page.

The pagelets that you assign to the WorkCenter page appear in the main body of the pagelet area. Each pagelet you assign to the area is a self contained unit that you control independently of other pagelets. Pagelets appear one above the next either in an expanded or collapsed state based on your configuration. Expanded pagelets divide the available space evenly until the user resizes a pagelet by dragging the pagelet handle. The system automatically adjusts the size of remaining pagelets to fill the available space.

At the bottom of the pagelet area, the More drop-down menu appears in the footer. You can configure the WorkCenter page to *hide* any pagelet that is minimized instead of collapsing the pagelet into its header. The hidden pagelet is minimized to the More menu. When you click the drop-down button, the titles of the minimized pagelets appear in a list. Select a pagelet from the list and the systems restores the pagelet to its original size and positions it at the bottom of any other expanded pagelets in the pagelet area.

The pagelet area is adjustable in width and its configuration is flexible. You can organize the pagelet area by creating pagelet groups. Each group appears as a separate page. If you configure multiple pagelet groups, a tab appears at the top of the each page to identify the pagelet group and provide a familiar vehicle for accessing the pagelets. You can define options such as the initial size of the pagelet area, the pagelet group order, and the pagelet group that appears by default when you access a WorkCenter page. This sample WorkCenter pagelet area contains three pagelet groups:



Pagelet area showing tabbed pagelet groups

After creating pagelet groups, you assign pagelets to the groups. You specify options such as the order of the pagelets, which pagelet receives focus by default, whether the pagelet is collapsed or expanded as its initial state, and the maximum number of pagelets that remain visible when in a minimized state. The pagelet area also supports user personalizations to the extent that you enable them.

Generally, pagelets that you assign to WorkCenter pages are narrow format pagelets that are specifically designed to appear in the resizable pagelet area. The pagelet area supports homepage and template pagelets that are registered in either the Pagelets or the Template Pagelets folder in the portal registry. You access these folders by selecting PeopleTools, Portal, Structure & Content and then clicking the Portal Objects link.

The pagelet area supports iScript-based pagelets, component-based pagelets, and even external URL-based pagelets. You can use Pagelet Wizard to create and publish iScript-based template and homepage pagelets such as those that use navigation collections and queries as data sources. You can also manually create and register component-based or other pagelets.

See Also

Chapter 12, "Developing and Registering Pagelets," Understanding Pagelet Development, page 457

Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," page 323

The Target Content Area

The primary purpose of the target content area is to display the transaction page, query results, analytics, or other resources that the application user requests as the result of clicking a link in the pagelet area. The target content area can display any type of content that a standard PeopleSoft transaction page can display. You use this area to search, view, edit, and complete your transactions. You can also set up and configure pages and pagelets to use interwindow communication, which enables pagelets to communicate in any direction: from pagelet area to target content area, from target content area to pagelet area, and even between elements that occupy the same section.

You can target resources to appear not only in the target content area, but to a new browser window or to a current browser replacing the entire WorkCenter page with the resource.

Example of Setting the Target to Open in a New Window

You can set the target to open in a new window when you need to view both the current page and the new page simultaneously. For example, an accounts payable representative is completing an invoice and needs to view customer history at the same time. You could design a pagelet so that the link opens the Customer History page in a new window.

Example of Setting the Target to Replace the Current Window

You can set the target to replace the current window entirely. For example, an account representative is reviewing her largest account and then wants to view a dashboard that shows all accounts in relation to one another. You could design a pagelet with a link that opens the Accounts Dashboard and replaces the current window.

Page Layout Templates

WorkCenter pages use an iframe-based portal template that supports four sections:

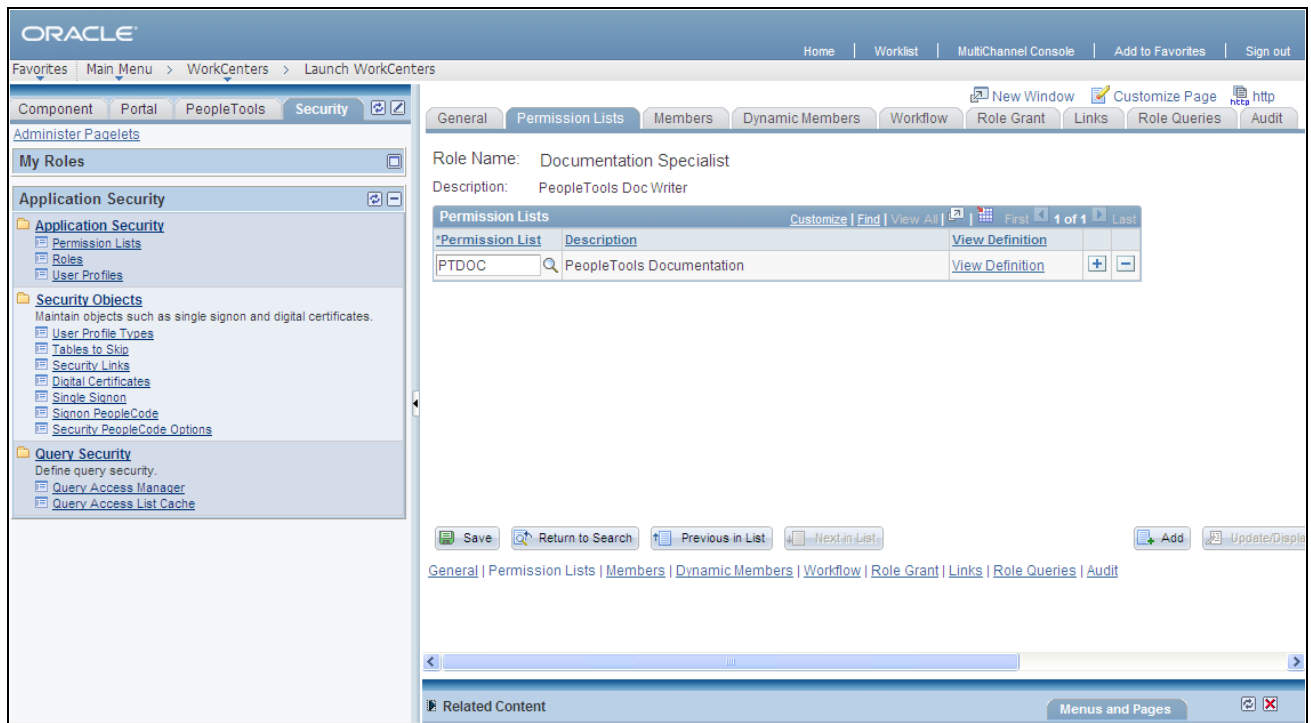
- The portal header and drop-down style menu.
- The pagelet area.
- The target content area.
- The related content frames.

The portal header and drop-down style menu, when enabled, appear at the top of the page above the pagelet and target content areas. The adjustable pagelet and target content areas occupy the middle of the page. The related content frames appear either below or beside the target content area when the current transaction page has related content configured.

You can change the layout of WorkCenter pages by using page layout templates. Page layout templates use Application class PeopleCode to determine the location of the pagelet area and target content area. WorkCenter pages deliver application class PeopleCode that renders the pagelet area on either the left side or the right side of the page.

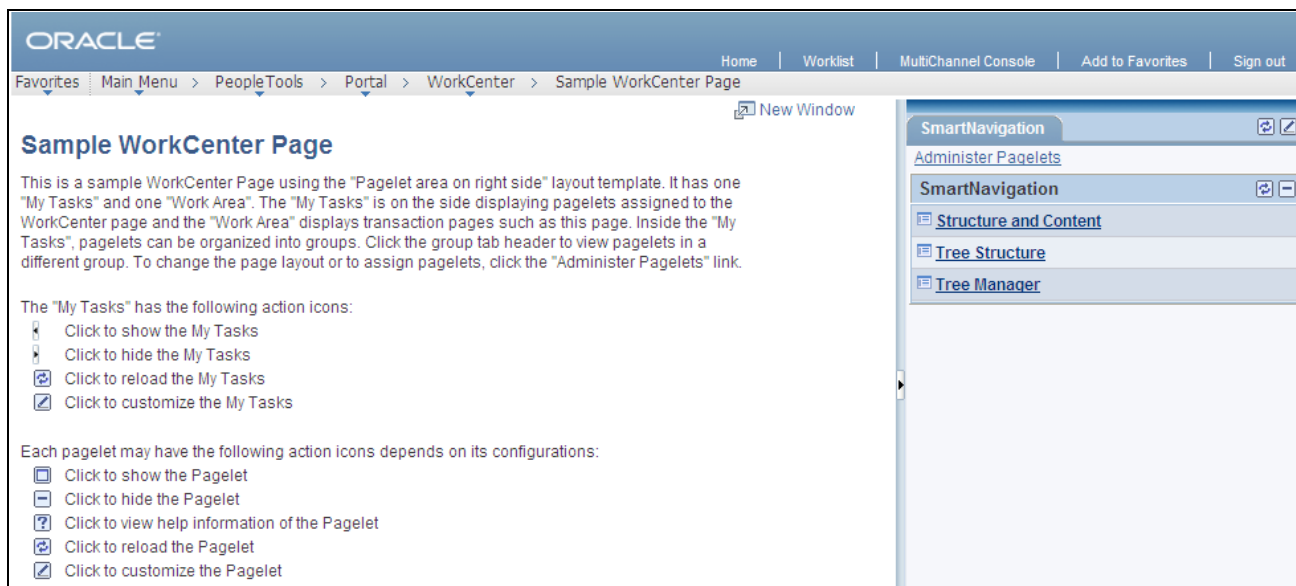
Page Layout Template Examples

In this example, which uses the default page layout template, you see the drop-down menu, the pagelet area on the left, and the related content frame beneath the target content area on the right:



Sample WorkCenter page showing related content frame

In this example, you see the pagelet area on the right side of the WorkCenter page:



Sample WorkCenter page that uses a page layout template that places the pagelet area on the right side of the page

Page Design and Processing Considerations for WorkCenter Pages

WorkCenter pages have different characteristics than standard PeopleSoft components and pages. This section discusses design and processing differences that you should consider when you work with WorkCenter pages and pagelets.

Page Design Considerations

When you create a WorkCenter page, consider the following differences and page design recommendations, which accommodate those differences:

- For WorkCenter pages, the initial page in the target content area is determined by the menu item associated with the content reference that you select when you create a new WorkCenter page.

While the page that appears initially in the target content area can be any page, you might want to start with an informational page to acquaint users with the WorkCenter page functionality and layout. You can use the sample WorkCenter page that comes with all PeopleSoft applications, or create a page that you design specifically for use as a WorkCenter page.

- WorkCenter pages should not generate a search page or start the search process when the user clicks the content reference to access the page. The user should immediately see the initial page.

Select a "dummy" record—one that has no keys or contains only one row of data—as the component search record on the Use tab.

The INSTALLATION record is an example of a "dummy" search record.

- WorkCenter pages generally require no save functionality.

Select the Disable Saving Page check box on the Use tab.

- Typically, the component should consist of only one page, but no tabs or page links should appear on the page.

Deselect the Multi-Page Navigation check boxes on the Internet tab.

- The elements that make up the tool bar and page bar are not applicable and should not appear on the WorkCenter page.

Select the Disable Toolbar and Disable Pagebar check boxes on the Internet tab.

Processing Considerations

Note these points about WorkCenter page processing:

- WorkCenter pages, like homepages and dashboards in your PeopleSoft applications, load pagelets in parallel by default. When a WorkCenter page uses parallel loading, each pagelet appears independently from one another. When the data for the pagelet has been loaded, the pagelet appears.

WorkCenter pages and the assigned pagelets, however, perform better when the pagelets use deferred loading. When you configure pagelets to use deferred loading, no pagelet appears until after all data for all pagelets has been loaded. After all pagelet data has loaded, then the system displays the page and the assigned pagelets simultaneously as one unit.

- WorkCenter pages and assigned pagelets support interwindow communication. Use the Interwindow Communication page to configure interactivity among pagelets that share a WorkCenter page.

See [Chapter 14, "Configuring Interwindow Communication," page 527](#).

- WorkCenter pages can process contextual parameters that you include in the WorkCenter page URL. The WorkCenter template accesses the parameters and passes them to the pagelets, which use the values to determine the records to retrieve and display. For example, you can create a WorkCenter page with pagelets that display customer information. You can pass the customer ID to the WorkCenter page URL as a context parameter so that all pagelets retrieve and display information about the same customer.

The context parameter, CONTEXTPARAMS, must use the NAME:VALUE format and be passed as a query string. Use this format when you build the WorkCenter page URL:

```
http://appservername/psp/sitename/portalname/nodename
/c/PTAL_ADMIN.PTAL_SAMPLE_PAGE.GBL? CONTEXTPARAMS=USERPROFILE:VP1&
CONTEXTPARAMS=EMPLOYEEID:1242.
```

Using the previous sample URL, the WorkCenter template retrieves the query string parameters, which it then sends to each pagelets in the following query string format ?USERPROFILE=VP1&EMPLOYEEID=1242.

When you develop WorkCenter pagelets, use the **%Request.getParameter()** PeopleCode function to retrieve and make use of the context parameter values.

Understanding iScript and Component-Based Pagelets

You can create iScript-based and component-based homepage pagelets and template pagelets for use with WorkCenter pages. You use Pagelet Wizard to create iScript-based pagelets. Pagelet Wizard pagelets can be based on many possible data sources, such as navigation collections, search records, HTML, and queries. You use PeopleSoftApplication Designer to create component-based pagelets that are based on PeopleTools definitions. This section presents overviews of pagelets based on these PeopleSoft elements:

- Navigation collections.
- PeopleSoft Query.
- Components.

See Also

[Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Understanding Pagelet Wizard Data Sources, page 325](#)

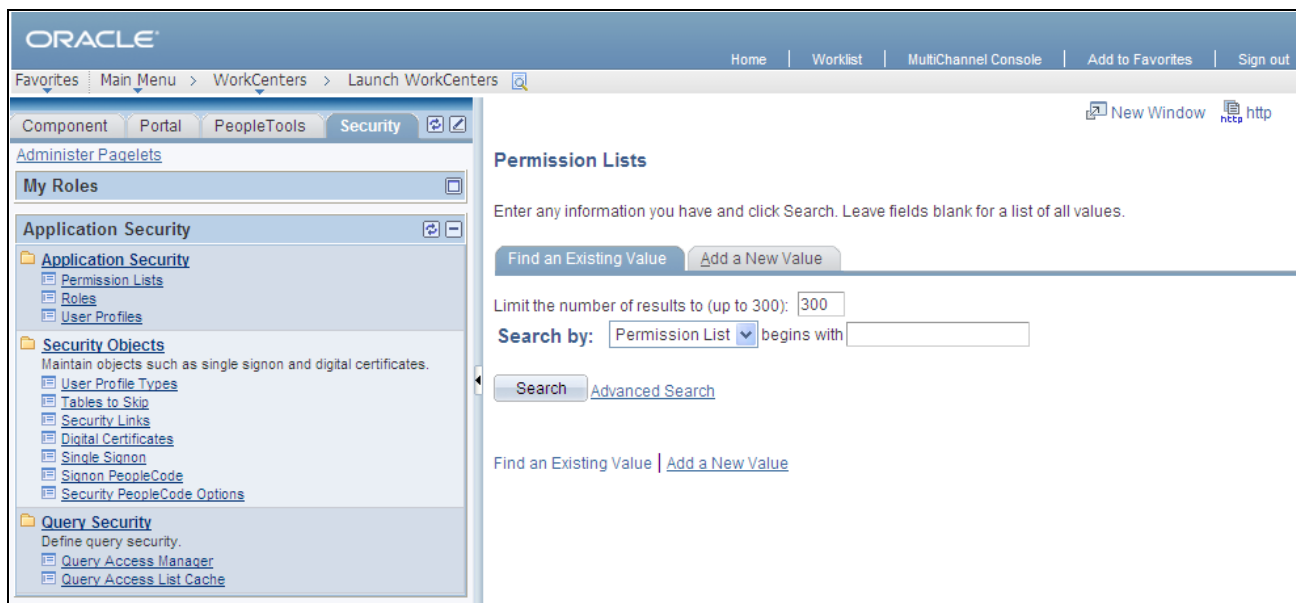
[Chapter 12, "Developing and Registering Pagelets," Developing iScript-Based Pagelets, page 484](#)

Navigation Collection-Based Pagelets

Navigation collections are portal objects that you use to create your own groupings of existing folders and content references, which you can then deploy or publish to different users or groups of users. Navigation collections enable you to create links to content that you already have in the portal registry and to organize the links in ways that specifically address your users' business needs.

Navigation collection-based pagelets appear in the pagelet area as folders and links. When users click a folder, the folder contents appear in the pagelet. When users click a link, the resource opens in the target content area by default. Because navigation collections are designed as alternate menu structures, you navigate through folders to find links, which generally reference components. When users click the link, the component search page appears in the target content area unless the search record is configured so that the search page is bypassed. In such cases, the component requires no search parameters and immediately opens in the target content area.

In this example, you see the navigation collection-based Application Security pagelet in the pagelet area of the WorkCenter page.



Application Security navigation collection-based template pagelet showing the Permission Lists search page in the target content area

Publishing Navigation Collection-Based Pagelets for Use With WorkCenter Pages

You create and maintain navigation collections using the navigation collections component. Using the pages in this component, you set the properties that determine the order in which the folders and links appear, and also the location where the target appears. If you decide to create pagelets based on navigation collections, you should review these properties to determine whether pages will open in the target content area, replace the current page, or in new browser windows.

You can use Pagelet Wizard to create pagelets based on any *published* navigation collection. At step two of the pagelet creation process in Pagelet Wizard, you must specify your navigation collection as the data source. Choose from any of your custom navigation collections or the PeopleSoft-delivered navigation collections. Continue through the process to step 6, where you specify *Homepage Pagelet* or *Template Pagelet* as a publishing option. When you save the pagelet, Pagelet Wizard creates the pagelet and you can then assign it to the pagelet area of a WorkCenter page.

Specific details for creating and publishing navigation collections and using Pagelet Wizard to publish them as pagelets are located in this PeopleBook.

See [Chapter 4, "Working with Navigation Pages," Creating or Editing a Navigation Collection, page 65.](#)

See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Selecting a Navigation Collection Data Source, page 364.](#)

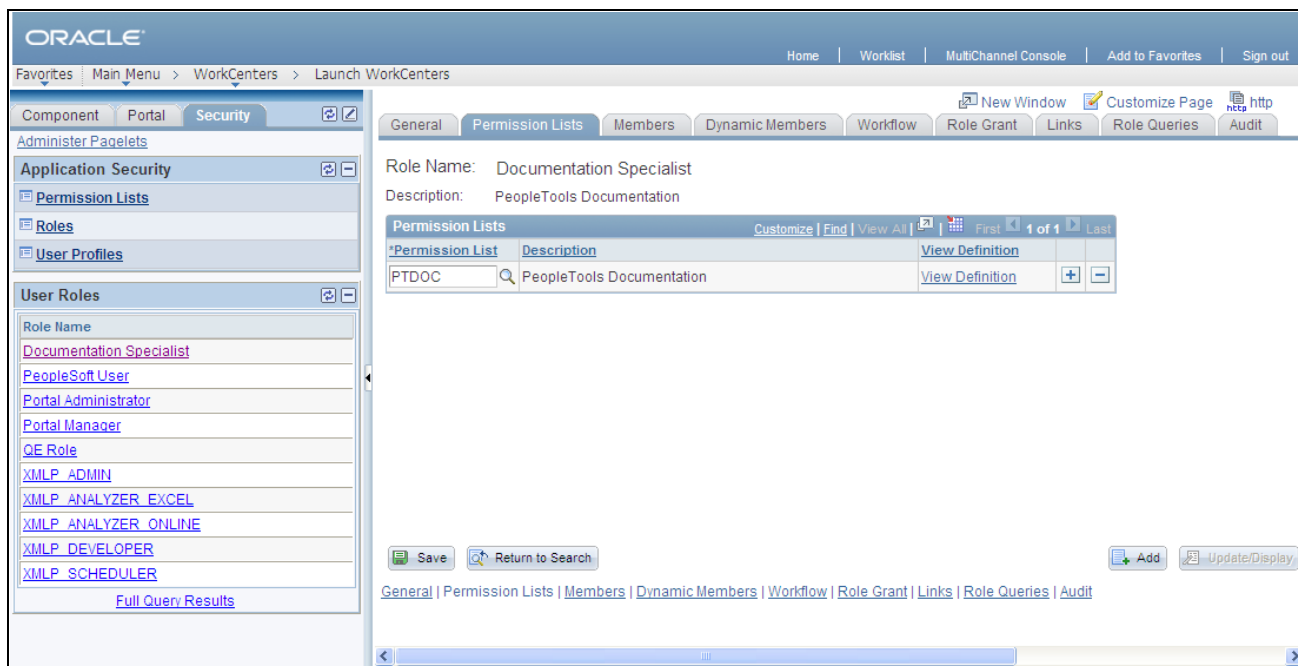
See [Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Step 6: Specifying Pagelet Publishing Options, page 419.](#)

PeopleSoft Query-Based Pagelets

PeopleSoft Query is a reporting tool that enables you to extract your data by using visual representations of your PeopleSoft database, without writing SQL statements. The queries that you create can be as simple or as complex as necessary. You can design queries that return basic lists of static text. You can also use more advanced features, such as *drilling URLs*. By configuring drilling URLs and associating them with query fields, the values that the query retrieves appear as links, which you click to access both internal and external content.

Query-based pagelets can take advantage of queries, whether they are simple or complex. You might base a pagelet on a query that retrieves informational text to display to the user. However, query-based pagelets also have access to the more advanced query metadata, such as the drilling URLs. If you link fields in query-based template pagelets to these drilling URLs, the values that the pagelet retrieves appear as links as they do in PeopleSoft Query. Unlike PeopleSoft Query, however, you can configure the links to open the resource in the target content area of the WorkCenter page.

Because you can configure the links so that the associated resources open in the WorkCenter page target content area, query-based pagelets that are configured to display the retrieved values as active links are ideal for using with WorkCenter pages. In the pagelet area of this WorkCenter page, the User Roles pagelet is based on the *PT_SEC_USER_ROLES* query definition, which returns the roles that are assigned to the current user. Each role name in the list is a link to the Roles component. When users click a role name such as *Documentation Specialist*, that role name value passes to the component as a search parameter. Using the value, the Roles component opens to the *Documentation Specialist* role.



PeopleSoft Query pagelet with drilling URLs

Publishing Query-Based Pagelets for Use With WorkCenter Pages

You create and maintain queries using the Query Manager component.

You can use Pagelet Wizard to create pagelets based on any query. At step two of the template pagelet creation process in Pagelet Wizard, you must specify your query as the data source. At step 5, you use the link icon to access the Link Details page, where you can configure fields to use the drilling URLs and appear as links. At this step, you also specify whether a link appears in the target content area or a new window. Continue through the process to step 6, where you specify *Homepage Pagelet* or *Template Pagelet* as a publishing option. When you save the pagelet, Pagelet Wizard creates the pagelet and you can then assign it to the pagelet area of a WorkCenter page.

Specific details for creating and maintaining queries are located in the PeopleTools 8.52 PeopleBook: *PeopleSoft Query*.

Specific details for using Pagelet Wizard to publish queries as template pagelets are located in this PeopleBook.

