PeopleSoft®

Enterprise PeopleTools 8.46 PeopleBook: System and Server Administration

February 2005

Enterprise PeopleTools 8.46 PeopleBook: System and Server Administration
SKU PT846SVT-B 0205
Copyright © 1988-2005 PeopleSoft, Inc. All rights reserved.

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

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

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

PeopleSoft, PeopleTools, PS/nVision, PeopleCode, PeopleBooks, PeopleTalk, and Vantive are registered trademarks, and Pure Internet Architecture, Intelligent Context Manager, and The Real-Time Enterprise are trademarks of PeopleSoft, Inc. All other company and product names may be trademarks of their respective owners. The information contained herein is subject to change without notice.

Open Source Disclosure

PeopleSoft takes no responsibility for its use or distribution of any open source or shareware software or documentation and disclaims any and all liability or damages resulting from use of said software or documentation. The following open source software may be used in PeopleSoft products and the following disclaimers are provided.

Apache Software Foundation

This product includes software developed by the Apache Software Foundation (http://www.apache.org/). Copyright © 1999-2000 The Apache Software Foundation. All rights reserved.

THIS SOFTWARE IS PROVIDED “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

OpenSSL

Copyright © 1998-2003 The OpenSSL Project. All rights reserved.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/).

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

SSLeay