See Also

PeopleTools 8.52: PeopleSoft Query, "Working with Advanced Query Options"

Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Selecting a PeopleSoft Query Data Source, page 366

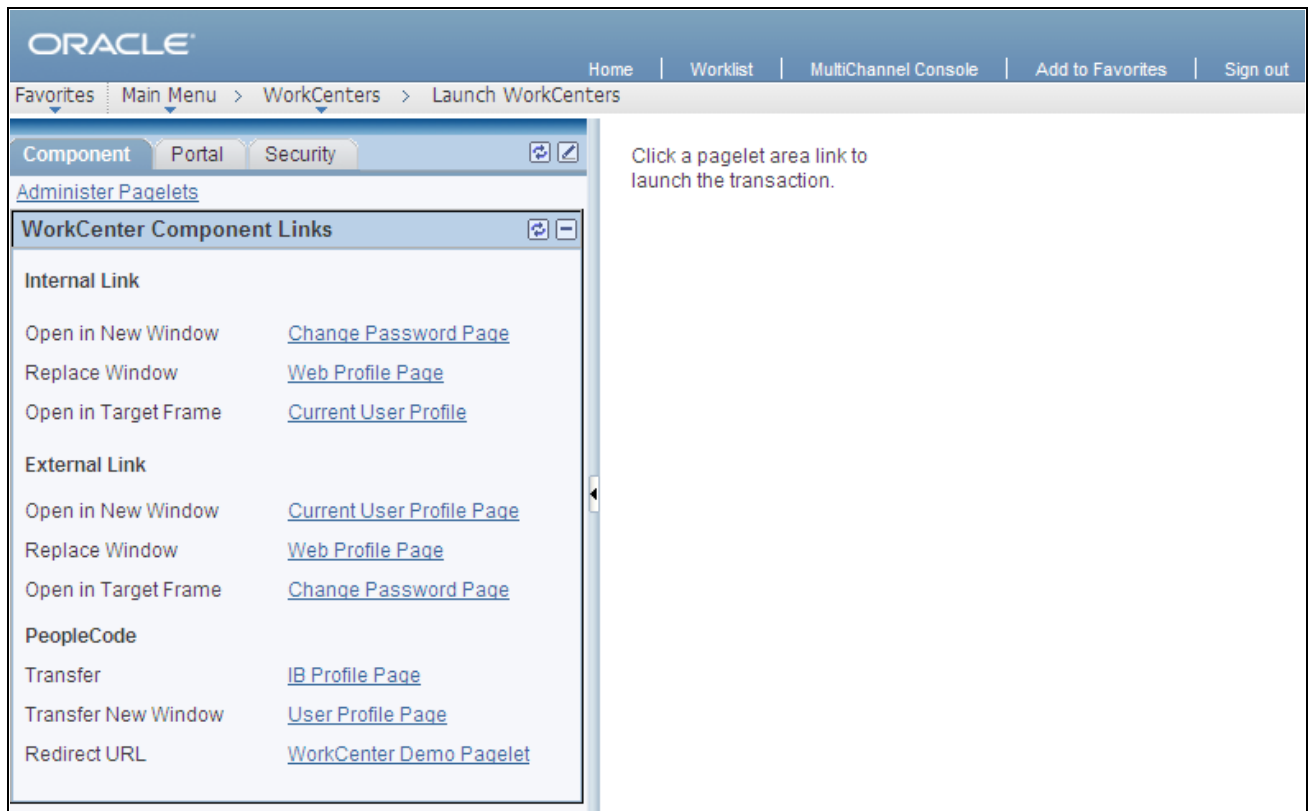
Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Specifying Data Link Details, page 416

Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Step 6: Specifying Pagelet Publishing Options, page 419

Component-Based Pagelets

Component-based pagelets are PeopleSoft component definitions that you create and configure in PeopleSoft Application Designer and then register in the PeopleTools portal as pagelets. With minor differences, the process that you use to create, configure, and assemble the definitions that compose this type of component definition follows closely the application development process. While the *process* is similar to that which you use to create standard component definitions, some definition *properties* are different. In addition to these definition properties, other portal-related differences can exist depending on whether you use an existing component or create a new one.

This section discusses the important differences in settings between standard component definitions and components that you use in component-based pagelets. All examples in this section use the component-based *WorkCenter Component Links* pagelet, which you see here in the pagelet area of the WorkCenter page:



WorkCenter Component Links component-based template pagelet assigned to a WorkCenter page

Record Definition Settings

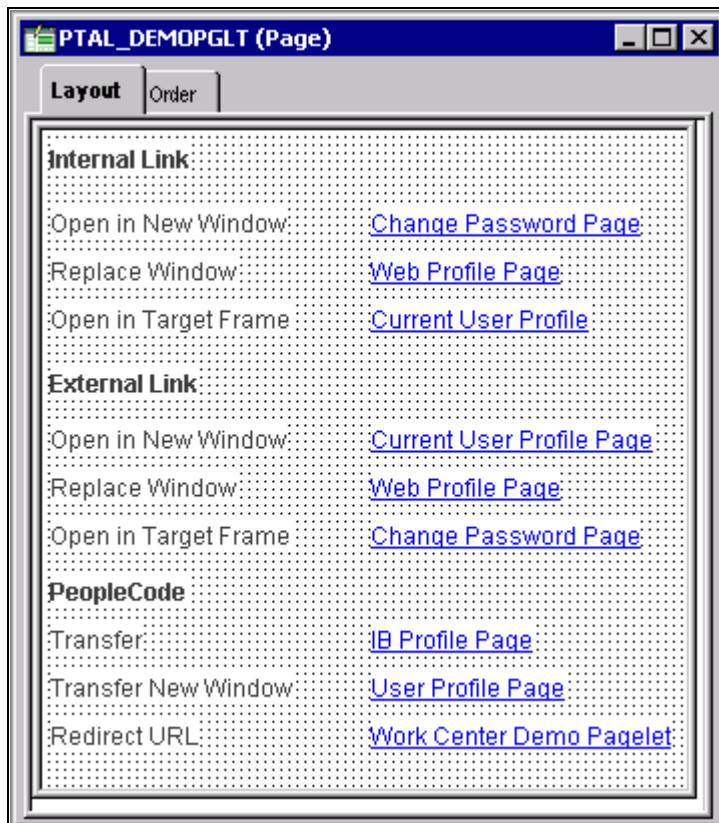
Record definitions that support component-based pagelets typically are not set to the SQL table record definition type because the pagelet area is not intended to write data to tables. Create or use record definitions with a record type of SQL View or Derived/Work.

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Creating Record Definitions," Setting Record Properties.

Page Field Definition Settings

All pagelets should be designed to produce interaction between the pagelet area and the target content area. When defining pages that are part of component-based pagelets, the page developer can most easily facilitate this interaction by inserting push button/hyperlink page controls and then configuring properties that are inherent to that type of control. The developer can also write PeopleCode programs that accomplish the same pagelet area to content area interaction.

The WorkCenter Component Links pagelet is based on the PTAL_DEMOPGLT page definition, which includes nine push button/hyperlink controls:



Page definition that is part of a component-based template pagelet

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Creating Page Definitions."

By setting the push button/hyperlink page control Type and Field Use properties at design time, the developer determines the resource that appears when the user clicks the link and also where that the resource appears, such as in the target content area, a modal window, or a new window. Secondary pages, PeopleCode, internal links and external links only are valid destination types for push button/hyperlink controls when you use them on component-based template pagelets.

The Push Button/Hyperlink Type properties specify an internal link type. The internal link properties specify the location of the CHANGE_PASSWORD page:

Push Button/Hyperlink Properties

Type | Label | Use | General

Type

☐ Push Button ☒ Hyperlink

Destination: Internal Link

Record Name: PTAL_DEMOPGLT

Field Name: PTAL_HYPERLINK1

Style: Use Default Style

External Link

☐ Static ☐ Dynamic

URL ID:

☐ URL Encoded by Application

Internal Link

Portal: Use Current

Node: LOCAL_NODE

Menu: MAINTAIN_SECURITY

Component: CHANGE_PASSWORD

Market: GBL

Page: CHANGE_PASSWORD

Action: Update/Display

☐ Use data from current page in search

Actions

Action Type:

Related Control:

Process

Type:

Name:

Secondary Page

Page:

OK Cancel

Push Button/Hyperlink Properties dialog box: Type tab showing Internal Link properties

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Using Page Controls," Using Push Buttons and Links.

Component Definition Settings

Standard components and the components that you use as component-based pagelets have different purposes, which require that the developer configure the types of components differently. The following list describes the recommended property settings for components that you use as component-based pagelets on WorkCenter pages:

- WorkCenter pages should not generate a search page or start the search process when the user clicks the content reference to access the page. The user should immediately see the initial page.

Select a "dummy" record—one that has no keys or contains only one row of data—as the component search record on the Use tab.

The INSTALLATION record is an example of a "dummy" search record.

- Component-based pagelets generally require no save functionality.

Select the Disable Saving Page check box on the Use tab.

- Typically, the component should consist of only one page, but no tabs or page links should appear on the pagelet.

Deselect the Multi-Page Navigation check boxes on the Internet tab.

- The elements that make up the tool bar and page bar are not applicable and should not appear on the page.

Select the Disable Toolbar and Disable Pagebar check boxes on the Internet tab.

Notice the settings of the PTAL_DEMOPGLT component in this example:

- The search record is *INSTALLATION* and the Disable Saving Page check box is selected:

Component Properties

General Use Internet

Access

Search record:

Add search record:

☐ Force Search Processing

Detail page:

Context search record:

Actions

☐ Add

☒ Update/Display

☐ Update/Display All

☐ Correction

☒ Disable Saving Page

☒ Include in Navigation

☐ Mandatory Spell Check

☐ Read Only

3-Tier Execution Location

Component Build

☐ Client

☐ Application server

☒ Default (application server)

Component Save

☐ Client

☐ Application server

☐ Application server (with edits)

☒ Default (application server)

OK Cancel

Component Properties dialog box: Use tab settings for component used as a component-based template pagelet

- The Multi-Page Navigation check boxes are deselected and the Pagebar and the Toolbar are disabled:

Component Properties

General Use Internet

Search Page
Primary Action: ☐ New ☒ Search
Default Search Action: Update/Display
Default Search/Lookup Type: ☒ Basic ☐ Advanced
☒ Allow Action Mode Selection

Link To Access Add Page -- Message Set/Nbr: 124 62
Link To Access Search Page -- Message Set/Nbr: 124 63
Instructional Text -- Message Set/Nbr: 124 50

Multi-Page Navigation
☐ Display Folder Tabs (top)
☐ Display Hyperlinks (bottom)

Processing Mode
☒ Interactive ☒ Deferred
☐ Allow Expert Entry
☐ W/SRP Compliant

Pagebar
☒ Help Link
☒ Copy URL Link
☒ New Window Link
☒ Customize Page Link
☒ Disable Pagebar

Toolbar
Selected Toolbar Actions:
☐ Save ☐ Refresh
☒ Cancel ☒ Notify
☐ Spell Check ☒ View WorkList
☒ Return to List ☒ Next in WorkList
☒ Next in List ☒ Previous in WorkList
☒ Previous in List ☐ Add
☐ Next Page in Component ☒ Update/Display
☐ Previous Page in Component ☐ Update/Display All
☐ Correction
☒ Disable Toolbar

OK Cancel

Component Properties dialog box: Internet tab settings for a component used as a component-based template pagelet

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Creating Component Definitions," Setting Component Properties.

Registering Components as Pagelets for Use as WorkCenter Pagelets

After creating the component definition, you must register the component as a homepage pagelet or template pagelet in the PeopleTools portal. The PeopleTools portal stores content references based on their URL information. As a result, the combination of elements that make up the URL of every content reference in a portal must be unique. If a component is already registered as a content reference in the portal, then you must enter a "dummy" value in the Additional Parameters field when you register that same component as a template pagelet. Adding this field value ensures that the two content references are unique.

A sample value that works well is *name1=value1*, as shown in this example:

The screenshot displays a web interface for configuring content references. It features two main sections: 'URL Information' and 'Component Parameters'. In the 'URL Information' section, the '*Node Name' is set to 'LOCAL_NODE' and the 'URL Type' is 'PeopleSoft Component'. The 'Component Parameters' section includes fields for '*Menu Name' (PTAL_ADMIN), '*Market' (GBL), and '*Component' (PTAL_DEMOPLT). There is also a field for 'Additional Parameters' with the value 'name1=value1'. An example string 'name1=value1&name2=value2' is shown below the field.

Content Ref Administration page: URL Information and Component Parameters group boxes showing additional parameters

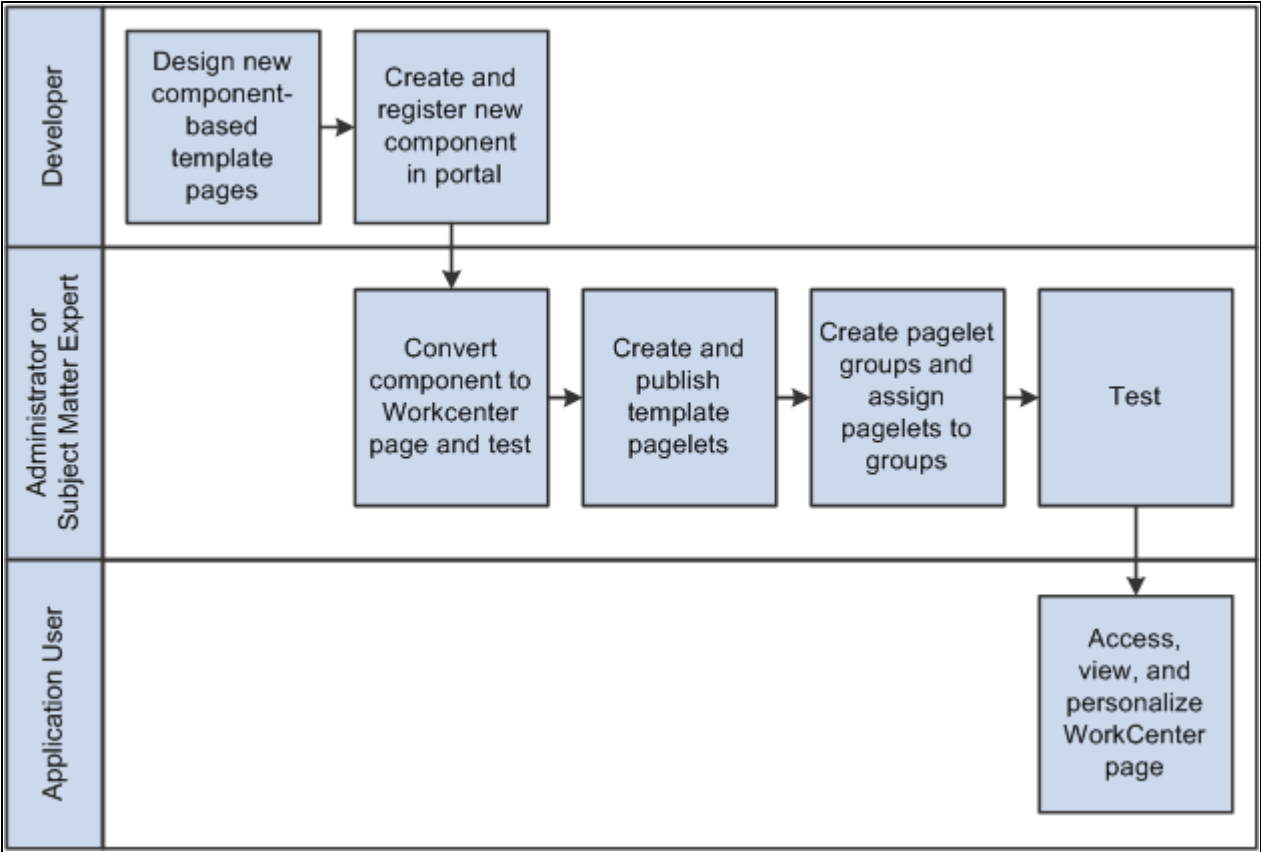
See [Chapter 5, "Administering Portals," Defining Content References, page 145](#) and [Chapter 5, "Administering Portals," URL Information, page 148](#).

See Also

PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide, "Getting Started with PeopleSoft Application Designer," Main Application Development Steps

WorkCenter Page Development and Implementation Process Overview

This diagram presents an overview of the process for developing, implementing, and using WorkCenter pages:



WorkCenter page development and implementation technology process flow

Note. You do not have to create new component-based pagelets. You can convert existing components or homepage pagelets to the WorkCenter template.

See [Chapter 17, "Creating and Configuring WorkCenter Pages,"](#) [Selecting the WorkCenter Starting Page,](#) page 635.

Prerequisites

This table lists the prerequisite steps that you must perform and references the corresponding documentation that you should review before you configure WorkCenter pages.

Prerequisite	Cross-Reference
Create navigation collections that you want to assign to the pagelet area.	See Chapter 4, "Working with Navigation Pages," Creating or Editing a Navigation Collection, page 65.
Create PeopleSoft Query definitions that you want to use in the pagelet area.	See <i>PeopleTools 8.52: PeopleSoft Query</i> , "Creating and Running Simple Queries," Creating New Queries.

Prerequisite	Cross-Reference
Create PeopleSoft page definitions, add and configure hyperlink page controls, and then define and register the components that you want to use as pagelets.	See <i>PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide</i> , "Creating Page Definitions." See <i>PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide</i> , "Using Page Controls," Using Push Buttons and Links. See <i>PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide</i> , "Creating Component Definitions," Defining Components.
Use Pagelet Wizard to create and publish template pagelets that use your navigation collections and query definitions as data sources.	See Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," page 323. See Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Selecting a PeopleSoft Query Data Source, page 366. See Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Selecting a Navigation Collection Data Source, page 364. See Chapter 11, "Using Pagelet Wizard to Create and Manage Pagelets," Template Pagelet, page 425.
Manually add component-based template pagelets to the template pagelets portal objects folder.	See Chapter 5, "Administering Portals," Defining Content References, page 145.

Defining Page Layout Templates

This section discusses how to define layout templates:

Page Used to Define Page Layout Templates

Page Name	Definition Name	Navigation	Usage
Define Page Layout Templates	PTAL_PAGE_TEMPLATE	PeopleTools, Portal, WorkCenter Pages, Define Page Layout Templates	Configure WorkCenter page appearance.

Configuring WorkCenter Page Layout

Access the Define Page Layout Templates page (Select PeopleTools, Portal, WorkCenter Pages, Define Page Layout Templates.)

Define Layout Templates

Define Page Layout Templates

Define the page layout templates used by WorkCenter Pages

Layout Template:

DEFAULT

*Description:

Default Layout

☒ Active☒ Default

Long Description:

This is the default layout template for all new Application Landing Pages.

Supporting Application Class

Package Name:

PTAL_PAGE

Path:

PageTemplate

Application Class ID:

LeftPageletAreaTemplate

Define Page Layout Templates page

- Layout Template

View the template ID, which is the name as entered on the Add New Value page. This field is display-only.
- Description

Enter text that briefly describes the purpose of the layout template. You can enter up to 30 characters.
- Active

Select to enable the template. If you select this check box, you must enter the supporting application class. This check box is disabled if the template is designated as the default template because the default template must be active.
- Default

Select to specify the template as the default template for all new WorkCenter pages. If you select this check box, you must designate the template as active.

The application allows only one default template. To replace the current default template, you must create a new template and designate it as the active, default template. When you save the page, the application makes the Default and Active check boxes display-only.

Supporting Application Class

After you define the application class that creates the WorkCenter template in PeopleSoft Application Designer, you enter the class-identifying information in these fields:

- Package Name

Enter the application class package name that contains the PeopleCode that defines the template.

Path

Enter the names—delimited by colons—of each subpackage in the hierarchy that drills down to the class that contains the method that defines the template. If the class is defined in the top-level package, the Path lookup search results display a colon. Select the colon.

Application Class ID

Enter the name of the application class that determines the layout of the page. The class must exist in the application package and subpackage (path) that you specify.

`LeftPageletAreaTemplate` and `RightPageletAreaTemplate` application classes come with all PeopleSoft applications.

Note. If the combination of the preceding fields creates an invalid application class path when you attempt to save the component, then the application displays an error message and prevents you from saving an invalid template. The error message specifies alternatives that you can take to eliminate the condition that causes the error.

Managing WorkCenter Page Configurations

This section discusses how to:

- Create and manage WorkCenter pages.
- Select the WorkCenter starting page.
- Assign pagelets to WorkCenter pages.

Pages Used to Manage WorkCenter Page Configurations

<i>Page Name</i>	<i>Definition Name</i>	<i>Navigation</i>	<i>Usage</i>
Manage WorkCenter Pages	PTAL_MANAGE_PAGES	PeopleTools, Portal, WorkCenter, Manage WorkCenter Pages	Administer WorkCenter pages.
Select the WorkCenter Starting Page	PTAL_SELECT_CREF	Click the Create New WorkCenter Page link.	Select the WorkCenter starting page.
Configure WorkCenter Page	PTAL_ASSIGN_PGLTS	<p>Select PeopleTools, Portal, WorkCenter, Manage WorkCenter Pages. Then click the Edit Configuration button.</p> <p>Alternatively, click the Pagelet Area Controls drop-down button in the pagelet area and select Configure WorkCenter page.</p>	<ul style="list-style-type: none"> • Specify WorkCenter page layout options. • Create pagelet groups. • Assign pagelets to pagelet groups.

Administering WorkCenter Pages

Access the Manage WorkCenter Pages page. (Select PeopleTools, Portal, WorkCenter, Manage WorkCenter Pages.)

If you have not configured any WorkCenter pages, this page appears:

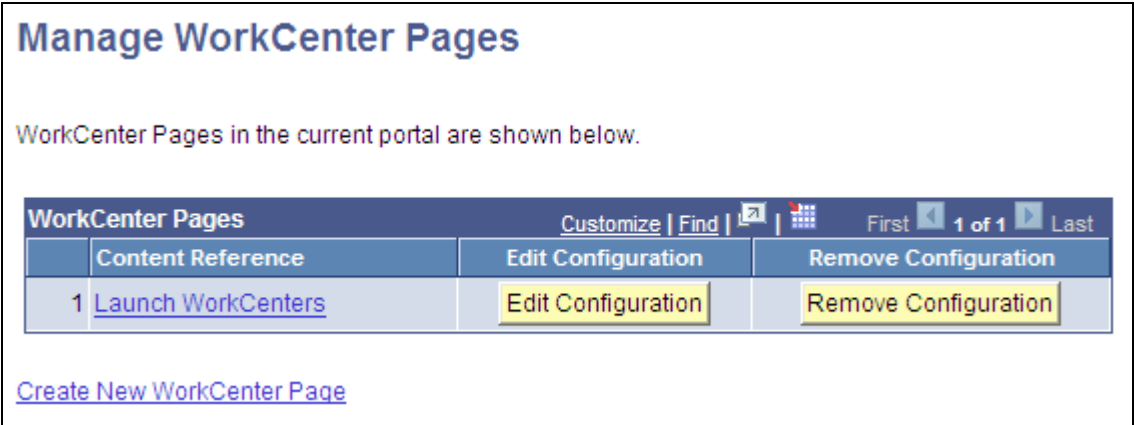


Manage WorkCenter Pages page before you add WorkCenter pages

Create New WorkCenter Page

Click to access the Select the WorkCenter Starting Page page and begin the process of creating a WorkCenter page.

If you have configured at least one WorkCenter page, your page will look similar to this one:



Manage WorkCenter Pages page after you add WorkCenter pages

Content Reference	Displays the content reference name as an active link. Click to access the content reference and view the WorkCenter page.
Edit Configuration	Click to access the Configure WorkCenter Page page.
Remove Configuration	Click to delete the WorkCenter page, all pagelet groups, and all pagelet configurations associated with the content reference.

Note. Removing WorkCenter page configurations does not delete the content reference upon which it is based nor does it delete the pagelets.

Selecting the WorkCenter Starting Page

Access the Select the WorkCenter Starting Page page. (Select PeopleTools, Portal, WorkCenter, Manage WorkCenter Pages. Then, click the Create new WorkCenter Page link.)

Select the WorkCenter Starting Page

Click the content reference name to pick a menu item to be used as the starting page of the WorkCenter page. Click the "Cancel" button to go back to the previous page.

☐ Include hidden items

Left | Right

Root

Administer Procurement
 Control Inventory
 Define Business Rules
 Maintain Items
 Manage Production
 Manage Sales Activities
 Manage Treasury
 PeopleTools Quality
 Plan Production
 Process Financial Information
 Schedule Resources
 Self Service
 Structure Manufacturing
 PeopleTools SDK
 Mobile Demo
 Manage Assets
 Worklist
 Application Diagnostics
 Tree Manager
 Reporting Tools
 PeopleTools
 [\[Usage Monitoring\]](#)[\[Change My Password\]](#)[\[My Personalizations\]](#)

Select the WorkCenter Starting Page page

The Select the WorkCenter Starting Page page appears when you are creating new WorkCenter pages only. If you are editing an existing WorkCenter page, the Configure WorkCenter Page page appears.

All content references to which you have been granted access in the current portal appear in the tree, including both local and imported content. Select the menu item that you want to convert to a WorkCenter page.

To select the WorkCenter starting page:

1. Click an upper-level folder to begin navigating through the menu structure.
2. Continue clicking folders and traversing the menu structure until you find the WorkCenter starting page.
3. Click the content reference.

The Configure WorkCenter Page page appears. If the selected content reference is already a WorkCenter page, the Configure WorkCenter Page page displays the page layout options, pagelet groups, and pagelets that are associated with the content reference.

Configuring WorkCenter Pages

Access the Configure WorkCenter Page page by using one of these navigation paths:

- Select PeopleTools, Portal, WorkCenter, Manage WorkCenter Pages. Click the Create a New WorkCenter Page, link and then select a content reference.
- Select PeopleTools, Portal, WorkCenter, Manage WorkCenter Pages. Click the Edit Configuration button.
- In the WorkCenter pagelet area, click the pagelet area controls drop-down button.

Configure WorkCenter Page

Configure the page layout, and assign pagelets to be displayed with this WorkCenter page.

*WorkCenter Label

Starting Page

This is the starting page of the WorkCenter page. To edit its properties, please go to the Structure and Content page.

Portal Name EMPLOYEE

Menu Item PT_PTAL_SAMPLE_PAGE_GBL [Sample WorkCenter Page](#)

Page Layout Options

*Layout Template

Layout HTML Object

*Initial Size Max no of Minimized Pagelets

Pagelet Groups

Find | View All First 1 of 1 Last

*Group ID

*Group Label

Display Order ☐ Default

Pagelets								Find	First	1 of 1	Last
*Pagelet Type	*Pagelet Name	Pagelet Label	Display Order	Set Focus	Required	Initially Minimized					
Homepage Pagelet	<input type="text"/>			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>					

Configure WorkCenter Page page

Configure WorkCenter Page

WorkCenter Label

Enter the name that appears in the header of the pagelet area. This field can contain up to 30 characters.

The default value is the name of the content reference that you select as the starting page. This value also appears in tool tip text, in the links that appear in the pagelet area controls drop-down menu, and on the User Personalization page if one exists.

Return to Manage WorkCenter Pages

Click to access the Manage WorkCenter Pages page. This link is visible only if you access the Assign Templates Pagelets page from the Manage WorkCenter Pages page.

Starting Page

Portal Name

Displays the current portal definition name.

Menu Item Displays the menu item name that is associated with the content reference link that you selected from the tree on the Select a WorkCenter Starting page. You can also click the link that appears beside the menu item to view the WorkCenter page.

Page Layout Options

Use these options to change the appearance and character of the WorkCenter page.

Changes that you make to the page layout options are visible after you save the page and click the Refresh button on your *browser*.

Layout Template	Select a layout template from the list of all active page layout templates. The name of the default page layout template appears here when you create a new WorkCenter page.
Layout HTML Object	Select a custom layout, which you define in an HTML definition in PeopleSoft Application Designer. Only HTML definitions that begin with <i>PTAL_LAYOUT_</i> appear in this prompt field.
Initial Size	Enter an integer between 200 and 9999 that determines the width of the pagelet area when the user initially accesses the WorkCenter page. The default value is 256.
Max no of Minimized Pagelets	Specify the number of pagelets that, when minimized, remain visible as collapsed headers in the pagelet area. If the number of minimized pagelets exceeds this number, any other minimized pagelets appear as links in the More drop-down menu at the bottom of the pagelet area.

Pagelet Groups

Use the fields in this scroll area to organize the pagelets in the pagelet area into tabbed groups.

Group ID	Enter a value that identifies the group. Each value must be unique to the WorkCenter page and can contain no more the 20 characters. After you initially add this value, the field becomes display-only. This field is required.
Group Label	Enter a meaningful name of up to 20 mixed-case characters for the pagelet group. The label appears on a tab in the pagelet area and is visible to the user. This field is required.
Display Order	Enter a number between 0 and 999 to manually sequence the order in which the pagelet groups appear. This field is empty by default and pagelet groups appear in the same order as the row sort order in the Pagelet Groups scroll area. If you omit the Display Order value then the system considers it to be 0 and appear before groups for which the field contains a value.

Default Select to display the group by default when the WorkCenter page is initially accessed. Only one group can be the default.

Note these points about pagelet groups:

- If you create only one pagelet group, no tab appears to the end user.
- If you delete a pagelet group, the application deletes all subordinate pagelet data and personalizations.

Pagelets

Use the fields in this grid to assign the pagelets to the group.

Pagelet Type	Select <i>Template Pagelet</i> or <i>Homepage Pagelet</i> .
Pagelet Name	Select the pagelet from a list of registered pagelets. Only registered pagelets appear in the list. A pagelet can be assigned to <i>only one</i> group on a WorkCenter page.
Pagelet Label	Displays the label that appears in the pagelet header.
Display Order	Enter a number between 0 and 999 to specifically determine the position of the pagelets in the group. This field is empty by default and pagelet groups appear in the same order as the row sort order in the Pagelets grid. Pagelets for which you leave the Display Order field blank are considered to have a value of <i>0</i> and appear before groups for which the field contains a value.
Set Focus	Select to move the cursor to the pagelet when the user initially accesses the WorkCenter page. You can select this check box for only <i>one</i> pagelet in the same group.
Required	Select to prevent the application user from hiding the pagelet in the group. The user can still minimize the pagelet.
Initially Minimized	Select to display the pagelet in the minimized state when the user initially accesses the WorkCenter page.

Part 5

Working with Portlet Technologies

Chapter 18

Understanding Web Services for Remote Portlets (WSRP)

Chapter 19

Configuring WS-Security for WSRP Consumption and Production

Chapter 20

Using WSRP to Consume and Produce Remote Portlets

Chapter 21

Working with JSR 168 Portlets

Chapter 18

Understanding Web Services for Remote Portlets (WSRP)

This chapter provides:

- An overview of Web Services for Remote Portlets (WSRP).
- A terminology list.
- A scenario of how to implement portlet producers and consumers.
- Considerations for WSRP and server cluster configuration.

Understanding WSRP

WSRP is a web services protocol that is used to bring together content and interactive web applications from remote sources. WSRP incorporates standards such as XML, Simple Object Access Protocol (SOAP), and Web Service Description Language (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, he or she requests a WSDL document to determine the location of the service, the associated function calls, and how to access them. The administrator then uses this information in the WSDL to form a SOAP request and send it via HTTP to the external system endpoint.

Oracle 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 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 applies 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, they 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 *PeopleTools 8.52: 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 that 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. These portlets then invoke the `performBlockingInteraction` operation to process user interactions to the producer.

WSRP also specifies the following optional interfaces:

Registration Interface: Provides an in-band mechanism for a consumer to register with a producer. It enables the producer to customize its behavior for each consumer based on registration information. WSRP also allows out-of-band registrations and no registration.

Portlet Management Interface: Enables 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, Oracle enables you 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.

With 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 the producer can verify the identity to run requested web services on behalf of the user without requiring a user to sign in. Single signon between the web services consumer and producer is currently supported in PeopleSoft WSRP Portal, PeopleSoft Integration Broker, and the BPEL product.

- SOAP message integrity. This ensures that no one has tampered with the messages.
- SOAP message confidentiality. This guarantees 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 and password or user authentication using user name and digital signature through the use of Web Services Security: Username Token Profile and SAML Token.

Note. Oracle provides multiple levels of security for WSRP. These levels, or options, are discussed in the following chapter. Oracle recommends that you determine the level that is appropriate for your needs before implementing WS-Security. Using SSL (Secure Sockets Layer) connections to secure transmissions may be sufficient.

See [Chapter 9, "Configuring the Portal Environment," Improving Same-Server Performance Under SSL, page 288](#) and [Chapter 19, "Configuring WS-Security for WSRP Consumption and Production," page 649](#).

Terminology

ASF	Apache Software Foundation.
Java Portlet	A Java software module that conforms to the Portlet API.
JCP	<i>Java Community Process</i> : Established for the development of Java technology.
JSR	<i>Java Specification Request</i> : Each submission to the JCP gets assigned a unique JSR number.
JSR 168	The JCP specification that describes the Portlet API.

SAML	<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 http://www.oasis-open.org/glossary/index.php</p>
SOAP	<p><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.</p>
WSDL	<p><i>Web Service Description Language:</i> An 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.</p>
WSRP	<p><i>Web Services for Remote Portlets:</i> A web services protocol for bringing together content and interactive web applications from remote sources.</p>
WSS	<p><i>Web Services Security Language:</i> Supports security mechanisms, each using implementation and language-neutral XML formats, which include the use of:</p> <ul style="list-style-type: none"> • XML signature to provide SOAP message integrity. • Use of XML encryption to provide SOAP message confidentiality. • Attaching and referencing security tokens and associating signatures with security tokens.
WSRP4J	<p>The ASF WSRP reference implementation project.</p>
XML	<p><i>Extensible Markup Language:</i> Describes data and focuses on what data is. XML is designed to structure, store, and send information.</p>

Implementing WSRP Protocol Scenario

This scenario depicts interactions between two companies: Oracle America (a WSRP producer), and Kane Consulting (a WSRP consumer).

In this scenario, Kane Consulting is an online company providing personalized financial services to clients by subscription. Oracle America wants to host a number of financial applications, including a web-based inventory planning application. Kane Consulting wants to offer this application to its clients via its web pages.

Without WSRP, to offer the inventory planning application to clients, Oracle America and Kane Consulting must agree on the following procedure:

1. Oracle America 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 a link to the inventory planning application.
3. Kane Consulting then transmits a request to Oracle America to obtain the initial view of the application. Oracle America 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 so that when the links 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 enters 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 Oracle America to process the client transaction.
8. Oracle America 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. Oracle America generates the markup and returns it to Kane Consulting.
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 preceding procedure, Oracle America and Kane Consulting can use WSRP as the protocol. Oracle America 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, Oracle America and Kane Consulting use WSRP to define various interactions, with Oracle America 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, Oracle America 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 Oracle America (Step 7).

By implementing the preceding interfaces and agreeing to conform to WSRP, both Oracle America and Kane Consulting can use a standard process to offer and consume portlets. In addition, Oracle America can offer the same portlets to company X as long as company X adheres to WSRP, and Kane Consulting can consume additional portlets offered by company Y provided that company Y also implements WSRP interfaces.

Considerations for WSRP and Server Cluster Configuration

Support for web server clustering with session replication is supported. However, the open source products that the application 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 19

Configuring WS-Security for WSRP Consumption and Production

This chapter provides overviews of determining security requirements and WS-Security for WSRP and discusses how to:

- Set up WSRP security options.
- Set up Security Assertion Markup Language (SAML).
- Configure WS-Security for PeopleSoft as a WSRP producer.
- 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, you first need to determine the level of security that you require. If you are consuming remote portlets, the level that 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, six types of token security options are available for WS-Security (set in the `server-config.wsdd` file):

Important! The security option used by the producer *must* be an exact match to the security option used by the consumer.

- *None*: No security token is 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. 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.

In addition, PeopleSoft features a WSRP WSS Enabled Response option. When specified in an outbound consumer request, the producer response must include a WSS header.

Understanding the PeopleSoft WS-Security for WSRP

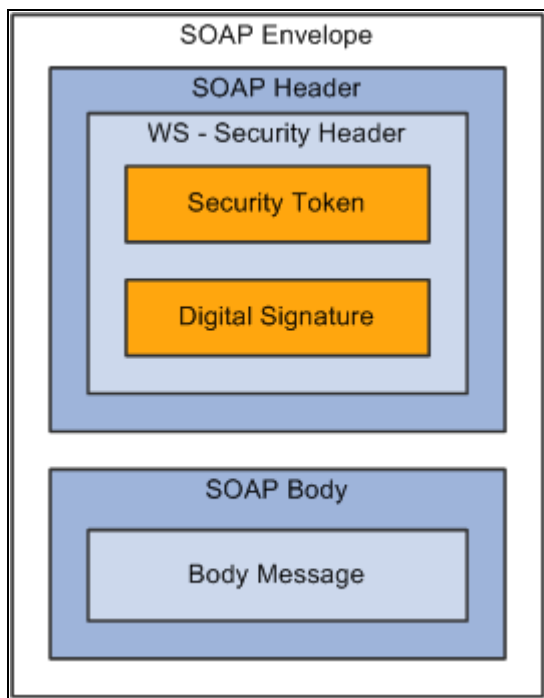
WSRP involves passing SOAP messages between the WSRP consumer and producer. To safely use WSRP, Oracle provides message-level security between the consumer and the producer by incorporating WS-Security.

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 sign in.
- *Message integrity:* Ensures that messages have not been tampered with.
- *Message confidentiality:* Guarantees that messages are protected.

This diagram shows the SOAP envelope, SOAP header, and SOAP body and how WS-Security embeds the security token in the SOAP messages:



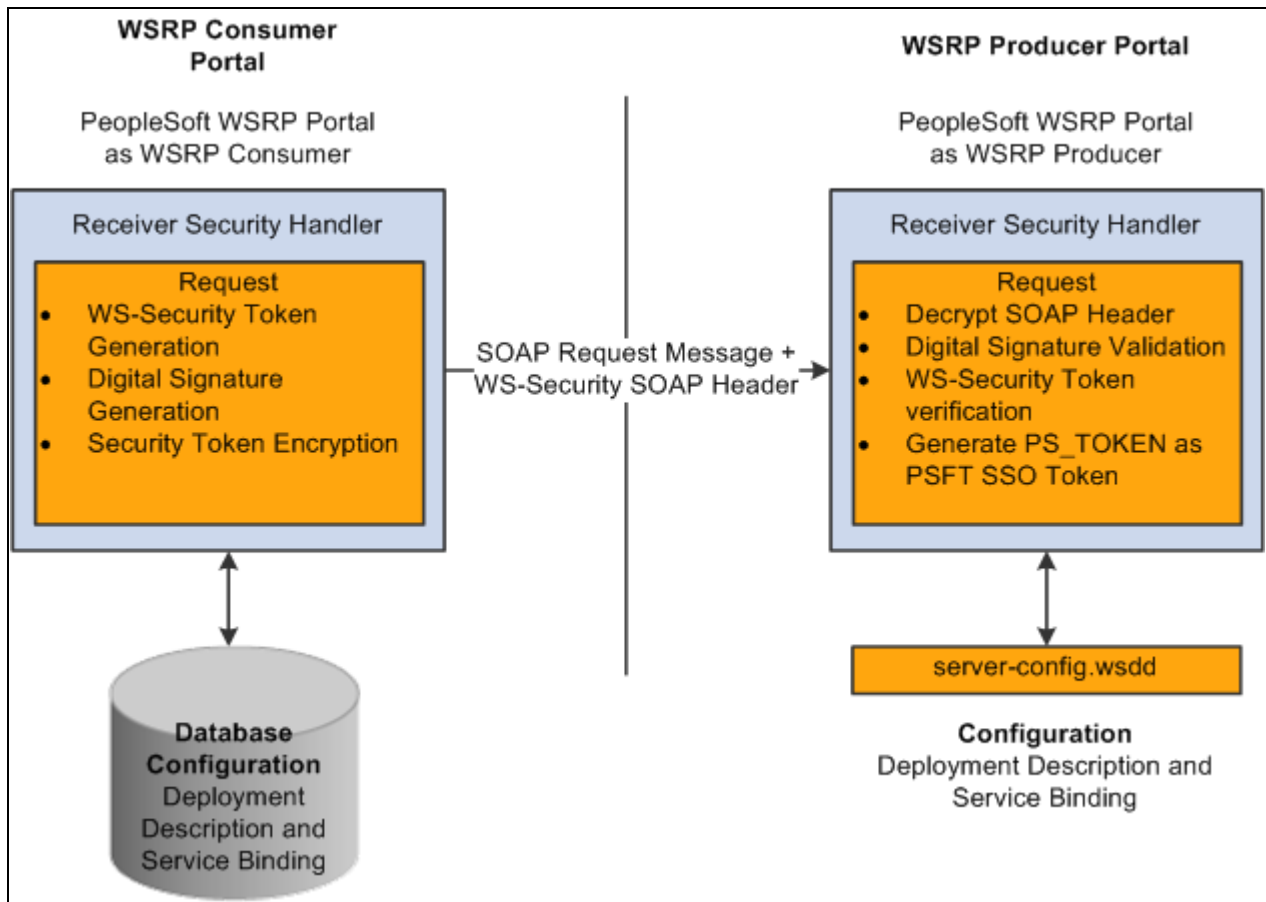
WS-Security SOAP message structure

When the PeopleSoft portal is a WSRP consumer, the user ID and password of the user signed in to 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 the PeopleSoft system 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 the PeopleSoft system. 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 the PeopleSoft Portal 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 clear text.

```
<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+cPp3NahsF1mVxcXzCjrui2Vc3vq
        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=
            PeopleToolsDevelopment,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:UsernameToken></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/LogTestService9"/></soapenv:Body>
  </soapenv:Envelope>
```

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 NotOnOrAfter conditions
- Subject
- Name
- Subject confirmation
- Conditions under which the assertion is valid

This is an XML example of an 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" 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 an 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: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</ConfirmationMethod>
</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">MIIElzCCBACgAwIBAgIKQrSKy→
QAAAAAL+DANBgkqhkiG9w0BAQUFADCBvJELMAKGA1UEBhMCVVMxCzAJBgNVBAGTAKNBMRMwEQYDVQQHEWp→
QbGVhc2FudG9wMRcwFQYDVQQKEw5QZW9wbGVUb2Z0IEluYzEgMB4GA1UECXMUGVvcGx1VG9vbHMg→
RGV2ZWxvcGllbnQxEzARBgoJkiaJk/IsZAEZFgNjb20xGjAYBggoJkiaJk/IsZAEZFgppwZw9wb→
GVzb2Z0MSEwHwYDVQQDEzhQZW9wbGVUb29scyBURVNUIHJvb3QgQ0EwHhcNMDE2MTc1MDIzWhc→
NMDgwODE2MTgwMDIzWjBxMQswCQYDVQQGEwJVUzETMBEGA1UECBMKQ2FsaWZvcml5pYETMBEGA1UEBx→
MKUGxlYXNhbnRvbWJEPMA0GA1UEChMGMTJhY2x1MRQwEgYDVQQLEwtQZW9wbGVUb29sczERMA8GA1UEAww→
IcWVfaWJ0Z3QwZz8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMrvvgIBLkotkSm+tnbxjyNq1→
Ti3h3p8h44WhIj1Ikdvhtgwk890dodxFlHkDaX0Zz7/9IcW4wc5l3z/C/r0kchcq95ToRxKZ5UC50Sx2wb→
T9/GL6GVUP/EBtGqcaZSeE4G90HiVmeXxe70pb3InyyAxU/dbuLj/bMpOvABgbmdhAgMBAAGjggHm→
MIIB4jAdBgNVHQ4EFgQUZ1KpPrdYQY+mEdNk/YWKe4Iqn+wwHwYDVR0jBBGwFoAU/jeXgdgdjCBw→
II0g3c+t5n06BswwgawGA1UdHwSBpDCBoTCBnqCBm6CBmIZMaHR0cDovL3B0bnRhcEYLnBlb3BsZXNvZn→
QuY29tL0NlcnRFbnJvbGwvUGVvcGx1VG9vbHM1MjBURVNUIHJvb3QgQ0EwHhcNMDE2MTc1MDIzWhc→
L1xcCHRudGFzMTIucGVvcGxlc29mdC5jb21cQ2VydEVucm9sbFxxQZW9wbGVUb29scyBURVNUIHJvb3Qg→
Q0EuY3JSMIHwBggrBgEFBQcBAQSB4zCB4DBwBggrBgEFBQcwAoZkaHR0cDovL3B0bnRhcEYLnBlb3Bs→
ZXNvZnQuY29tL0NlcnRFbnJvbGwvUGVvcGx1VG9vbHM1MjBURVNUIHJvb3QgQ0EwHhcNMDE2MTc1MDIzWhc→
BURVNUIHJvb3QgQ0EwHhcNMDE2MTc1MDIzWhcGxlc29mdC5jb21cQ2VydEVucm9sbFxxQZW9wbGVUb29scy→
BURVNUIHJvb3QgQ0EuY3J0MA0GCSqGSIb3DQEBAQUAA4GBABaZwo6xhKZFRbESi3ICewrBTKFjtDbmJvVq→
BaB0pConBMRGDJ0bQf9Rwo6/Ucm/BoUEEP/dzBkLYM0NEuEgoQvF4ZHRD73qNV9CCUHU3nlwfn→
L5K54qrda4V2CoBvgppHEU7EVdt47YV2E8HUAUfyDaXZ0prRMB6I2KtKaaYBNI</wsse:BinarySecurityToken>
- <ds:Signature Id="Signature-22949069" 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="#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-c14n#" ⇒
/>
  </wsse:TransformationParameters>
</ds:Transform>
</ds:Transforms>
  <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
  <ds:DigestValue>TK2a7xf+ldF9MkIlXYut8g5RG+A=</ds:DigestValue>
</ds:Reference>
- <ds:Reference URI="#id-78219">
- <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>wTVh0pHi6NrTKDWnyXbX/WNCw68=</ds:DigestValue>
</ds:Reference>
</ds:SignedInfo>
  <ds:SignatureValue>LGWmXxak++duS8IfY+/4BphfPJW+Ka6a8DxUfYmUGUZ57d1HOMTs0fLR0Roqd⇒
IOKJtSD33qRyU6p 7lufZXsRoiJD5iEUJr+El7KgBEmFPFV5hDx2a+dnHN8Zd9A1DRh7qzrlewKcRpd⇒
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">
  <ns1:testMethod xmlns:ns1="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, Oracle recommends that you use HTTPS to protect the username and password. For you 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. In the preceding example, <Producer Hostname> is the hostname of the producer web server and <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>\webserv\peoplesoft\applications\peoplesoft\pspc\WEB-INF\config\services\PIAService.properties.
```

Note. This is similar to using Web Profile Default Addressing fields.

See [Chapter 9, "Configuring the Portal Environment," Configuring Web Profiles, page 238.](#)

In this properties file, specify the default protocol, host, and port to override the incoming request protocol, host, and port. The settings that WSRP checks in the PIAService.properties file are:

usePIAConfig = true	Checks whether settings in the PIAService.properties file should be used. If you set usePIAConfig to 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.
portalHost=<your webserver host>	The default host to use.
portalPort=<your port>	The default port to use.
SSLToPIA=true	If you set SSLToPIA to true, the system uses HTTPS; otherwise, it uses the HTTP scheme.

See Also

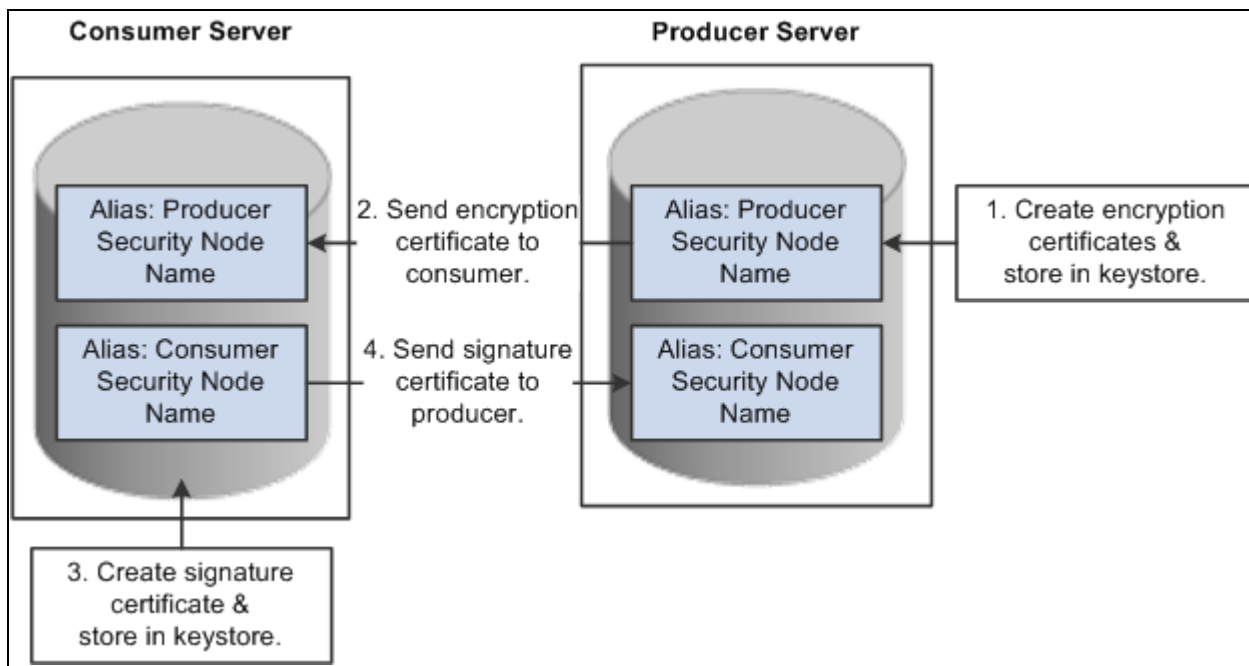
PeopleTools 8.52: System and Server Administration, "Working with Oracle WebLogic," Implementing WebLogic SSL Keys and Certificates

PeopleTools 8.52: 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 *PeopleTools 8.52: System and Server Administration*, "Working with Oracle WebLogic," Implementing WebLogic SSL Keys and Certificates and *PeopleTools 8.52: 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 that 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. You can do this by 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 that 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. 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. You can do this by 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. 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

Here 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 the `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. Update the encrypted password in the `wss.properties` file for `org.apache.ws.security.crypto.merlin.keystore.password` with the output from PSCipher.

See *PeopleTools 8.52: 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

PeopleTools 8.52: 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 *PeopleTools 8.52: Security Administration*, "Working with SSL/TLS and Digital Certificates," Configuring Digital Certificates.

Configuring the SAML Inbound Setup

The SAML Inbound Setup page creates an InBound web service 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 web service for each external user who accesses the PeopleSoft system and 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 that are 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

qe_local

*Active Flag

A

*Issuer

.PEOPLESOFT.COM

*SubjectName

QEDMO

*QualifierName

.PEOPLESOFT.COM

*Mapping Peoplesoft
UserID

QESAML

Long Description

Security Assertion Markup Language [SAML] Inbound Setup page

Certificate Alias	Enter the sender's public key, which you imported in the previous step (Importing Digital Certificates).
	Note. This key must be base-64 encoded.
Issuer	Enter the domain name of the issuing entity.
SubjectName	Enter a user ID or email address.
QualifierName	Enter the domain name of the issuing entity.

Mapping PeopleSoft UserID

Enter the user ID to map to the SubjectName. This field sets the PeopleSoft internal permissions for the external user and prevents cross-site vulnerability.

Note. This user ID does not have to be the user ID of the sender, but must be a valid PeopleSoft user in the PSOPRDEFN table.

Note. 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:

1. Navigate to the producer web server folder, for example *PSHOME/Webserver/bin*.
2. Double-click the file to launch the program.
3. Select *Option 5: Redeploy WSRPBaseService with the 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:

Security Option	Filename
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. UsernameToken, No Password, Digitally Signed	server-config.wsdd.unsigned
5. SAMLToken Full Security	server-config.wsdd.saml-token
6. SAMLToken Full Security Option (timestamp)**	server-config.wsdd.saml-token-timestamp
7. UsernameToken Full Security Option With WSS Response*	server-config.wsdd.fullsecurityR
8. UsernameToken, No Password Full Security Option With WSS Response*	server-config.wsdd.unsignedR
9. SAMLToken Full Security Option With WSS Response*	server-config.wsdd.saml-tokenR

Security Option	Filename
10. SAMLToken Full Security Option (timestamp) With WSS Response* **	server-config.wsdd.samltokentimestampR

* The response message must be signed and encrypted.

** A PeoplesSoft SAML token request does not contain a timestamp. However PeopleSoft can accept third-party SAML requests that contain timestamps. No configuration on the PeopleSoft system is required.

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\webserv\%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 the 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 the 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 configure WS-Security for WSRP consumers.

Pages Used to Define Nodes With WS-Security

Page Name	Definition Name	Navigation	Usage
Portal	IB_NODEURI	PeopleTools, Portal, Node Definitions, Portal	Enter required information for the WS-Security node.
WS Security	IB_NODESECURITY	Click the WS Security tab on the Portal page.	<p>Enable secure access based on WS-Security for remote producers.</p> <p>Note. Secure access is optional. You must be an administrator to access this page.</p>

Configuring WS-Security for WSRP Consumers

Access the WS Security page. (Select PeopleTools, Portal, Node Definitions and click the WS Security tab.)

Node Definition – WS Security page

Use the Node Definition – WS Security page to select the authentication token type, as well as encryption, digital signature, and WSS response options.

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 deselect all check boxes.

Security Option	Authentication Token Options
Authentication Token as Username Token	Select <i>Username Token</i> as the authentication token type, and deselect 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. Note. With this setting, the Username token is encrypted and messages are digitally signed.
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. Note. By default, SAML tokens are digitally signed.
Authentication Token as SAML Token with full security	Select <i>SAML Token</i> and select the Encrypted check box. Note. With this setting, the SAML token is encrypted and by default messages are digitally signed.
Authentication Token as Username Token with full security and WSS Response*	Select <i>Username Token</i> as the authentication token type, and select the Encrypted, Digitally Signed, and <i>WSRP WSS Enabled Response</i> check boxes. Note. With this setting, the Username token is encrypted and messages are digitally signed.
Authentication Token as Username Token , no password, with full security and WSS response*	Select <i>Username Token, no password</i> as the authentication token type, and select the Digitally Signed, and <i>WSRP WSS Enabled Response</i> check boxes.
Authentication Token as SAML Token with full security and WSS response*	Select <i>SAML Token</i> as the authentication token type, and select the Encrypted and <i>WSRP WSS Enabled Response</i> check boxes. Note. With this setting, the SAML token is encrypted and by default messages are digitally signed.

* The response message must be signed and encrypted.

See Also

Chapter 20, "Using WSRP to Consume and Produce Remote Portlets," Importing Producer Information, page 672

Chapter 19, "Configuring WS-Security for WSRP Consumption and Production," Determining Security Requirements, page 649

Chapter 9, "Configuring the Portal Environment," Defining Portal Nodes, page 279

Chapter 20

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 10, "Administering Homepages, Dashboard Pages and Pagelets," page 301

<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. (Select PeopleTools, Integration Broker, Configuration, Gateways.)

Gateways

Gateway ID: LOCAL

☒ Local Gateway ☐ Load Balancer

URL:

[Gateway Setup Properties](#)

Connectors				Customize	Find	First	1-9 of 9	Last
*Connector ID	Description	*Connector Class Name						
1	SMTPTARGET	SMTPTargetConnector	Properties	+	-			
2	FILEOUTPUT	SimpleFileTargetConnector	Properties	+	-			
3	PSFTTARGET	PeopleSoftTargetConnector	Properties	+	-			
4	JMSTARGET	JMSTargetConnector	Properties	+	-			
5	HTTPTARGET	HttpTargetConnector	Properties	+	-			
6	GETMAILTARGET	GetMailTargetConnector	Properties	+	-			
7	FTPTARGET	FTPTargetConnector	Properties	+	-			
8	AS2TARGET	AS2TargetConnector	Properties	+	-			
9	PSFT81TARGET	ApplicationMessagingTargetConnector	Properties	+	-			

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 appear in the Connectors grid.

4. Click Save.

Using the WSRP Consumption Wizard (Consumer)

This section provides an overview of the WSRP Consumption Wizard data, lists common elements used in this section, 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 available only to users of the current portal, an administrator can sign in to a different portal and manually create CRefs for any of the producer-offered portlets.

See Also

Chapter 5, "Administering Portals," Administering Content References, page 143

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 colors signify the status of the page:</p> <ul style="list-style-type: none">• <i>Pale Blue</i>: The page has been updated.• <i>Orange</i>: Current page.• <i>Dark Blue</i>: The page has not been updated.
Previous and Next	Click to navigate page by page.
Producer Title and Description	<p>Enter meaningful titles and descriptions.</p> <hr/> <p>Note. The producer title is also used as the default folder label for the content references created based on the producer's offered portlets.</p> <hr/>
Creation Date/Time and Last Update Date/Time	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 producer's information, or delete a producer.
Import Producer Information	PT_WSRP_IMPORT	Click Edit or Add a Import Producer Information.	Enter the producer's WSDL URL to import required information.
Registration	PT_WSRP_REGISTER	Click Save on the Import Producer Information page.	Enter 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 producer's 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 button.)

WSRP Consumption
Step 1 of 3

1

2

3

Next >

Import Producer Information

Enter new or modify existing producer information.

Producer Name: XYZ_CORP

Creation Date/Time: 11/01/2004 11:45:39AM

Last Update Date/Time:

***Producer Title:**

Description:

Security Node:

Web Service Endpoint URL

***URL:**

Status: Import Complete

Import

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 that are created based on this producer's offered portlets.

Security Node (Optional) Select the node definition that was created for WS-Security.
See [Chapter 19, "Configuring WS-Security for WSRP Consumption and Production," page 649](#).

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 be updated automatically 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 about how to make corrections.

Save

Click to save the producer information and go to the Registration page.

Registering Producers

Access the Registration page. (Click the Next button.)

WSRP Consumption**Step 2 of 3**

123

< PreviousNext >

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: 4WMUDGQAAAAANQASGFQQAAA

☑ 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:	<input type="text" value="erica_kane@peoplesoft.com"/>	?
*Company Name:	<input type="text" value="PeopleSoft, Inc."/>	?
*Location:	<input type="text" value="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 appears 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 producer's 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"/>	View	Details	Delete				
GooglePortlet	Sun: Google Search	<input checked="" type="checkbox"/>	View	Details	Delete				
HelloWorldPortlet	Sun: HelloWorldPortlet	<input checked="" type="checkbox"/>	View	Details	Delete				
InputStreamPortlet	Sun: Test Input Stream	<input checked="" type="checkbox"/>	View	Details	Delete				
JSPPortlet	Sun: ShowTime Portlet	<input checked="" type="checkbox"/>	View	Details	Delete				
NamedDispatcherGetAttributePortlet	Sun: Test NamedDispatcher	<input checked="" type="checkbox"/>	View	Details	Delete				
NavStatePortlet	Sun: Nav State	<input checked="" type="checkbox"/>	View	Details	Delete				
NotepadPortlet	Sun: Notepad Portlet	<input checked="" type="checkbox"/>	View	Details	Delete				
PrefPortlet	Sun: PrefPortlet	<input checked="" type="checkbox"/>	View	Details	Delete				
RequestDispatcherGetAttributePortlet	Sun: Test RequestDispatcher	<input checked="" type="checkbox"/>	View	Details	Delete				
RssPortlet	Sun: RSS	<input checked="" type="checkbox"/>	View	Details	Delete				
TestPortlet	Sun: TestPortlet	<input checked="" type="checkbox"/>	View	Details	Delete				
WeatherPortlet	Sun: WeatherPortlet	<input checked="" type="checkbox"/>	View	Details	Delete				

Producer Offered Portlets page

Update Portlet List

Click to refresh the list of offered portlets.

Important! When previously listed portlets are no longer offered by a producer, the Delete button associated with those portlets becomes active.

Producer Details

Click to go to the Producer Details page and view the producer's actual ServiceDescriptionResponse.

Create Pagelets for All Selected Portlets

Click to create content references for the selected portlets.

These content references can be located in a folder created automatically by means of the values entered for this producer name and title for its name.

This will make the portlets available when end users are personalizing their homepages.

Note. You can also create the content references manually in any portal using the Structure and Content page.

See [Chapter 5, "Administering Portals," Administering Content References, page 143.](#)

Producer Offered Portlets**Selected**

Select the portlets that you want to consume, then click the Create Pagelets for All Selected Portlets button to create content references.

You can also use the Select All and Deselect All buttons.

View

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.

Note. This button is disabled until a content reference is created.

Details

Click to go to the Portlet Details page to view the XML that was returned from the producer for this portlet.

Delete

Click 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.

Click the Delete button to remove the portlet and associated content reference.

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"> • PeopleTools, Portal, Structure and Content, Portal Objects, Pagelets Click the desired folder name, then click Edit for the desired portlet. • 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. 	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.	Select the remote portlets that you want to appear on your homepage.

Reviewing Content References

Access the Content Ref Administration page for a remote portlet. (Select 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 >

Content Ref Administration

*Name:

*Label:

Long Description: (254 Characters)

Product:

Sequence number:

Object Owner ID

Usage Type:

Storage Type:

Template Name:

Author: VP1

Parent Folder: Pagelets

*Valid from date:

Valid to date:

Creation Date: 09/11/2009

☐ No Template

URL Information

URL Type:

*Producer:

*Portlet:

[Producer Details](#)

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 | The default is <code>WSRP_<portlet handle></code> .
This field is read-only and cannot be changed. |
| Label | Enter a meaningful label description for this portlet.
The default is the remote portlet short title. |
| Parent Folder | The default is the producer's title.
Click the Select New Parent button to move the content reference to a different folder. |
| Usage Type | The default is <i>Pagelet</i> . |
| Storage Type | The default is <i>Remote by URL</i> .
This field is read-only and cannot be changed. |

URL Information

- | | |
|-----------------|--|
| URL Type | The default is <i>Remote Portlet</i> . |
|-----------------|--|

Producer	The default is the producer's name.
Portlet	The default is the portlet handle.
Producer Details	Click to go to the consumption wizard's Import Producer Information page for this producer.

Note. 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 145.](#)

Displaying Remote Portlets on Homepages

Access the Tab Content page. (Select PeopleTools, Portal, Structure and Content, Portal Objects, Homepage, Tabs. Click the Edit button associated with the My Page content reference. Select 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.

QE
☒ Include all?

<input checked="" type="checkbox"/> Display Cookies	Optional
<input checked="" type="checkbox"/> QE Portal Links	Optional
<input checked="" type="checkbox"/> QE Redirect Test	Optional
<input checked="" type="checkbox"/> QE_COMMAINCOOKIE	Optional
<input checked="" type="checkbox"/> QE_Counter	Optional
<input checked="" type="checkbox"/> Signin	Optional
<input checked="" type="checkbox"/> Test Global Variables	Optional
<input checked="" type="checkbox"/> Who Am I	Optional
<input checked="" type="checkbox"/> User-based Cache	Optional
<input checked="" type="checkbox"/> Role-based Cache	Optional
<input checked="" type="checkbox"/> Application-based Cache	Optional

XYZ Corp
☒ Include all?

<input checked="" type="checkbox"/> Sun: BookmarkPortlet	Optional
<input checked="" type="checkbox"/> Sun: Google Search	Optional
<input checked="" type="checkbox"/> Sun: Notepad Portlet	Optional
<input checked="" type="checkbox"/> Sun: WeatherPortlet	Optional

News
☐ Include all?

<input type="checkbox"/> Excite Business News	Optional
<input type="checkbox"/> Technology News	Optional

Tab Content page

Note. This procedure is performed by the administrator.

To make newly created remote portlets available to end users:

1. On the Tab Content page, select the required portlets, or select the Include all? check box to display all portlets from the portal registry for remote content group box.
2. 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. Select 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 that you want to offer to third-party consumers.

Page Name	Definition Name	Navigation	Usage
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_PROPOROTDTL	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.
Content Ref Security	PORTAL_CREF_SEC	Select 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 components 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 can select WSRP Portlet for Homepage pagelets and Template pagelets. This enables the pagelet's WSRP Producible 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 select Caching Type and Minutes to Cache.

Note. When you create a pagelet using 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.

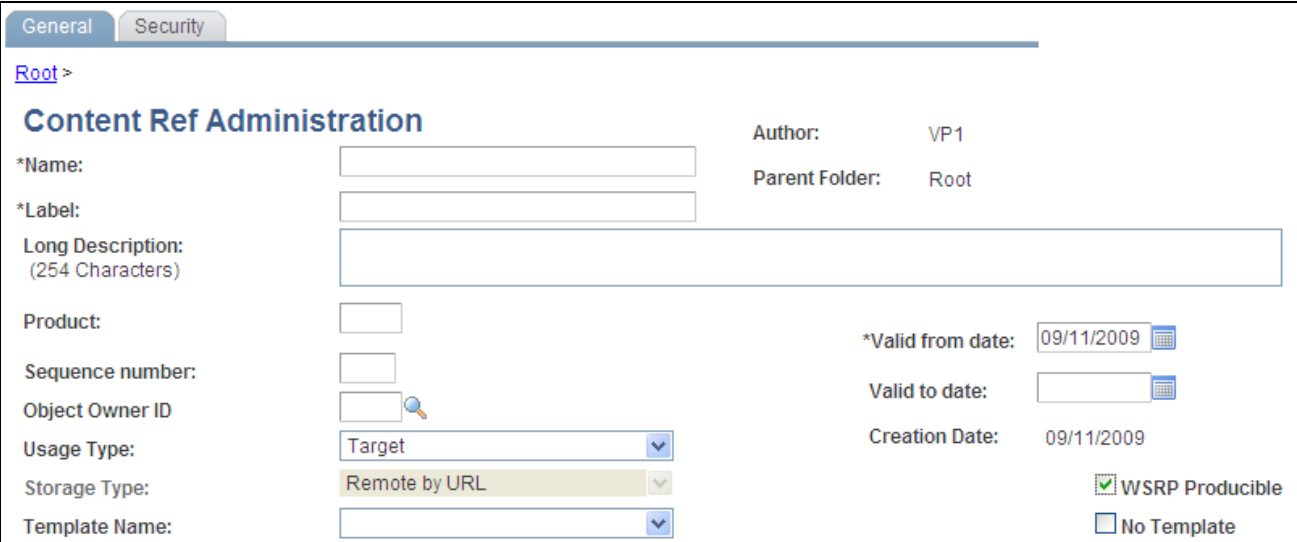
See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Creating Component Definitions," Setting Internet Properties.

Specifying Content References for WSRP Production

While you can also specify pagelets for WSRP production using 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 Pagelet Wizard. This applies also to component-based and iScript-based pagelets

Access the Content Ref Administration page (PeopleTools, Portal, Structure and Content..)



Content Ref Administration page: WSRP Producing option

The WSRP Producing option appears only 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 Producing 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-producing content must originate from the local node, if a content reference is modified to reference a remote node, the WSRP Producing 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 from the default content reference description.	Entered text.
WSRP_WINDOWSTATE	WSRPWindowState	Enables you 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 Producing check box also display the WSRP Producing check box. By default, the option is not selected when you create a new content reference link, even if the WSRP Producing check box is selected for the associated content reference. As with the content reference, if you select the WSRP Producing 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 points:

- Portal registry folders do not display the WSRP Producing 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 be available only 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. (Select 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 Pagelet Wizard, and any content reference that appears on this page has the WSRP Producing option selected on the Content Ref Administration page.

WSRP Production

Producer Offered Portlets

Search Criteria

*Portal Name: EMPLOYEE

Search

Portlet Title: begins with

Producer Offered Portlets

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 |

First 1 of 1 Last

Portal Name	Title	Description	Selected	Details	Path
EMPLOYEE	Demonstration	Demonstration WSRP portlet.	<input checked="" type="checkbox"/>	Details	Root>Portal Objects>Pagelets>QE>

Save

[Producer Details](#)

Producer Offered Portlets page

Portal Name

Choose the portal where the content you want to offer is located, and click Search.

Note. To display portlets within all portals, select *All Portals* and omit the Portlet Title value.

Portlet Title

Enter the portlet title or a portion of it for increased search options. Use the % character for wild card values.

Web Service Endpoint URL

Displays the URL that the WSRP consumer portal requires to import the offered content.

Copyright © 1988, 2011, Oracle and/or its affiliates. All Rights Reserved.

685

Oracle Portal Web Provider URL	Displays the URL that the JPDK-compliant Oracle Portal requires to import the offered content.
---------------------------------------	--

Note. This value applies only for Pagelet Wizard pagelets.

Portlets

This section of the page is named Portlets, despite the fact that 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 that 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 access the Portlet Details page, which displays the generated XML that is 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 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 the same way as a portlet, provided that 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 that is navigated to from a portlet will be rendered in the same browser window but will 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 the PeopleSoft system, the following numbered list 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 whether 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

PeopleTools 8.52: 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 applications provide PeopleSoft WSRP-produced markup references with WSRP style names only, not PeopleSoft styles. If a particular PeopleSoft style is not mapped to a corresponding WSRP style, the style cannot be generated in the WSRP markup. The appearance and character of the style in the WSRP-consuming portal is determined by the WSRP CSS style sheets on the consuming portal.

PeopleSoft-to-WSRP styles 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 causes 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 select 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 sign in 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 link or the Multi-Language drop-down list box will not appear (so multilanguage 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 21

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.
- 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 21, "Working with JSR 168 Portlets," Describing the Portlet Entity Registry, page 693.](#)

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 such as 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 about 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 means of sharing data among portlets of the same portal application.

Window States

The PeopleSoft JSR 168 engine accepts all of the required window states as well as any custom window states. However, although the engine allows any window state, PeopleSoft JSR 168 does not react to them. Portlets always appear to have a *Normal* amount of screen space.

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 8, "Using Portal Caching Features," Implementing Pagelet Caching, page 224](#).

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 user's 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 code 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 code 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 that:

- 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 that you want 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 that you may use will need to retrieve the new service description, which will include the new Java portlets.

Appendix A

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 9, "Configuring the Portal Environment," Configuring Web Profiles, page 238](#)

General Properties

You can now set these properties from the configuration.properties file by using the Web Profile Configuration - General page:

<i>Release 8.43 Configuration Property</i>	<i>Corresponding Web Profile General Field</i>
AuthTokenDomain	Authentication Domain
compressCacheFiles	Compress Response References
compressResponse	Compress Responses
compressMimeTypes	Compress MIME Types (compress Multipurpose Internet Mail Extensions types)

Release 8.43 Configuration Property	Corresponding Web Profile General Field
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 9, "Configuring the Portal Environment," Configuring General Portal Properties, page 242

Security Properties

You can now set these properties from the configuration.properties file by using the Web Profile Configuration - Security page:

Release 8.43 Configuration Property	Corresponding Web Profile Security Fields
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).

Release 8.43 Configuration Property	Corresponding Web Profile Security Fields
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:

Release 8.43 Pstools Property	Corresponding Web Profile Security Field
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 9, "Configuring the Portal Environment," Configuring Portal Security, page 247](#)

Virtual Addressing Properties

You can now set these properties from the configuration.properties file by using the Web Profile Configuration - Virtual Addressing page:

Release 8.43 Configuration Property	Corresponding Web Profile Virtual Addressing Fields
defaultPort	Port field in the Default Addressing group box.
defaultScheme	Protocol field in the Default Addressing group box.

Release 8.43 Configuration Property	Corresponding Web Profile Virtual Addressing Fields
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 9, "Configuring the Portal Environment," Configuring Virtual Addressing, page 254](#)

Cookie Rules

You can now set these elements and attributes from the cookierules.xml file by using the Web Profile Configuration - Cookie Rules page:

Release 8.43 Cookierules.xml Element or Attribute	Corresponding Web Profile Cookie Rules Field
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 9, "Configuring the Portal Environment," Configuring Cookie Rules, page 257](#)

Caching Settings

You can now set these properties from the configuration.properties file by using the Web Profile Configuration - Caching page:

Release 8.43 Configuration Property	Corresponding Web Profile Caching Field
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:

Release 8.43 Browserprops.xml Element or Attribute	Corresponding Web Profile Caching Field
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 9, "Configuring the Portal Environment," Configuring Caching, page 258](#)

Trace and Debug Properties

You can now set these properties from the configuration.properties file by using the Web Profile Configuration - Debugging page:

Release 8.43 Configuration Property	Corresponding Web Profile Debugging Field
connectionInformation	Show Connection Information
debug_inlinejavascript	Show JavaScript Inline HTML
debug_inlinstylesheet	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 9, "Configuring the Portal Environment," Configuring Trace and Debug Options, page 264](#)

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:

8.43 Configuration Property	Corresponding Web Profile Look and Feel Fields
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 9, "Configuring the Portal Environment," Configuring Trace and Debug Options, page 264

PeopleTools 8.52: Global Technology, "Adding New Languages"

Appendix B

Configuring the Application for 8.4x Navigation

The 8.5x navigation is based on iframe templates and features drop-down navigation. The 8.4x navigation is based on frame (HTML) templates and uses menu pagelet navigation. This appendix describes how to reconfigure an 8.5x application for 8.4x navigation.

To reconfigure the application, you must complete the following tasks:

- Reconfigure the default template.
- Reconfigure the navigation pagelet.

Important! HTML templates are deprecated as of PeopleTools 8.50. PeopleSoft applications require iframe templates to function as designed. Features that are not supported in HTML templates include:

Modal windows and prompts

Main Menu drop-down navigation

Related content

WorkCenter pages

SmartNavigation.

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:

The screenshot displays the 'Content Ref Administration' web page. At the top, there are tabs for 'General' and 'Security'. Below the tabs is a breadcrumb trail: 'Root > Portal Objects > Templates >'. The main heading is 'Content Ref Administration'. The form is divided into several sections:

- General Information:**
 - Name:** PT_84X_DEFAULT_TEMPLATE
 - Author:** PTDMO
 - Parent Folder:** Templates (with buttons for 'Copy object' and 'Select New Parent Folder')
 - *Label:** 8.4x default template
 - Long Description:** (254 Characters) 8.4x default frames template
 - Product:** PT
 - Sequence number:** (empty)
 - Object Owner ID:** PPT (with a search icon and 'PeopleTools' text)
 - Usage Type:** Frame template (dropdown)
 - Storage Type:** Remote by URL (dropdown)
 - *Valid from date:** 05/18/2000 (calendar icon)
 - Valid to date:** (empty) (calendar icon)
 - Creation Date:** 05/18/2000
- URL Information:**
 - *Node Name:** LOCAL_NODE (with a search icon)
 - URL Type:** PeopleSoft Script (dropdown)
 - 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:** (empty text area with example: name1=value1&name2=value2)
 - Producer:** (empty text field with a search icon)
 - Portlet:** (empty text field with a search icon)
 - [Producer Details](#)
- Content Reference Attributes:**
 - Name:** (empty text field)
 - Label:** (empty text field)
 - Attribute value:** (empty text area)
 - ☒ Translate
 - [Attribute Information](#)
 - [Delete](#) (button)
 - [Add](#) (button)

8.4X DEFAULT_TEMPLATE Template after reconfiguration

6. In the navigation path, 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:

The screenshot displays the 'Content Ref Administration' web page. At the top, there are tabs for 'General' and 'Security'. The breadcrumb trail is 'Root > Portal Objects > Templates >'. The page title is 'Content Ref Administration'. The 'Name' field is 'DEFAULT_TEMPLATE', the 'Author' is 'PTDMO', and the 'Parent Folder' is 'Templates'. There are buttons for 'Copy object' and 'Select New Parent Folder'. The '*Label' field contains '8.50 default template'. The 'Long Description' field also contains '8.50 default template'. The 'Product' is 'PT'. The 'Sequence number' is empty. The 'Object Owner ID' is 'PPT' with a search icon and the text 'PeopleTools'. The 'Usage Type' is 'Frame template' and the 'Storage Type' is 'Remote by URL'. The '*Valid from date' is '05/18/2000' and the 'Valid to date' is empty. The 'Creation Date' is '05/18/2000'. There is a link 'Add Content Reference'. Below this is the 'URL Information' section with fields for '*Node Name' (LOCAL_NODE), 'URL Type' (PeopleSoft Script), and 'iScript Parameters'. The 'iScript Parameters' section includes fields for '*Record (Table) Name' (WEBLIB_PT_NAV), '*Field Name' (ISCRIP2), '*PeopleCode Event Name' (FieldFormula), and '*PeopleCode Function Name' (IScript_PT_NAV_TPL_FRAME). There is an 'Additional Parameters' field with an example: 'name1=value1&name2=value2'. Below this are fields for 'Producer' and 'Portlet'. There is a link 'Producer Details'. At the bottom is the 'Content Reference Attributes' section with fields for 'Name', 'Label', and 'Attribute value'. There is a 'Delete' button, a 'Translate' checkbox (checked), and a link 'Attribute Information'. There is an 'Add' button at the bottom left.

8.50 DEFAULT_TEMPLATE Template after reconfiguration

Reconfiguring the Navigation Pagelet

To reconfigure the navigation template:

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 the Main Menu.
4. In the Field Name field, delete the existing value and enter *ISCRIP2*.
5. 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 [Search](#)

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: templatetype=html

Example: name1=value1&name2=value2

Menu pagelet after reconfiguration

6. In the breadcrumbs, click the PeopleSoft Applications link.
7. Click the Edit link for Menu - Classic.
8. In the Label field, replace *Menu - Classic* with *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 - Classic pagelet after reconfiguration

11. Stop and restart the application and web servers.

Appendix C

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 8.5x drop-down navigation template.

Converting a frames-based template to an iframe-based template 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 that the current frames-based template 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 specific permissions or roles 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 shown previously.

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 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>
```

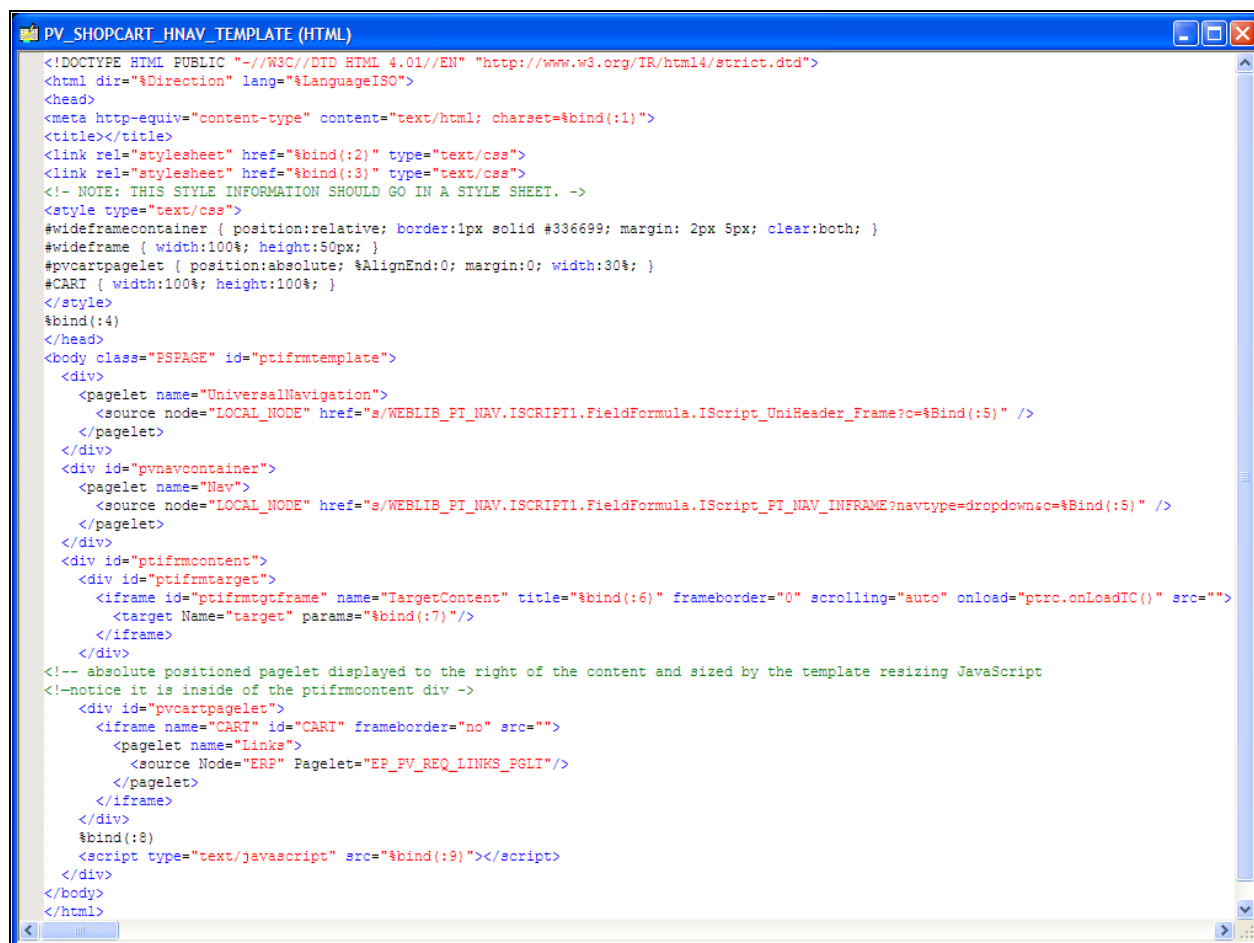
6. Add the style information for the new pagelet. Ideally, these styles should go into a style sheet to take advantage of style sheet caching on the web server and browser. For this example, the embedded style approach places the style in the head section of the template.

```
<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_UniReader_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=dropdownsc=%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_PV.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="ptcrl.onLoadIC()" 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_PV_REQ_LINKS_PG1"/>
</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.

Troubleshooting Errors

This table describes some common errors and possible causes:

Error	Solution
JavaScript errors	<p>Check existing pagelet JavaScript. Make sure to correct statements like this:</p> <pre>parent.parent.frames["UniversalHeader"] /* no longer valid in an iframe template */ parent.frames[3]. /* use a iframe name or an iframe id */</pre>

Appendix D

Replacing SmartNavigation Images

This appendix presents an overview of SmartNavigation icons and discusses how to:

- Create replacement images.
- Define new style classes.
- Modify folder settings.
- Write PeopleCode.

Understanding SmartNavigation Icons

SmartNavigation folder and content reference icons appear next to each folder and menu item element in the SmartNavigation menu hierarchy. The application generates the icons for the folders and menu items by using:

- Image definitions.
- Style sheet definitions.
- PeopleCode function calls.

Image Definitions

SmartNavigation folders and content references use these images by default:



Default folder image.



Default content reference image.

Note these points about the images and style sheets:

- The dimensions of the images must be 14 pixels by 14 pixels.
- You must use PeopleSoft Application Designer to save the images in the portal site.
- You must include a style class for each image—a folder image and a content reference image—in a free form style sheet.
- You must use PeopleSoft Application Designer to save the style sheet in the portal site.

Sample Style Sheet Code

Each style class defines the background style and specifies the image identifier.

```
.fldrdynimage {  
background: transparent url(%Image(Folder_Image_ID)) =>  
no-repeat center center;  
}  
  
.crefdynimage {  
background: transparent url(%Image(Cref_Image_ID)) =>  
no-repeat center center;  
}
```

Note. You do not have to use the same class names that you see in the sample code.

Creating Replacement Images

To create replacement images:

1. Use the graphic application of your choice to create the replacement images.
2. Open PeopleSoft Application Designer in the portal site database and create an image definition for each of the images .

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Creating Image Definitions."

3. Save each image definition.
4. Make note of the image name. You must reference the image name, or image ID, in the style sheet.

Defining New Style Classes

To define new style classes:

1. Open PeopleSoft Application Designer in the portal site database and create a free form style sheet definition.

See *PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide*, "Creating Style Sheet Definitions," Free Form Sub Style Sheet.

2. Enter text to create the style sheet. Use the format that you see here:

```
.fldrdynimage {
background: transparent url(%Image(Folder_Image_ID)) =>
no-repeat center center;
}

.crefdynimage {
background: transparent url(%Image(Cref_Image_ID)) =>
no-repeat center center;
}
```

Note. You must reference the images you created in PeopleSoft Application Designer.

3. Save the style sheet.

Modifying Folder Settings

To modify folder settings:

1. Access the Folder Administration page (Select PeopleTools, Portal, Structure and Content. Click the SmartNavigation folder that you want to change).
2. In the Style Sheet field, enter the name of the style sheet that you created in the previous step.
3. Save the page.

Writing PeopleCode

The code that you write depends upon the data source. This section discusses the PeopleCode for:

- Rowset data sources.
- Tree data sources.

Rowset Data Source

When you use a rowset as the SmartNavigation folder data source, use the following function call in the PeopleCode program to generate folder and content reference icons in the menu:

```
LoadABN(&rs_DataSource, &rs_ChartRowset, &rs_RelatedActions, GetABNNode(&reqParams), =>
GetABNInitialNode(&reqParams), "fldrdynimage", "crefdynimage");
```

This table describes the function parameters:

Parameter	Description
<i>fldrdynimage</i>	Folder image class ID (name)
<i>crefdynimage</i>	Content reference image class ID (name)

The class id parameters are optional. If you want to use the default icons, make one of the two following calls:

```
LoadABN(&rs_DataSource, &rs_ChartRowset, &rs_RelatedActions, =>
GetABNNode(&reqParams), GetABNInitialNode(&reqParams), "", "");
```

```
LoadABN(&rs_DataSource, &rs_ChartRowset, &rs_RelatedActions, =>
GetABNNode(&reqParams), GetABNInitialNode(&reqParams));
```

See Also

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," LoadABN

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNNode

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GetABNInitialNode

Tree Data Source

When you use a tree as the SmartNavigation folder data source and you also use PeopleCode, you use function calls in the program to generate:

- Images for tree nodes (folders).
- Images for tree leaves (content references).

Images for Tree Nodes

When you generate the menu item of a tree node, you must pass both the folder image class name and the content reference image class name because a tree node without children appears as a content reference. Use the following function calls in the PeopleCode program to generate folder icons in the menu and create the HTML fragment that the portal consumes that creates the drop-down menu items:

```
&ChildNode.GenABNMenuElementWithImage(GetABNInitialNode(&reqParams), =>
"fldrdynimage", "crefdynimage");
GenHTMLMenu(&szLI, "folderdynimage");
```

This table describes the function parameters:

Parameter	Description
<i>folderdynimage</i>	Folder image class name
<i>crefdynimage</i>	Content reference image class name

Images for Tree Leaves

When you generate the menu item of a tree leaf, you pass the content reference image class name. Use the following function calls in the PeopleCode program to generate content reference icons in the menu and create the HTML fragment that the portal consumes that creates the drop-down menu items:

```
&ChildLeaf.GenABNMenuElementWithImage("crefdynimage");
GenHTMLMenu(&szLI, "crefdynimage");
```

This table describes the function parameters:

Parameter	Description
<i>crefdynimage</i>	Content reference image class name

Default Images

If you want the default icons to appear, make the following calls in the PeopleCode program to generate folders and content references in the menu and create the HTML fragment that the portal consumes that creates the drop-down menu items:

```
&ChildNode.GenABNMenuElementWithImage(GetABNInitialNode(&reqParams), "", "");
&ChildLeaf.GenABNMenuElementWithImage("");
GenHTMLMenu(&szLI, "");
GenHTMLMenu(&szLI);
```

You must pass an empty string as the parameter for both the folder image class name and the content reference image class name.

Note. You must pass parameter values for the class name into the Tree API methods because the Tree API is a psoftapi object; psoftapi object methods do not allow optional parameters.

The class name parameter is optional in the GenHTMLMenu function. If you want the default icons to appear, make one of the two following function calls:

- GenHTMLMenu(&szLI, "")
- GenHTMLMenu(&szLI);

Oracle recommends that you use the first function call in the previous list because it offers more flexibility in the event that additional parameters are added to the GenHTMLMenu function.

Note. If you use a tree as the datasource *without* PeopleCode, the default images always appear as the folder and content reference elements in the menu.

See Also

PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," GenHTMLMenu

Appendix E

Related Content Reserved Words and Service Parameters

When you have a service using any of the following reserved words as service parameters, they will have some default mapping done when you first access the Map Fields page. Here is the list of related content-specific reserved words, their service parameters, and the default mapping data.

Reserved Words Default Mapping

This table lists the reserved words and their default mapping:

<i>Reserved Word</i>	<i>Description</i>	<i>Default Map Type</i>
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.

Appendix F

Configuring UPK Topics as Related Content

You can configure Oracle User Productivity Kit (UPK) modules and topics as component-level and page-level related content services. This appendix provides a brief overview of Oracle UPK and related content and describes how to:

- Create UPK-based related content service definitions.
- Assign UPK-based services as related content.

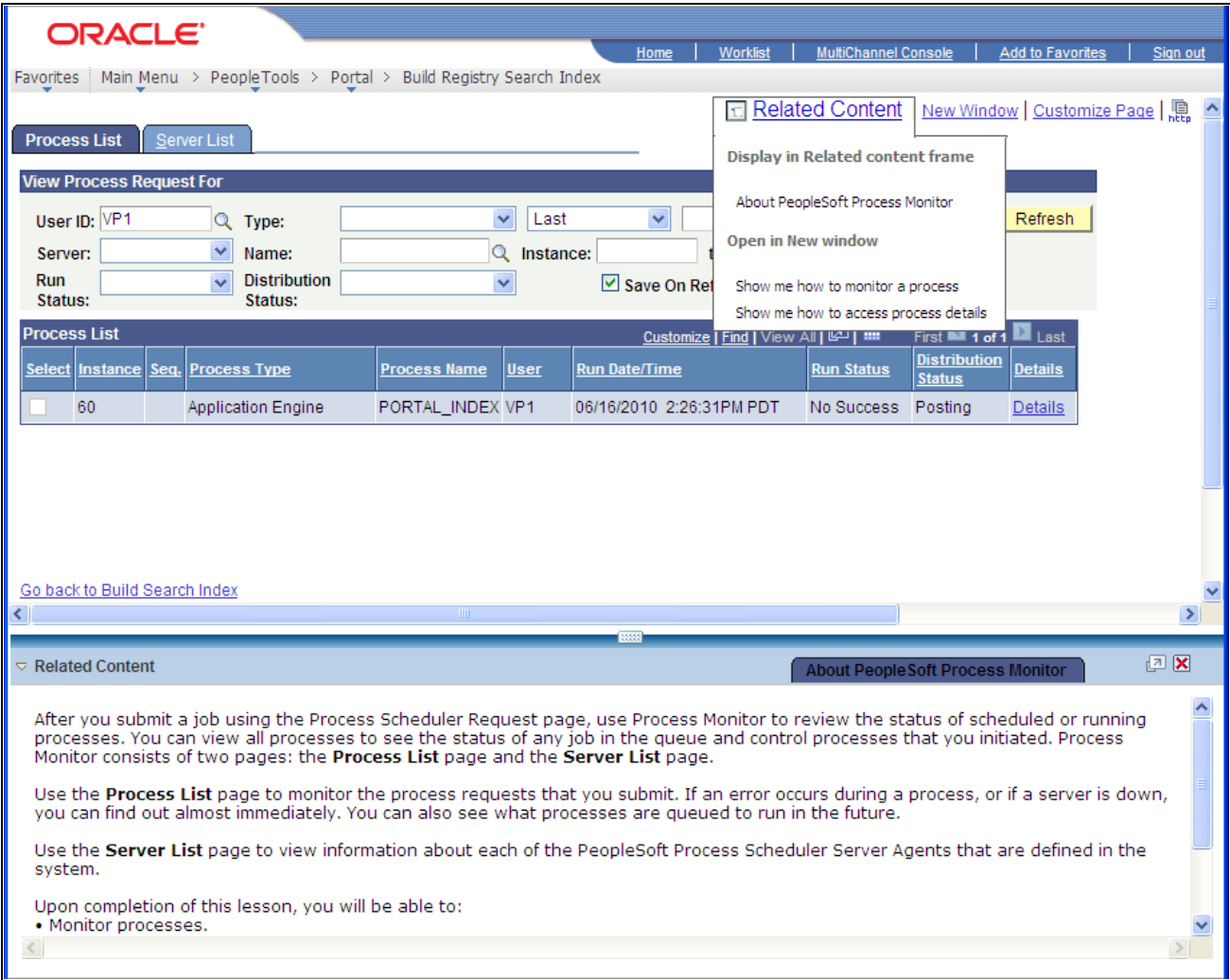
Note. In this appendix, the word *topic* is used to represent both topics and modules unless specified otherwise.

Understanding Oracle UPK and Related Content

Oracle UPK Developer and content is an end-user training solution that provides a single-source development and customization tool to enable rapid content creation and customization. UPK also can provide system process documentation, web-based training materials, role-based performance support, and other types of documentation.

PeopleSoft Related Content Framework is a set of PeopleTools definitions and processes that enable developers and subject matter experts to link application pages with contextually relevant content. The primary goal of related content services is to provide immediate access to relevant, contextual information for any PeopleSoft transaction without requiring any additional user effort. PeopleSoft Related Content Framework acts as an enterprise mashup, tying together all types of content such as Oracle Business Intelligence Enterprise Edition (OBIEE) analytics, queries, discussion threads, documentation, and relevant non-PeopleSoft data into a single tool—PeopleSoft application pages.

You can use the related content framework to set up access to UPK modules and topics so that any UPK content is available at the page or component levels under the Related Content menu as shown here:



UPK links under the Related Content contextual menu with a conceptual topic in the related content frame

Prerequisites

This table lists the necessary prerequisites for integrating UPK and related content and provides references to information about these prerequisites:

Prerequisite	Reference
Perform all related content configurations.	
Create, deploy, and configure UPK player package to your web server.	See Oracle's <i>PeopleSoft Enterprise Help Menu Integration</i> See Oracle's <i>User Productivity Kit In-Application Support Guide</i>

UPK Player Package Links

The UPK administrator will have access to the player package link (URL). The link should look similar to this:

`http://upkservername/upk/content_subject/toc.html`

If you change the player package link ending from *toc.html* to *kp.html* you can access the kp.html file.

This is a sample kp.html page:

PeopleTools 8.52 - Process Scheduler [\[Concept\]](#)

Understanding Process Scheduler [\[Concept\]](#)

- Understanding PeopleSoft Process Scheduler [\[Try It\]](#)[\[Concept\]](#)
- Understanding PeopleSoft Process Scheduler Architecture [\[Try It\]](#)[\[Concept\]](#)
- Understanding PeopleSoft Process Scheduler Components [\[Try It\]](#)[\[Concept\]](#)
- Understanding the Activities to Implement Process Scheduler [\[Try It\]](#)[\[Concept\]](#)

Submitting Process Request [\[Concept\]](#)

- Submitting Process Requests [\[See It\]](#)[\[Try It\]](#)[\[Know It\]](#)[\[Do It\]](#)[\[Concept\]](#)
- Changing Distribution Options [\[See It\]](#)[\[Try It\]](#)[\[Know It\]](#)[\[Do It\]](#)[\[Concept\]](#)
- Submitting a Process from an Application [\[See It\]](#)[\[Try It\]](#)[\[Know It\]](#)[\[Do It\]](#)[\[Concept\]](#)
- Scheduling Process Requests [\[See It\]](#)[\[Try It\]](#)[\[Know It\]](#)[\[Do It\]](#)[\[Concept\]](#)

Using Process Monitor [\[Concept\]](#)

- Monitoring Processes [\[See It\]](#)[\[Try It\]](#)[\[Know It\]](#)[\[Do It\]](#)[\[Concept\]](#)
- Viewing Process Details [\[See It\]](#)[\[Try It\]](#)[\[Know It\]](#)[\[Do It\]](#)[\[Concept\]](#)
- Viewing Report Details [\[See It\]](#)[\[Try It\]](#)[\[Know It\]](#)[\[Do It\]](#)[\[Concept\]](#)
- Viewing PeopleSoft Process Scheduler Server Agents [\[See It\]](#)[\[Try It\]](#)[\[Know It\]](#)[\[Do It\]](#)[\[Concept\]](#)

Sample kp.html page

KP.HTML File Topic Modes

Each UPK installation includes a kp.html file that lists your UPK modules and topics. The kp.html file, which is located in the player package folder, enables you to manually link to a specific mode for a topic.

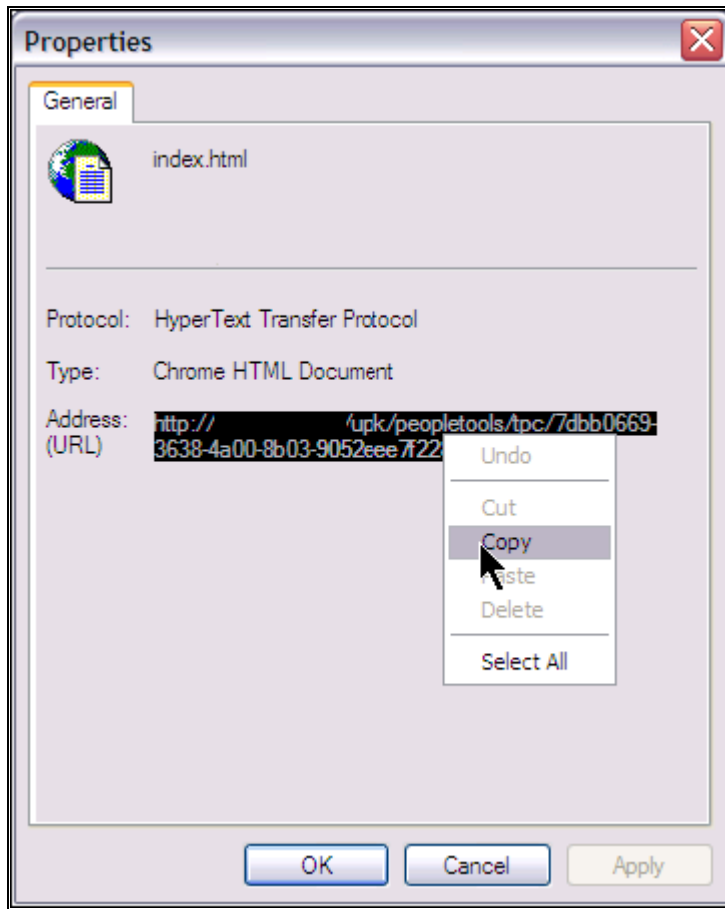
UPK modules can appear in five modes:

- Do It
- Know It
- Try It
- See It

- Concept

Right-click the topic link and select Properties to access the topic-specific URL. If you are creating task-oriented related content, right-click the **Do It** link. If you are creating concept-oriented related content, right-click the **Concept** link.

The Properties dialog box shows the URL in the Address field. Although the fields on the dialog box are display-only, you can highlight and copy the address as this example shows:



UPK topic URL

Note these points about the format of the URLs:

- For task-oriented information, the URL should look similar to this:
`http://upkserver/upk/peopletools/dhtml_kp.html?Guid=69f0376e-70fc-417c-a43c-ecd6afb72b29&Mode=D&Back`
- For conceptual information, the URL should look similar to this:
`http://upkserver/upk/peopletools/tpc/0350d9a9-6013-4cc9-9022-016f815029ba/Parts/index.html`

Use this table to verify that the link accesses the correct mode:

<i>Link Fragment</i>	<i>Mode</i>
Mode=D	Do It
Mode=K	Know It
Mode=T	Try It
Mode=S	See It
index.html	Concept

For each URL, the &Back fragment is extraneous for related content purposes. You delete this part of the URL when you create the service definition.

Service Definitions

You configure UPK topics as non-PeopleSoft URL service definitions.

In this example, you see the View Report Detail UPK topic that is configured as a non-PeopleSoft URL type:

Define Related Content Service

Service Information

Service ID: UPK_PRCMON_VWRPT_D
*Service Name: View Report Detail
Description: Do It Mode
*URL Type: Non-PeopleSoft URL

Write help text
Copy Service Definition

URL Information

*External URL: http://xx.xx.xx.xx/upk/peopletools/dhtml_kp.html?Guid=5455b1b5-1d36-4261-b314-4628215a1fd2&Mode=D
☐ Post mapping definition data
☒ Escape URL Parameters
Note: parameter names are case-sensitive.

Service URL Parameters

	*Parameter Name	Required Flag	*Description
1		<input type="checkbox"/>	

[Show Formed URL](#)
[Test Related Content Service](#)

Display Options

☐ Refresh
☒ New Window

Select Security Options

☒ Public Access
☐ Related Content Provider Security
☐ Related Content Consumer Security

Related content service definition for UPK topic

Related Content Menus and Frames

When you assign UPK services to content references, you assign the services as component-level related content only for these reasons:

- UPK service definitions do not require any parameters; therefore topics cannot be mapped at the field level.
- Users should be able to access UPK content from the Related Content drop-down menu. Only services assigned as related content appear in the Related Content drop-down menu.

Menu Item Labels

When you create a label for the service, consider the specific context of the service and use direct language. For example, if you want to add a service in which the UPK topic shows users how to submit a process in Do It mode, then *Show me how to submit a process* is an appropriate label.

Related Content Frame

If you assign the UPK topics to appear in the right frame, users can access multiple help topics simultaneously because they appear in individual pagelets. If you assign UPK topics to the bottom frame, one topic only is visible at a time because they appear in tabs.

Example UPK Topic in Related Content Frame

In this example, the UPK topic appears in the right related content frame:

The screenshot shows the Oracle Process Monitor interface. The main content area displays a table of process instances. The right-hand side features a 'Related Content' frame with a help topic titled 'Tell me about the Process Monitor'. A context menu is visible over the 'Related Content' frame, showing options like 'Display in Related Content frame', 'Tell me about the Process Monitor', 'Show me how to find the process details?', 'Show me how to find the report details', and 'Show me how to monitor processes'.

Select	Instance	Seq	Process Type	Process Name	User	Run Date/Time	Run Status	Distribution Status	Details
<input type="checkbox"/>	119		SQR Report	XRFRCFL	QEDMO	08/18/2011 4:06:43PM PDT	Success	Posted	Details
<input type="checkbox"/>	113		SQR Report	PTSQRTST	QEDMO	08/17/2011 8:24:20AM PDT	Success	Posted	Details

Example UPK topic as related content on Process Monitor page

See Also

Chapter 15, "Developing and Configuring Related Content Services," Framework Flexibility, page 537

Chapter 15, "Developing and Configuring Related Content Services," Configuring the Related Content Menu, page 593

Chapter 15, "Developing and Configuring Related Content Services," Configuring the Related Content Frames, page 596

Creating UPK-Based Related Content Service Definitions

This section discusses how to:

1. Modify the UPK player package link.
2. Create the UPK-based service definition

Modifying the UPK Player Package Link

To modify the UPK player package link:

1. Contact your UPK system administrator to obtain the player package link (URL).
2. Change the file name *toc.html* to *kp.html* so that the URL looks similar to this:

`http://upkservername/upk/content_subject/kp.html`
3. Paste the link into your browser and access the file.
4. Find the topic. Right-click the **Concept** or **Do It** link associated with the topic. If you are creating task-oriented related content, right-click the **Do It** link; otherwise, click the **Concept** link.
5. From the pop-up menu, select **Copy Shortcut**.

Alternately, select **Properties** and then copy the URL that you see on the dialog box.

6. If you are creating multiple service definitions, you might consider creating a spreadsheet that contains information similar to that which exists in this table::

Topic	URL
Process Scheduler - Concept	<code>http://upkserver/upk/peopletools/tpc/0350d9a9-6013-4cc9-9022-016f815029ba/Parts/index.html</code>
Viewing Process Details - Do It	<code>http://upkserver/upk/peopletools/dhtml_kp.html?Guid=69f0376e-70fc-417c-a43c-ecd6afb72b29&Mode=D&Back</code>

7. For each task-oriented URL, remove the *&Back* at the end of the address.
8. Save your spreadsheet.

Creating the UPK-Based Service Definition

To create the UPK-based service definition:

1. Create a new related-content service definition.

Choose a naming convention that makes UPK service definitions easily identifiable, for example *UPK_<service_definition_name>*

2. Enter the service name and description.
3. Select *Non-PeopleSoft URL* as the URL type.
4. In the External URL field, enter the URL to your kp.html file.

It should look similar to this: `http://myupk.site.com/upk/mycontent/dhtml_kp.html?Guid=5cd5ba9f-c3a8-441f-923b-a8aca7e95b6e&Mode=D`

5. Click the **Test Related Content Service** link.

The **Test Related Content Service** page opens.

6. On the **Test Related Content Service** page, click the **Test Related Content Service** link.
7. Click the **Return** button to go back to the **Define Related Content Service** page.
8. For conceptual information, select the **New Window** check box to give the administrator the ability to enable the new window functionality for the user.
9. In the **Select security options group box**, select the **Public Access** check box.
10. Click the **Save** button.

Note. UPK-based service definitions do not require additional parameters.

This completed service definition shows an external URL that accesses UPK content in Do It mode:

Define Related Content Service

Service Information ?

Service ID

UPK_PRCMON_VWRPT_D

*Service Name

View Report Detail

Description

Do It Mode

*URL Type

Non-PeopleSoft URL

Write help text

Copy Service Definition

URL Information

*External URL

http://xx.xx.xx.xx/upk/peopletools/dhtml_kp.html?Guid=5455b1b5-1d36-4261-b314-4628215a1fd2&Mode=D

☐ Post mapping definition data

☒ Escape URL Parameters

Note: parameter names are case-sensitive.

Service URL Parameters

1

*Parameter Name

Required Flag

☐

*Description

+

-

2

First

1 of 1

Last

Show Formed URL

Test Related Content Service

Display Options

☐ Refresh

☒ New Window

Select Security Options

☒ Public Access

☐ Related Content Provider Security

☐ Related Content Consumer Security

UPK related content service definition

Note. You can use the **Copy Service Definition** button to clone this definition and save time creating new UPK-based service definitions.

See Also

Chapter 15, "Developing and Configuring Related Content Services," *Creating and Maintaining Non-PeopleSoft Related Content Service Definitions*, page 560

Assigning UPK-Based Services as Related Content

Access the Manage Related Content Configuration page. (Select PeopleTools, Portal, Related Content Service, Manage Related Content Service.)

To assign UPK-based services as related content:

734

Copyright © 1988, 2011, Oracle and/or its affiliates. All Rights Reserved.

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

1. Click the Assign Related Content to Application Pages link.
2. Select the content reference.

See Chapter 15, "Developing and Configuring Related Content Services," Selecting a Content Reference, page 578.

3. On the Related Content tab, add the UPK service as component-level related content.

See Chapter 15, "Developing and Configuring Related Content Services," Assigning Component-Level Related Content, page 580.

4. Configure the related content menu.

See Chapter 15, "Developing and Configuring Related Content Services," Configuring the Related Content Menu, page 593.

5. Configure the related content frame.

See Chapter 15, "Developing and Configuring Related Content Services," Configuring the Related Content Frames, page 596.

6. Click the Save button.
7. Click the content reference link to test your configuration.

Appendix G

Configuring OBIEE-Related Content Services

You can configure related content services to display Oracle Business Intelligence Enterprise Edition (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, you must complete certain prerequisites before you assign 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 later.
- 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 on the OBIEE report definition and save the report.
- The sizing of reports needs to be handled on the OBIEE database. The associated grid, width of the page, and centering of the objects cannot be adjusted.
- 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.
- 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 reload the report definitions.

Appendix H

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.

Two options are available:

- 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

Service Operation	Definition	Input Parameter
<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 that is available within the PeopleSoft source system.	Component name as string
<code>getComponentMetadata()</code>	Returns PeopleSoft component URLs of all the components that are 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 that is available as target URLs for an application in which the action links to various report columns are being embedded.

Note. Any PeopleSoft component can be registered under various portal names. For any given component, multiple associated URLs could exist. 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 onetime 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 to consume the WSDLs. Multiple PeopleSoft application installations, such as CRM, HCM, SCM, EPM, and so on can coexist 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 development of OBIEE reports and again after a PeopleTools upgrade to reload the PeopleTools metadata.

Appendix I

Enabling Locale Switching Through Query Parameters in Oracle WebCenter

This appendix discusses how to register provider classes so that you can enable locale switching through query parameters in Oracle WebCenter.

Registering Provider Classes

To support locale switching by using the `WCGS_PREFERREDLANGUAGE=LangCode` query parameter, you must manually register the `URLBaseProvider` class.

You can do that manually or by using a WebLogic Scripting Tool (WLST) command.

Manual Registration

To register the provider class manually:

1. Using a text editor, open the `adf-config.xml` file.
2. Find the `adf-config` element and enter the following xml inside the tag:

```
<adf-webcenter-generalsettings-config xmlns=>
"http://xmlns.oracle.com/webcenter/generalsettings/config">
  <provider-class>
    oracle.webcenter.generalsettings.model.provider.>
URLBaseProviderprovider
  </provider-class>
</adf-webcenter-generalsettings-config>
```

3. Save the file.

WLST Registration

To use WLST to register the provider class:

1. Open the WebLogic Scripting Tool.

2. Run this command:

```
setWebCenterGeneralSettingsProvider('webcenter', =>  
'oracle.webcenter.generalsettings.model.provider.URLBaseProvider')
```

Note. The first parameter is the application name. The second parameter is the providerClass. Sometimes you must supply a third parameter.

or this command:

```
setWebCenterGeneralSettingsProvider('webcenter', =>  
'oracle.webcenter.generalsettings.model.provider.URLBaseProvider', =>  
'WLS_Spaces1')
```

Note. The third parameter is the spaces cluster service name or spaces service name.

3. Restart the server.

See Also

Oracle® Fusion Middleware Web User Interface Developer's Guide for Oracle Application Development Framework 11g Release 1 (11.1.1.5.0), "Getting Started Using the Oracle WebLogic Scripting Tool."
http://download.oracle.com/docs/cd/E21764_01/core.1111/e10105/getstart.htm#CHDGGAE

Appendix J

Configuring OBIEE Admin User Permissions in Enterprise Manager 11g

This appendix discusses how to configure the OBIEE Server Admin user with the correct permissions in Enterprise Manager 11g.

Configuring OBIEE Server Admin User Permissions in Enterprise Manager 11g

This section discusses the necessary permission configuration for using OBIEE.

Note. For use with 11g only.

The user that you configure in the Server Setup page must have *impersonate* permissions in Enterprise Manager. In release 10g, the BI Administrator satisfies this requirement. In release 11g, the user must be assigned the BI System application role.

To enable the BI System application role in Enterprise Manager 11g:

1. Sign in to Webserver Enterprise Manager as an administrator.
2. In the left side navigation, click Business Intelligence, coreapplication.
3. When the new page appears, click Configure and manage Application Roles.
4. When the new page appears, click BI System.
5. In the users column, click Add User and add the same user that you entered as the Admin User on the Server Setup page.
6. Click the OK button to save your changes.

Appendix K

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 the <offeredPortlets> tag, other language strings are inside <resources> tags at the end of xml file.

PeopleSoft applications store 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 multilanguage data is 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 any multilanguage strings (such as cref label, description, and keywords) exist, 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`, the PeopleSoft system tries to find matches for the desired locales to PeopleSoft installed languages using the following rules:

- If the locale is installed, the system returns it.
- If only the parent is installed, the system returns the parent. For example, if the desired locale is *fr-ca* and only *fr* is found, the system returns *fr*.
- If no match is found, the system returns *en* (English).

Appendix L

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 the name of which 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 means of 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:

Web Library Name	Description
WEBLIB_CTI	Contains iScripts for generating the Computer Telephony Integration (CTI) console and the CTI free-seat extension page.
WEBLIB_IB	Contains iScripts for define PeopleSoft Integration Broker functions.
WEBLIB_MCF	Contains iScripts for generating 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.

Web Library Name	Description
WEBLIB_OPT	Contains iScripts for generating Optimization Framework functions.
WEBLIB_PORTAL	<p>Contains the following six fields with FieldFormula iScript PeopleCode, each of which relates to a functional area of the portal:</p> <ul style="list-style-type: none"> • PORTAL_DYN_TEMP: Support for the dynamic template. • PORTAL_HEADER: Support for the header portion of the page and some generic routines. • PORTAL_HOMEPAGE: Support for homepage runtime interaction, including the homepage version of the menu navigation. • PORTAL_NAV: Main support routines for side navigation. • PORTAL_PGLT_PREV: Support for the pagelet preview functionality. • PT_WSRP: Used to sign in to the PeopleSoft application when it is defined as a producer.
WEBLIB_PPM	Contains iScripts for generating Performance Monitor functions.
WEBLIB_PTAF WEBLIB_PTAFEMC	Contains iScripts for generating 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_PTIFRAME	Contains iScripts for portal favorites and popup functionality.
WEBLIB_PTOBIEE	Contains iScripts for displaying Business Intelligence (Analytics) pagelets.
WEBLIB_PTPP_SC	Contains iScripts for generating navigation pages and the Main Menu navigation pagelet.
WELIB_PTPP_PGT	Renders navigation pagelets without Pagelet Wizard.

Web Library Name	Description
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_PTRC	Contains iScripts for displaying related content.
WEBLIB_PTRTE	Contains iScripts for uploading inline images in rich text.
WEBLIB_PT_NAV	Contains iScripts for the menu pagelet, drop-down navigation, and left-hand navigation for transaction pages.
WEBLIB_PT_TPL	Contains iScripts for portal template processing.
WEBLIB_QUERY	Contains the iScripts for generating 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	Contains iScripts for generating software development kit functions.
WEBLIB_SDK_BI	Contains iScripts for generating 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	Contains iScripts for generating an inactivity timeout warning script.
WEBLIB_XMLLINK	Contains iScripts for generating PeopleSoft Business Interlinks XML functions.

Index

Symbols

<%=resultDoc%> session variable 272

A

accelerators, SSL *See* SSL accelerators

action bars

creating 190

hiding 306

HTML objects for 195

Add Folder page 68

Add Link page 75

addressing, virtual *See* virtual addressing

anchor tags 39

API

Java Portlet 692

Application Based Navigation (ABN)

See SmartNavigation

application classes, pagelet search 345

application class URL types

creating 557

URL information 557

Application Designer

changing menu definitions 147

creating/storing application objects 8

designing pagelets based on pages 483

inheriting style classes 217

understanding 8

viewing iScripts 206

Application Engine programs

PORTAL_CSS 169

PORTAL_INDEX 187

PORTAL_MENU

See Also Menu Import process

Application Navigation *See* SmartNavigation

application objects 8

Application Portal

importing menus 285

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 *See Also* handlers

installing Process Scheduler 16

Jolt *See Also* Jolt

listeners *See Also* listeners

managing memory consumption 228

processes *See Also* server processes

queues *See Also* queues

relationship to database servers 8

server processes *See Also* server processes

services *See Also* services

Tuxedo *See Also* Tuxedo

understanding 9, 16

Approval Framework web library 748

ASF 645

Assign Related Content page 580, 584

attributes

content reference 150, 506

folder 113

pagelet 506

authentication domains

defining 236

selecting for portals 244

understanding 236

authentication token *See* PS_TOKEN

AuthTokenDomain 257

autocomplete

disabling 276

B

banner pagelets 511

base-level authentication domains 236

batch servers 15

bind variables

displaying error messages 270

using for images, tabs and pagelets 197

using in templates 204, 209, 214

using the PORTAL_HP_COMPONENT

object 190

breadcrumbs 52

browsers

applying timeouts to individual windows 261

browser-based caching 229

caching 221

compressing communication with web

servers 244

configuring caching 259

setting state options 232, 259

understanding 18

BuildStaticHPTemplate function 190

Business Interlinks

using 22

Business Interlinks web library 749

C

Cache Enabled Value 58

caching

browser-based caching, understanding 229

browsers, configuring on 258

compressing cache files 245

configuring directories 263

configuring for portals 6, 221, 698

configuring on the web server 262

homepage 230

homepage, default 222

JSR 168 portlet content 692

metadata 227

navigation pagelet 231

page 232

pagelet *See Also* pagelet caching

- pagelet refreshing, disabling 508
 - proxied JavaScript 229
 - server-based 222
 - target content 151, 223
 - understanding 221
 - user IDs 248
 - web server-based navigation 233, 262
 - web server memory 223
- caching, enabling for navigation pages 47
- cascading style sheet (CSS) *See* style sheets
- certificates, installing 658
- character set encoding
 - overriding for content references 151
 - overriding for portal pages 283
- Clean Portal Project page 92
- Clean Portal Project process
 - running 92
 - understanding 89
- collaborative content 533
- colors
 - changing for links 200
 - enabling color attributes in table layouts for pages 265
- COM 28
- component-based pagelets
 - guidelines 624
- Component Interfaces 22
- Component Object Model (COM) 28
- components
 - adding default 190
 - including in portal navigation 287
 - most recently used 175
 - registering in multiple portals 170
 - registering multiple times 170
 - URL format for 34, 36
- Computer Telephony Integration (CTI) console
 - web library 747
- connector 336
- consumer properties 675
- consuming remote portlets
 - language support for 745
- Content Ref Administration page 678, 686
 - adding/editing templates 179
 - content references, defining 145
 - content references/folders, moving 166
 - pagelet attributes, managing 507
 - pagelet Help, configuring 510
 - pagelet timeouts, configuring 510
 - tabs, adding 303
 - tabs, creating 196
- content reference
 - attributes 509
- content reference attributes 150
 - KEYWORD 150
 - listing of 150
 - NAVNEWWIN 151
 - NAVTOPWIN 151
 - PORTAL_BANNER_PGLT 151
 - PORTAL_ENCODING_OVERRIDE 151
 - PSCACHECONTROL 151
 - PSTIMEOUT 151
 - PT_PORTAL_PGLT_HDR_IMG 152
 - PTPGLTDEFERLOAD 151
 - PTPPB_PAGELET_HELP_LINK 152
 - RCF_SERVICE_PARAMETER 152
 - RELLINK 152
- Content Reference Link Administration page 160
- content reference links
 - copying attributes 163
 - defining 160
 - defining attributes 163
 - deleting 162
 - setting link information 162
 - setting permissions 164
 - setting target information 162
 - understanding 162
 - using URLs for 162
- content references
 - accessing 31
 - adding internal PeopleSoft 147
 - copying 166
 - creating 676
 - creating content reference groups 155
 - defining 145
 - deleting 168
 - links *See Also* content reference links
 - moving 166
 - registering 31
 - related links *See Also* related links
 - reviewing 678
 - reviewing permission list security 159
 - reviewing permissions 158
 - reviewing role security 159
 - reviewing security 158
 - setting attributes 150
 - setting permissions 157
 - setting URL information 149
 - specifying pagelet caching 225
 - synchronizing portal object security 168
 - testing 159
 - type parameters 152
 - understanding 27
 - using the KEYWORD attribute 187
 - viewing for a folder 182
 - viewing for WSRP portlets 686
 - viewing permission list, role and user access 184
- Content Reference Security page 157
- content servlets
 - accessing unwrapped content 35
 - proxying and relative URLs 37
- Context Manager Home custom navigation page 46
- Context Manager Home navigation page 49
- Context Manager web library 748
- cookies
 - allowing session cookies 19
 - authentication 18
 - checking for duplicate 276
 - configuring rules 257
 - configuring rules for PeopleSoft portal 698
 - deleting 258
 - passing to servers 257
 - proxying 258
 - PS_TOKEN *See Also* PS_TOKEN
 - securing 258
 - selecting the cookies required page 272
 - sharing 236
- Copy Related Content Service Definition page 568
- CSS *See* style sheets
- CTI console web library 747
- custom navigation pages
 - describing 46
 - describing folder and link sequence on 47

D

- dashboard components web library 748
- dashboard pages
 - See* dashboards, dashboards, dashboards
 - administering 301
- dashboards 302
 - creating 313
 - creating menu links 317
 - labelling 304, 305
 - managing 312
 - naming 304, 305
 - properties 314
 - registering in portal 306
 - selecting content 308
 - specifying layout 309
 - specifying security 316
- dashboard security 307
- databases
 - application server domains
 - See Also* application server domains
 - connectivity software 15
 - defining portals 104
 - importing menus 285
 - servers *See Also* database servers
 - supporting portal registries 28
- database servers
 - configuring 20
 - logical server configuration 20
 - understanding 8
- data source
 - selecting an IB 362
 - selecting a URL 371
 - selecting IB connector 361
 - setting up default gateway 336
 - setting up Pagelet Wizard IB 337
 - setting up security 336
 - specifying IB Connector parameters 378
 - specifying URL parameters 391
- debugging options, configuring 264, 700
- default template
 - for WebCenter services 607
- Define IB Connector Security - Connectors page 336
- Define Related Content Service page
 - application class URL type 557
 - embeddable pagelet URL type 562
 - non-PeopleSoft URL type 560
 - OBIEE pagelet URL type 563
 - PeopleSoft component URL type 561
 - PeopleSoft Query URL type 564
 - PeopleSoft script URL type 565
- definitions
 - application object 8
 - menu 147, 284
 - menu group 287
 - node 25, 279
 - 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
- delivered services
 - Oracle WebCenter 600
- DEV web profile 238

- Diagnostic Framework web library 748
- disconnect URL format 35
- DNS 237
- Domain Name Server (DNS) 237
- domains
 - application server
 - See Also* application server domains
 - authentication
 - See Also* authentication domains
- drop-down menu
 - recently used components 175
 - sort, disabling 175
- drop-down navigation
 - enabling 175
 - example 23
- dynamic templates
 - frame-based example 206
 - iframe-based example 210
 - understanding 204

E

- Edit Collection 52
- Edit folder 52
- Edit Link page 79
- EIP WSDL web library 747
- embeddable pagelet URL types
 - creating 562
 - example 562
 - service URL parameters 562
 - URL information 562
- Enable Drill Down Cache 58
- encryption and signature 658
- encryption enabling 659
- errors
 - accessing content references 31
 - checking for duplicate cookies 276
 - configuring error pages 270
 - displaying/hiding for pagelets 220
 - selecting error pages for unavailable pagelets 270
 - selecting error pages for unavailable target content 270
 - selecting pages for MCF authorization failure 270
 - selecting pages for user authentication failure 272
 - selecting signon error pages 272
 - sending PUT requests to portal servlets 29
 - setting authentication domains 236
 - setting target parameters for template pagelets 216
 - setting the base physical path 246
 - understanding causes 270
 - unlocking web profile accounts 239
 - using SSL accelerators 290
- Example for Images page 50
- exceptions
 - logging for Jolt 266
 - selecting pages for Java exceptions 270
 - using mutual authentication 276
- expiration page 269
- extended authentication domains 236
- external URL format 34

F

- favorites
 - system-level options 175
- favorites, adding 27
- file URL format 35
- filters
 - related content 576
- Find an Existing Collection page 64
- firewalls
 - configuring web servers 21
 - using with reverse proxy servers 293
- Folder Administration page 111
- folder attributes 113
- folder configuration
 - SmartNavigation 115
- folder navigation
 - clearing non-default overrides 174
 - disabling 174
- folder navigation pages 100
 - portal-level enabling 106
- folders
 - copying 166
 - defining 111
 - defining attributes 113
 - deleting 168
 - hiding from users 102
 - moving 166
 - organizational structure 98
 - Portal Objects folder
 - See Also* Portal Objects folder, Portal Objects folder
 - root folder 27
 - setting menu visibility 185
 - setting permissions 114
 - SmartNavigation 115, 138
 - synchronizing portal object security 168
 - understanding 27
 - viewing menu folder structure information
 - 182
 - viewing permission list, role and user access
 - 184
- folder security
 - setting security 114
- Folder Security page 114
- forms
 - combining with frames 220
 - converting form tags 39
 - posting data from frame-based templates 219
- frame-based templates
 - constructing URL references 218
 - dynamic example 206
 - handling 30
 - posting form data 219
 - static example 205
 - understanding 204
- frames
 - browser caching 229
 - combining with forms 220
 - frame-based templates
 - See Also* frame-based templates, frame-based templates
 - iframe-based templates
 - See Also* iframe-based templates, iframe-based templates
 - using single signon 282
- frames-based templates

- converting to iframes 709
- functions
 - BuildStaticHPTemplate 190
 - displaying JavaScript used in pages 266
 - forcing refreshes 201
 - generating URL references 218
 - web libraries 747

G

- Gateways page 670
- General Settings page 172
- GenerateXXXXPortalRelURL function 218
- GenerateXXXXPortalURL function 218
- global variables 236
- guest access 252

H

- handlers
 - Jolt server 12
 - understanding 10
 - workstation 12
- headers
 - custom response 218
 - hiding pagelet buttons 227, 508
 - HTTP header caching information 230
 - PortalRegisteredURL 218
 - universal navigation header
 - See Also* universal navigation header
 - using custom portal 199
 - using UsesPortalRelativeURL 37
- Help
 - configuring for pagelets 510
 - creating pagelet 152
 - text for related content 566
- help, online
 - accessing homepage help 306, 508
 - setting the URL 244
- homepage
 - enable renaming 305
- Homepage Generation Service 198
- homepages 301
 - adding default components 190
 - administering 301
 - arranging the tab sequence 311
 - caching 230
 - caching, default 222
 - caching on browsers 262
 - constructing 191
 - constructing headers 193
 - creating 303
 - creating and managing 302
 - creating three-column 190
 - creating two-column 190
 - Homepage Generation Service 198
 - inserting custom tab images 151
 - navigating 23
 - opening 26
 - overriding images/objects 151, 196
 - registering 303
 - selecting content 308
 - sequencing 305
 - specifying layout 309

- tabs 303
 - URL format for 35
 - using predefined HTML objects 191
- homepage security 307
- homepage tabs *See* homepages
- registering *See* homepages
- hosts file
 - configuring for firewalls 294
 - configuring for load balancing 297
 - configuring for SSL accelerators 296
- HTML
 - areas 204
 - caching target content 262
 - catalog 177
 - content supported for content references 152
 - generating for testing 266
 - objects *See Also* HTML objects
 - pagelets *See Also* pagelets
 - template pagelets *See Also* template pagelets
 - templates *See Also* templates
 - using tables for page layout 204
 - viewing/debugging for pages 266
- HTML objects
 - available 190
 - constructing headers 193
 - constructing pagelets 195
 - creating tabs 196
 - overriding default 151, 196
 - PORTAL_HP_ACTIONBAR 195
 - PORTAL_HP_ACTIONBAR_EDIT 195
 - PORTAL_HP_ACTIONBAR_HELP 195
 - PORTAL_HP_ACTIONBAR_REFRESH 195
 - PORTAL_HP_ACTIONBAR_REMOVE 195
 - PORTAL_HP_COMPONENT 195
 - PT_HP2_BANNER 195
 - PT_HP2_PGLT_HDR_IMG 195
 - PT_HP2_PGLTHNDL 195
 - PT_HP2_PGLTHNDL_DRAG 195
 - using predefined homepage 191
 - using predefined template 189
- HTTP_PROXY_NONPROXY_HTTPHOSTS
 - environment variable 293
- HTTP_PROXY_NONPROXY_HTTPSHOSTS
 - environment variable 293
- HTTPS
 - improving same-server performance under SSL 288
 - overriding the protocol for URL references 255
 - specifying for portals 249
 - using SSL accelerators 290
- HTTPS protocol
 - using to communicate 657
- hyperlinks *See* links
- I**
- IB *See* Integration Broker
 - selecting a data source 362
 - setting up Pagelet Wizard 337
 - setting up security 336
- IB Connector
 - defining security 336
 - selecting data source 361
 - setting up security 336
 - specifying data source parameters 378
- IBM WebSphere *See* WebSphere
- iframe 204
- iframe-based templates
 - converting from frames-based templates 709
 - dynamic example 210
 - handling 30
- images
 - adding to tabs 197
 - configuring cache directories 263
 - inserting on homepage tabs 151
 - objects 196
 - overriding default homepage 151, 196
- images, navigation collection 50
- Import Producer Information page 672
- indexes, registry search 186
- information, producer 672
- inline frames 204
- installing certificates 658
- Integration Broker, using 22
- Integration Broker web library 747
- International Standards Organization (ISO) locale codes 274
- Internet scripts *See* iScripts
- internet technology *See* portal technology
- iScripts
 - configuring start pages 269
 - designing pagelets based on 484
 - navigation 37
 - setting parameters for content references 149, 179
 - setting parameters for pagelets 509
 - understanding web libraries 747
 - using dynamic templates 204
 - using frame-based templates 206
 - using iframe-based templates 210
- ISO locale codes 274
- J**
- Java
 - applying load balancing 297
 - controlling proxy server settings 292
 - encrypting passwords (PSCipher) 241
 - JavaScripts *See Also* JavaScripts
 - JSR 168 *See Also* JSR 168
 - portlets 691
 - selecting pages for exceptions 270
 - servlets *See Also* Java servlets
 - setting Java Virtual Machine properties 292
 - web servers *See Also* web servers
- Java Portlet API 1.0 692
- JavaScripts
 - caching proxied 229, 262
 - configuring cache directories 263
 - converting URL references 40
 - displaying functions used in pages 266
 - setting authentication domains 236
- Java servlets
 - portal servlets *See Also* portal servlets
 - understanding 16
- JCP 645
- Jolt
 - logging exceptions 266
 - server listeners/handlers 12

- setting web server options in web profiles 253
 - transmitting requests and data 17
 - understanding 14
- JSH 12
- JSL 12
- JSR 645
- JSR 168 645
 - configuring the portal container 692
 - describing the portlet entity registry 693
 - Java Portlet API 692
 - PeopleSoft compliance 691
 - portlets *See Also* JSR 168 portlets
 - setting up 6
 - understanding 691
- JSR 168 portlets
 - caching content 692
 - understanding 691
 - understanding modes 692
 - window states 692
- JVM 292

K

- Keystore Password 660
- keywords
 - associating to content references 187
 - using keywords to search for CREFs 150
- KIOSK web profile 238

L

- labels
 - content reference attributes 150
 - dashboards 305
 - labelling dashboards 304
 - labelling portals 104
 - on homepage tabs 305
- language support 274
 - rules for consuming and producing remote portlets 745
- 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 276
- libraries
 - web 747
- licensing
 - homepages 301
 - portal technology products xxiii
- links
 - changing colors 200
 - content reference
 - See Also* content reference links
 - proxying 38
 - related *See Also* related links
 - XML 254

- listeners
 - Jolt server 12
 - understanding 10
 - workstation 12
- load balancing
 - applying 296
 - overriding the port/server name for URL references 256
- Local Gateway URL 670

M

- Main Menu Collection Name 55
- Manage Related Content Configuration page 578, 589
- mapping related content service parameters 588
- mashup 533
- Maximum Child Links 56
- Maximum Columns 56
- MCF
 - selecting pages for authorization failure 270
- Menu Folder Structure page 182
- Menu Import page 285
- Menu Import process
 - importing menu groups 285
 - monitoring the status 287
 - selecting components to include in portal navigation 287
 - understanding 284
 - using with Application Portal 285
- Menu Item Details page 180
- menu navigation
 - drop-down 175
 - left 175
- menus
 - enabling folder navigation 106
 - importing groups into the portal registry
 - See Also* Menu Import process
 - navigation 26
 - reviewing item details 180
 - setting visibility 185
 - viewing associated permission lists 186
 - viewing folder structure information 182
 - viewing permission lists, roles and users with menu item access 182
 - viewing security details 184
- Menu Security page 184
- message signature, enabling 659
- metadata
 - caching 227, 262
 - understanding 8
- Microsoft Windows *See* Windows
- Move to New Parent Folder 75
- Multi-channel Framework (MCF) *See* MCF
- MultiChannel Framework web library 747
- My Favorites
 - enabling and disabling 175

N

- navigation
 - drop-down 23
 - iScripts 37
 - selecting components to include 287

- understanding 26
 - URL format 35
 - web server-based caching 233, 262
- navigation collection options
 - understanding 53
 - understanding default values for 53
- navigation collection properties
 - configuring WorkCenter pages 621
- navigation collections
 - adding folders to 68
 - adding links to 75
 - creating 65
 - editing 65
 - editing folders in 74
 - editing links in 79
 - maintaining 61
 - publishing 81
 - publishing as a page 49
 - publishing as a pagelet 50
 - publishing as pagelets 621
 - searching for 64
 - selecting a source folder 71
 - selecting a source link 77
 - selecting images for 73
 - understanding 48
 - using images 50
- navigation folders
 - types 108
- navigation pagelet 50
- navigation pagelets
 - caching 231
- navigation pagelets web library 748
- navigation pages
 - describing common elements on 43
 - describing common elements used on 52
 - enabling caching for 47
 - enabling folder navigation 106
 - enabling for folders 106
 - understanding 43
 - using breadcrumbs on 43
- navigation pages web library 749
- Node Definitions page 281
- Node Name
 - default values 58
- nodes
 - applying templates for partial path 219
 - configuring node-specific template
 - information 176
 - setting 279
 - specifying types 281
 - understanding 25, 279
- non-PeopleSoft URL types
 - creating 560
 - example 560
 - service URL parameters 560
 - URL information 560
- numeric message variables 271

O

- OBIEE
 - configuring for related content 737
- OBIEE pagelets
 - related content 563
- OBIEE pagelet URL types
 - creating 563
 - service URL parameters 563
 - URL information 563
- OBIEE web library 748
- objects
 - application 8
 - Component Object Model (COM) 28
 - content references
 - See Also* content references
 - HTML *See Also* HTML objects
 - image 196
 - PeopleTools 103, 113
 - portal *See Also* portal objects
 - Portal Objects folder
 - See Also* Portal Objects folder
- online help *See* help, online
- operational dashboard 306
- Optimization Framework
 - PSOPTENG server process 14
- Optimization Framework web library 748
- options 53
- Oracle Jolt *See* Jolt
- Oracle Tuxedo *See* Tuxedo
- Oracle WebCenter Blogs service 601
- Oracle WebCenter Discussion service 601
- Oracle WebCenter Documents service 601
- Oracle WebCenter Links service 602
- Oracle WebCenter services
 - Blogs 545, 601
 - Discussion 545, 601
 - Documents 545, 601
 - Links 546, 602
 - Wiki 546, 602
- Oracle WebCenter Wiki Service 602
- Oracle WebLogic *See* WebLogic
- Owner ID 58

P

- page-based templates
 - assembling pages 30
 - handling 29
 - proxying 38
 - static example 205
 - understanding 204
- page layout
 - WorkCenter pages 631
- pagelet
 - attributes 508
 - icons in header 152
 - portal banner 151
- pagelet caching
 - caching personalized pagelets 226
 - disabling manual pagelet refresh 227
 - implementing 151
 - specifying in content references 225
 - specifying in templates 224
 - understanding 224
- pagelet display options, search box 396
- pagelet header
 - adding images 512
 - adding images to 152
- pagelet load
 - deferring 151
- pagelets
 - accessing online help 508

- action bars *See Also* action bars
- administering 301
- banner 511
- building 444
- caching *See Also* pagelet caching
- configuring Help 510
- configuring refresh attributes 511
- configuring timeouts 510
- constructing 189, 190, 195
- content reference attributes 509
- dashboards, selecting for 308, 319
- description 444
- designing based on iScripts 484
- designing omponent-Based 483
- displaying/hiding error messages for 220
- extensions 446
- forcing the refresh 508
- header icons 445
- hiding the minimize button 508
- hiding the refresh button 508
- homepages, deleting from 311, 322
- homepages, selecting for 308, 319
- managing attributes 507, 512
- Pagelet tag 31
- personalization page, selecting the 509
- publishing to another portal 87
- publishing to multiple portals 85
- publishing to other portals 84
- running location 513
- selecting error pages for unavailable pagelets 270
- setting display options 309, 320
- setting names in templates 219
- setting the number of columns 508
- specifying layout 310, 322
- template *See Also* template pagelets
- understanding URLs 36
- using refresh tags 513
- pagelet search
 - application classes 345
 - setting default options 396
- Pagelet tag 31
- Pagelet Wizard
 - copying pagelets 434
 - defining pagelet categories 354
 - defining transformation types 338
 - defining XSL prototypes 346
 - deleting pagelets 434
 - describing display sizes 332
 - generating pagelet export and import scripts 437
 - overview 323
 - publishing a pagelet to multiple portals 85
 - publishing pagelets to another portal 87
 - publishing pagelets to other portals 84
 - reviewing pagelet details 440
 - setting up data source 337
 - setting up default gateway 336
 - setting up IB connector data source security 336
 - setting up the Integration Broker 336
 - specifying pagelet information 356
 - specifying publishing options 419
- Pagelet Wizard - Specify Publishing Options page 682
- Pagelet Wizard data source parameters
 - specifying 372
 - specifying for navigation collections 380
 - specifying for pivot grids 382
 - specifying for queries 386
 - specifying for search records 388
 - using system variables as 373
- Pagelet Wizard data sources
 - describing 325
 - selecting 358
 - selecting a navigation collection 364
 - selecting a pivot grid 367
 - selecting a query 366
 - selecting a rowset 368
 - selecting a search record 369
 - selecting HTML 359
- Pagelet Wizard data types, defining 338, 342
- Pagelet Wizard display formats
 - defining 338, 340
 - describing 326
 - describing the chart format 328
 - describing the custom format 331
 - describing the list format 327
 - describing the menu format 329
 - describing the passthru format 330
 - describing the search list format 329
 - describing the table format 327
 - selecting 393
- Pagelet Wizard display options
 - specifying 395
 - specifying data links 416
 - specifying for charts 410
 - specifying for customs 414
 - specifying for lists 409
 - specifying for menus 406
 - specifying for passthurs 403
 - specifying for search lists 405
 - specifying for tables 400
- Pagelet Wizard footers
 - defining 348, 352
 - defining images 353
- Pagelet Wizard headers
 - defining 348, 349
 - defining images 351
- Pagelet Wizard pagelet types
 - describing 332
 - describing the embeddable type 334
 - describing the homepage type 332
 - describing the template type 333
- pagelet wizard portlets web library 749
- Pagelet Wizard post-transformation processing
 - formatting currency elements 429
 - formatting date elements 429
 - formatting datetime elements 429
 - formatting number elements 429
 - formatting time elements 429
 - generating charts 431
 - inserting content reference links 430
 - inserting content reference URLs 431
 - inserting footers 431
 - inserting headers 431
 - inserting message catalog text 428
 - understanding 427
- Pagelet Wizard search options
 - specifying 396
- pagelet wizard web library 749
- pages
 - assembling via HTML objects 189
 - assembling via templates 25, 29, 30
 - caching 232, 259
 - combining forms and frames 220

- configuring error pages 270
- configuring expiration pages 269
- configuring start pages 269
- creating via HTML objects 190
- diagnosing layout problems 266
- enabling border/color attributes in table layouts 265
- inheriting style classes 217
- inserting style sheets into page HTML 266
- overriding character set encoding 283
- PTCS_CRFURL_SELECT 579
- PTCS_SERVICECFG 580, 584
- PTCS_SERVICECFGRC_SEC 588
- PTCS_SRVCFG_SRCH 578, 589
- selecting for authentication token enable 271
- selecting for passwords 273
- selecting for setting trace parameters 271
- selecting for signon/signoff 272
- selecting for user authentication failure 272
- selecting the cookies required page 272
- selecting the SSL required pages 272
- selecting the user profile page 272
- template pagelets based on 216
- viewing/debugging HTML 266
- parallel pagelet loading
 - disabling 151
- Parent Folder 75
- passwords
 - default for web profiles 239
 - encrypting for web profiles 241
 - selecting pages for 273
 - setting for public users 253
 - setting for web profiles 253
 - setting for XML Link users 254
- PeopleSoft Application Designer
 - See* Application Designer
- PeopleSoft Application Portal
 - Related Discussion service 544
 - Related Links service 544
 - Related Tags service 544
- PeopleSoft Business Interlinks
 - See* Business Interlinks
- PeopleSoft Component Interfaces 22
- PeopleSoft component URL types
 - component parameters 561
 - creating 561
 - example 561
- PeopleSoft Integration Broker
 - See* Integration Broker, Integration Broker
- PeopleSoft OBIEE pagelet URL types
 - example 563
- PeopleSoft Optimization Framework
 - See* Optimization Framework
- PeopleSoft Performance Monitor
 - See* Performance Monitor
- PeopleSoft portal
 - configuring 235, 695
 - configuring caching 698
 - configuring cookie rules 698
 - configuring general properties 695
 - configuring look and feel 701
 - configuring security properties 696
 - configuring tracing/debugging 700
 - configuring virtual addressing 697
 - 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 288
 - installing 4
 - overriding the port/protocol/server name for URL references 255
 - single signon *See Also* single signon
 - understanding 3, 7
 - unwrapped content URL formats 35
 - URL formats 34
 - using SSL accelerators 289
 - web servers *See Also* web servers
- PeopleSoft Query URL types
 - creating 564
 - example 564
 - service URL parameters 564
 - URL information 564
- PeopleSoft Related Content Framework
 - understanding 533
- PeopleSoft script, related content 565
- PeopleSoft script URL types
 - creating 565
 - example 565
 - iScript parameters 565
 - service URL parameters 565
- PeopleSoft tags 29, 203, 217
- PeopleTools
 - accessing 21
 - base portal *See Also* PeopleSoft portal
 - naming objects 103, 113
 - portal technology *See Also* portal technology
 - web libraries 747
- performance issues
 - applying load balancing 296
 - applying timeouts to browser windows 261
 - caching, browser-based 229
 - caching, configuring portal 6
 - caching, page 232
 - caching, understanding portal 221
 - caching, web server-based navigation 233
 - caching metadata 227
 - creating content reference links 162
 - Performance Monitor
 - See Also* Performance Monitor
 - portal homepage pagelets 301
 - proxying 37
 - server memory consumption, managing 228
 - servers, configuring application/database 20
 - SSL, improving same-server performance under 288
 - SSL accelerators, using 289
 - tracing 266
- Performance Monitor
 - enabling agents 245
 - enabling agent tracing 265
 - logging debug information 265
 - setting the monitor buffer maximum 245
- Performance Monitor web library 748
- permission lists
 - menu items, accessing 182
 - menu items, viewing permission lists

- associated with 186
 - menu security, setting 184
 - reviewing content reference security 159
 - setting folder security 115
 - setting homepage and dashboard security 307
- persistent search results
 - See Also* recent search results
 - enabling caching 264
- personalization
 - caching personalized pagelets 226
 - selecting personalization pages for pagelets 509
- pivot grids
 - viewing pagelets 384
- Placement in Folder 75
- popup menus
 - related content 534
- portal
 - search autocomplete 174
 - search options 174
- PORTAL_CSS program 169
- PORTAL_DEFAULT template 179
- PORTAL_HP_2COL_LAYOUT object 190
- PORTAL_HP_3COL_LAYOUT object 190
- PORTAL_HP_ACTIONBARIMAGE object 190
- PORTAL_HP_ACTIONBAR object 190
- PORTAL_HP_COMPONENT object 190
- PORTAL_HP_TABS object 193, 194
- PORTAL_HP_USER_TEMPLATE object 190, 192, 193
- PORTAL_INDEX program 187
- PORTAL_MENU process
 - See Also* Menu Import process
- PORTAL_UNI_HEADER_NEW object 190
- portal banner pagelet
 - implementing 151
- portal definitions
 - adding/editing 106
 - adding portals to other databases 104
 - administering 104
 - changing default portals 107
- Portal Definitions - Add Portal page 106
- portal favorites and popups web library 748
- portal objects
 - caching 228, 262
 - managing 176
 - Portal Objects folder
 - See Also* Portal Objects folder
 - synchronizing security 168
- Portal Objects folder
 - changing/viewing settings 177
 - understanding 27
- Portal page 279
- PortalRegisteredURL response header 218
- portal registry
 - administering 97
 - building search indexes 186
 - database support 28
 - importing menu groups
 - See Also* Menu Import process
 - maintaining 28
 - navigating 26
 - understanding 27
- portals
 - adding to databases 104
 - administering content references 143
 - administering folders 107
 - administering SmartNavigation folders 115
 - allowing unregistered content 249
 - authentication domains
 - See Also* authentication domains
 - caching *See Also* caching
 - changing default 107
 - configuring general properties 242
 - configuring security 5, 247
 - definitions *See Also* portal definitions
 - homepages *See Also* homepages
 - improving same-server performance under SSL 288
 - JSR 168 *See Also* JSR 168
 - load balancing 296
 - managing settings 172
 - multiple applications on 237
 - nodes *See Also* nodes
 - objects *See Also* portal objects
 - overriding character set encoding for pages 283
 - overriding the protocol/server name for URL references 255
 - PeopleSoft portal *See Also* PeopleSoft portal
 - PeopleSoft Pure Internet Architecture
 - See Also* PeopleSoft Pure Internet Architecture
 - portal-aware applications 150
 - Portal Objects folder
 - See Also* Portal Objects folder
 - portal servlets *See Also* portal servlets
 - portal technology *See Also* portal technology
 - registering components 170
 - registry *See Also* portal registry
 - selecting components to include in navigation 287
 - setting up 5
 - signing on to non-default 282
 - templates *See Also* templates
 - using custom headers 199
 - using HTTPS 249
 - viewing portal attributes for web servers 278
 - web profiles *See Also* web profiles
- Portal Security Synchronization page 168
- 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 286
 - setting up 30
 - understanding 17, 29
- portal technologies
 - understanding 3
- portal technology
 - architecture 23
 - implementing 3
- portal template processing web library 749
- portal utilities
 - creating navigation collections 61
 - defining at the registry level 59
 - defining at the system level 56
 - publishing navigation collections 81
 - running 88
- portal utilities options
 - defining at the registry level 59
 - defining at the system level 56
- portal web library 748

- portlet container
 - configuring for JSR 168 692
- portlet entity registry 693
- portlets, JSR 168 *See* JSR 168 portlets
- ports, overriding 255
- process flow
 - related content 543
- Process Scheduler 15
- producer
 - importing information 672
 - registering 674
- Producer Offered Portlets page 676
- producing remote portlets
 - language support for 745
- PROD web profile 238
- protocol
 - understanding WSRP 643
 - using HTTPS 657
- protocols, specifying default 255
- proxying
 - caching proxied JavaScript 229
 - converting anchor tags 39
 - converting form tags 39
 - converting JavaScripts 40
 - converting URLs from relative to absolute 39
 - cookies 258
 - handling page-based templates 38
 - proxy servers *See Also* proxy servers
 - reverse proxy servers
 - See Also* reverse proxy servers
 - specifying hosts not to proxy 293
 - understanding 30
 - understanding relative URLs 37
- proxy servers
 - issuing requests 292
 - reverse proxy servers
 - See Also* reverse proxy servers
 - specifying hosts not to proxy 293
- PS_TOKEN
 - securing 250
 - selecting authorization token enable pages
 - 271
 - understanding authentication domains 236
 - understanding single signon 281
- PSADMIN
 - managing application/Process Scheduler
 - servers 16
 - understanding 9
- PSAPPSRV server process 13
- PSCipher 241, 660
- 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 13
- PSSAMSRV server process 14
- PSSTYLEDEF style sheet 200
- PSSUBDSP server process 14
- PSSUBHND server process 14
- PT_HNAV object 193
- PT_IFRAME_HDR object 193
- PTPP_CPPROJ 89
- PTPP_FD_DEL 89
- PTPP_SCSYNC 89
- public access
 - allowing 252

- public users
 - authenticating 253
 - understanding 252
- Publish Collection page 81

Q

- queries
 - adding query strings for template pagelets
 - 216
 - publishing as pagelets 623
 - reviewing content reference security 158
 - URL format 35
- queries web library 749
- query URL format 35
- queues
 - request 12
 - understanding 10

R

- recently used
 - enabling 175
- recent search results
 - caching 264
 - system-level disabling and enabling 264
- red glyph 534
- redirecting pages 283
- red triangle field icon 534
- refreshing
 - automatic cache refresh 225
 - disabling for pagelets 508
 - disabling manual pagelet 227
 - forcing for pagelets 508
 - metadata caches 228
 - updating pages 201
- refresh tags
 - order of precedence for multiple pagelets 513
- registration
 - registering components in multiple portals
 - 170
 - registering components multiple times 170
 - registering content references 31
 - registering related link URLs 156
 - Registration Wizard
 - See Also* Registration Wizard
- Registration page 674
- registration properties 675
- Registration Wizard
 - adding internal PeopleSoft content references
 - 147
 - registering components 170, 285
 - registering menus 285
- Registry Name 60
- Registry Object Prefix 58
- registry options, defining for navigation
 - collections 59
- Registry Options page 53
- related actions
 - assigning component-level 582
 - assigning page-level 586
 - managing 574
 - on rowset-based SmartNavigation pages 572
 - on tree-based SmartNavigation pages 571

- related actions menus
 - configuring folders 592, 595
 - related content 533
 - application class URL 557
 - assigning component-level 580
 - assigning page-level 584
 - assigning services to application pages 578
 - assigning to content references 580, 584
 - cloning 568
 - configuring OBIEE 737
 - configuring service parameters 588
 - configuring UPK modules 725
 - copying service definitions 568
 - defining services 546
 - developing 533
 - development process flow 543
 - embeddable pagelet URL 562
 - filtering links 576
 - frames
 - configuring 596
 - iScripts 565
 - managing 577
 - managing configuration 589
 - managing layout 589
 - menus
 - configuring related actions 591
 - configuring Related Content 593
 - configuring page field 594
 - non-PeopleSoft URL 560
 - OBIEE URL 563
 - on SmartNavigation pages 571
 - PeopleSoft component URL 561
 - PeopleSoft Query URL 564
 - PeopleSoft script URL 565
 - popup menus 534
 - service parameters
 - as content reference attributes 152
 - service type for UPK 560
 - steps to test service parameters 566
 - URL testing 566
 - User Productivity Kit (UPK) 548
 - writing help information 566
 - related content frames
 - configuring 596
 - understanding 590
 - Related Content Framework
 - supported service types 547
 - Related Content menu
 - configuring 593
 - related content menus
 - field level 534
 - understanding 589
 - related content page field menus
 - configuring 594
 - related content service
 - filters 576
 - Related Content Service Definition page
 - help information 566
 - related content service parameters
 - configuring 588
 - mapping to fields 588
 - related content services 533
 - delivered in PeopleTools 543, 544
 - Oracle WebCenter 544
 - Oracle WebCenter Blogs 601
 - Oracle WebCenter Blogs service 545
 - Oracle WebCenter Discussion 601
 - Oracle WebCenter Discussion service 545
 - Oracle WebCenter Documents 601
 - Oracle WebCenter Documents service 545
 - Oracle WebCenter Links 602
 - Oracle WebCenter Links service 546
 - Oracle WebCenter Wiki 602
 - Oracle WebCenter Wiki service 546
 - PeopleSoft Application Portal 543
 - Related Discussion service 544
 - Related Links service 544
 - Related Tags service 544
 - understanding security 550
 - related content service types
 - understanding 547
 - related content web library 749
 - Related Discussion service 544
 - related links
 - creating 153
 - creating groups 155
 - defining the RELLINK attribute 154
 - testing 156
 - using the RELLINK attribute 152
 - Related Links service 544
 - Related Tags service 544
 - remote portlets 677, 681
 - remote portlets, setting up 664
 - reporting web library 749
 - reports
 - report repository servlet 17
 - selecting properties 246
 - request queues 12
 - reverse proxy servers
 - overriding the port/protocol/server name for
 - URL references 255
 - proxying and relative URLs 37
 - proxy servers *See Also* proxy servers
 - retrieving portal content 256
 - selecting 291
 - setting up the hosts file 294
 - understanding 290
 - using firewalls 293
 - using SSL accelerators 295
 - rich text editor image upload web library 749
 - roles
 - accessing menu items 182
 - reviewing content reference security 159
 - server-based caching 222
 - setting folder security 115
 - setting homepage and dashboard security 307
 - setting menu security 184
 - root folder 27
 - RPS *See* reverse proxy servers
 - rules, configuring cookie 257
- ## S
- SAML 645
 - scripts
 - encrypting passwords (PSCipher) 241
 - iScripts *See Also* iScripts
 - JavaScripts *See Also* JavaScripts
 - security settings for content references
 - 146, 158
 - time-out warning 749
 - timeout warning 251
 - URL format 34, 36
 - script URL format 34, 36

- SDK web library 749
- search
 - building registry search indexes 186
 - using 26
- Secure Sockets Layer (SSL) *See* SSL
- securing data 283
- security
 - accessing portal administration pages 99
 - accessing web libraries 747
 - configuring application 5
 - configuring for PeopleSoft portal 696
 - configuring for portals 5
 - configuring portal 247
 - configuring web servers 21
 - modifying for web libraries 485
 - related content 550
 - reviewing for content references 158
 - setting for content reference links 164
 - setting for content references 157
 - setting for folders 114
 - SSL *See Also* SSL
 - synchronizing for portal objects 168
 - viewing menu details 184
 - viewing permission lists, roles and users with
 - menu item access 182
 - web profiles 239
- security requirements for WSRP
 - determining 649
- Security standard navigation page 44
- Select a Content Reference page 579
- Select Source Folder page 71
- Select Source Link page 77
- sensitive data pages 283
- 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) 237
 - improving same-server performance under
 - SSL 288
 - monitoring 14
 - overriding the server name for URL
 - references 256
 - passing cookies 257
 - Process Scheduler 16
 - proxy servers *See Also* proxy servers
 - reverse proxy *See Also* reverse proxy servers
 - server-based caching 222
 - web *See Also* web servers
- service definitions, steps to copy 568
- services
 - Homepage Generation Service 198
 - modifying application page assignment 589
 - server processes *See Also* server processes
 - single signon authentication 281
 - understanding 14
 - web 17
- service types
 - supported by Related Content Framework
 - 547
- servlets
 - integration gateway 17
- Java *See Also* Java servlets
- portal *See Also* portal servlets
- report repository 17
- servlet engine 17
- setting timeouts 253
- understanding Jolt and Tuxedo 17
- session cookies, allowing *See* cookies
- session variables 271, 272
- Show Breadcrumbs 55
- Show Images
 - defining at registry level 61
 - defining at system level 59
- Show Main Menu Breadcrumb 55
- signing out 27
- signon
 - checking for duplicate cookies 276
 - displaying the trace setting link 265
 - selecting error pages for 272
 - selecting pages for 272
 - signing on to non-default portals 282
 - single *See Also* single signon
- single signon
 - accessing frame-based content 282
 - authentication domains
 - See Also* authentication domains
 - securing PS_TOKEN 250
 - understanding 281
- 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
- SmartNavigation
 - application package parameters 137
 - creating folders 138
 - custom navigation object 102
 - default navigation object 102
 - sample PeopleCode
 - rowset data source 119
 - tree data source 124
 - understanding related actions 570
- SmartNavigation folders
 - understanding 116
- SOAP 646
- soap, web library 749
- Specify Data Source Parameters page 378
- SQL
 - application server domains
 - See Also* application server domains
 - server processes 13
- SSL
 - accelerators *See Also* SSL accelerators
 - improving same-server performance 288
 - overriding the port/protocol for URL
 - references 255
 - selecting the SSL required pages 272
 - setting web profile options 249
 - specifying HTTPS for portals 249
- SSL accelerators
 - applying load balancing 296
 - using 289
 - using reverse proxy servers 295
- standard folder navigation
 - understanding 100

- standard folders
 - navigation pages 100
- standard navigation pages
 - describing folder and link sequence on 46
 - enabling 45
 - understanding 44
- states
 - browser 232, 259
 - menu 231
 - window 692
- static templates
 - frame-based example 205
 - page-based example 205
 - understanding 204
- Status 675
- Structure and Content for Portal Objects folder 177
- Style Sheet Name 55
- style sheets
 - assembling pages 31
 - changing link colors 200
 - configuring cache directories 263
 - inheriting classes 217
 - inserting style sheets into page HTML 266
- 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

- tabbed homepages *See* homepages
- Tab Content page 308
- Tab Layout page 309
- tables, HTML 204
- tabs
 - accessing online help 306
 - adding *See* homepages
 - adding images 197
 - changing order 311
 - creating 196
 - hiding action bars *See Also* action bars
 - HTML objects for 193
 - labelling 305
 - naming 305
 - overriding images/objects 196
 - setting accessibility by date 304
- tags
 - anchor 39
 - form 39
 - PeopleSoft 29, 203, 217
 - template pagelet HTML 217
- target area
 - forcing new window 151
- Target tag 31
- template pagelets
 - based on pages 216
 - understanding 25
 - understanding HTML tags for 217
- templates
 - adding 179
 - applying to unregistered URLs 219
 - configuring node-specific information 176

- conflicting style classes 217
- deleting 179
- developing 216
- dynamic *See Also* dynamic templates
- editing 179
- frame-based *See Also* frame-based templates
- iframe-based
 - See Also* iframe-based templates
- managing 177
- page-based *See Also* page-based templates
- pagelets *See Also* template pagelets
- PORTAL_DEFAULT template 179
- selecting for portal servlets 286
- setting pagelet names in 219
- specifying pagelet caching 224
- static *See Also* static templates
- storing (HTML catalog) 177
- types of 203
- understanding 25, 203
- using non-PeopleSoft content providers 218
- using predefined template HTML objects 189
- template types
 - applying 203
- testing
 - content references 159
 - generating HTML for 266
 - related content service 566
 - related content service parameters 566
 - related links 156
 - using the TEST web profile 239
- Test Related Content Service page 566
- TEST web profile 239
- timeouts
 - applying to browser windows 261
 - configuring expiration pages 269
 - configuring for pagelets 510
 - setting for authenticated users 250
 - setting for Jolt 253
 - setting for pagelet loading 151
 - setting for public users 253
 - setting for servlets 253
- timeouts web library 749
- tracing
 - configuring 264, 700
 - selecting pages for setting trace parameters 271
- transactions
 - managing *See Also* Tuxedo
 - understanding server processes 13
 - web servers *See Also* web servers
- Tuxedo
 - transmitting requests and data 17
 - understanding 14
 - workstation listeners/handlers 12

U

- uniform resource locators (URLs) *See* URLs
- universal navigation header
 - constructing 189, 190, 193
 - understanding the buttons 26
- UNIX
 - monitoring servers 14
 - using PSCipher 241
- UPK modules

- configuring as related content 725
- URL
 - selecting a data source 371
 - specifying data source parameters 391
- URL Information 679
- URLs
 - applying templates for partial path nodes 219
 - changing default portals 107
 - constructing references for frame-based templates 218
 - content reference links 162
 - converting from relative to absolute 38
 - defining content references 149
 - generating relative 255
 - invoking iScripts 747
 - nodes *See Also* nodes
 - overriding the port/protocol/server name for URL references 255
 - PeopleSoft portal URL formats 32
 - PeopleSoft Pure Internet Architecture content types 34
 - proxying *See Also* proxying
 - retrieving non-PeopleSoft content 218
 - understanding pagelet 36
 - understanding relative 37
 - understanding system 36
 - understanding the portal registry 27
 - unwrapped content formats 35
 - using authentication domains 237
- URL type
 - creating OBIEE pagelet 563
 - creating PeopleSoft Query 564
 - creating PeopleSoft script 565
 - embeddable pagelet 562
- URL types
 - creating application class 557
 - creating non-PeopleSoft 560
 - creating PeopleSoft component 561
- User ID
 - default for public users 253
- User Productivity Kit (UPK)
 - related content service type 548
- User Profiles standard navigation page 44
- users
 - accessing menu items 182
 - reviewing content reference security 158
 - selecting the user profile page 272
 - server-based caching 222, 225, 226
 - setting access options for web profiles 252
- utilities
 - PSADMIN 9
 - PSCipher 241

V

- variables
 - bind *See Also* bind variables
 - global 236
 - numeric message 271
 - session 272
 - session message 271
 - sharing between components 236
 - system 257
- virtual addressing
 - configuring 254, 697
 - running portals under SSL 288

- setting default 255
- using firewalls with reverse proxy servers 294
- using reverse proxy servers with SSL accelerators 295
- using SSL accelerators 290

W

- WebCenter group spaces
 - adding links to 611
- WebCenter integration
 - mapping privilege roles 607
 - options 606
 - setting the default template 607
 - synchronizing PeopleSoft security 608
- WebCenter links
 - adding 611, 612
 - adding external URLs 612
 - managing 609
- WebCenter services
 - managing links 609
- web libraries 747
- WebLogic
 - applying load balancing 296
 - preventing cookie overwrites 257
 - running portals under SSL 288
 - specifying hosts not to proxy 293
 - using proxy servers to issue requests 292
 - using SSL accelerators 290
- Web Profile Configuration component
 - Caching page 258
 - Cookie Rules page 257
 - Custom Properties page 274
 - Debugging page 264
 - General page 242
 - Look and Feel page 267
 - Security page 247
 - Virtual Addressing page 254
- Web Profile History page 278
- web profiles
 - changing 240
 - configuring 5
 - configuring browser caching 259
 - configuring cache directories 263
 - configuring caching 698
 - configuring cookie rules 257, 698
 - configuring custom properties 274
 - configuring error pages 270
 - configuring expiration pages 269
 - configuring general portal properties 242
 - configuring general properties 695
 - configuring look and feel 267, 701
 - configuring password pages 273
 - configuring portal security 247
 - configuring recent search results caching 264
 - configuring security 696
 - configuring security/tracing pages 271
 - configuring signon/signoff pages 272
 - configuring tracing/debugging 700
 - configuring tracing/debugging options 264
 - configuring virtual addressing 254, 697
 - configuring web server caching 262
 - encrypting user IDs/passwords 241
 - improving same-server performance under SSL 288

- mapping language codes 274
 - selecting reverse proxy servers 256, 291
 - setting default addressing 255
 - setting SSL options 249
 - setting timeout options 250
 - setting user access options 252
 - setting web server Jolt options 253
 - specifying initial 239
 - understanding 238
 - unlocking accounts 239
 - using firewalls with reverse proxy servers 293
 - using reverse proxy servers with SSL accelerators 295
 - using SSL accelerators 289
 - viewing histories 278
 - web servers
 - applying load balancing 296
 - browsers *See Also* browsers
 - caching, configuring 262
 - caching pagelet content 224
 - caching target content 223
 - configuring 20
 - Jolt *See Also* Jolt
 - managing memory consumption 228
 - maximum name length 33
 - PeopleSoft servlets 17
 - portal servlets *See Also* portal servlets
 - setting up the hosts file 294
 - software elements 16
 - Tuxedo *See Also* Tuxedo
 - understanding 16
 - viewing portal attributes 278
 - WebLogic *See Also* WebLogic
 - WebSphere *See Also* WebSphere
 - Web Service Endpoint URL 673
 - web services 17
 - web services, understanding 643
 - Web Services Description Language (WSDL)
 - web library 747
 - WebSphere
 - preventing cookie overwrites 257
 - specifying hosts not to proxy 293
 - using proxy servers to issue requests 292
 - Windows
 - accessing online help 244
 - monitoring servers 14
 - using the PeopleTools development environment 8, 21
 - window states, JSR 168 692
 - WorkCenter pagelets
 - component-based 624
 - navigation collection-based 620
 - query-based 622
 - template pagelets 614
 - WorkCenter pages
 - configuring layout 631
 - layout templates 616
 - page layout templates 631
 - pagelet area 614
 - prerequisites 630
 - technology process overview 629
 - understanding 613
 - worklists, accessing 26
 - worklist URL format 35
 - workstation handlers (WSH) 12
 - workstation listeners (WSL) 12
 - WS-Security
 - configuring 663
 - configuring for WSRP consumers 665
 - understanding 650
 - WS-Security SOAP Message Structure 650
 - WSDL 646
 - WSDL web library 747
 - WSH 12
 - WSL 12
 - WSRP 646
 - configuring WS-Security 663
 - configuring WS-Security for 649
 - defining interfaces and operations 644
 - defining terminologies 645
 - implementing protocol scenario 646
 - language support for 745
 - offering portlets 684
 - setting up security options 657
 - specifying pagelets for production 682
 - understanding 643
 - understanding the Consumption Wizard Data 671
 - using the consumption wizard 670
 - viewing content references 686
 - WSRP4J 646
 - WSRP Consumption Wizard
 - understanding the data 671
 - using 670
 - WSS 646
-
- # X
-
- XML 646
 - XML Link 254
 - xmllinks web library 749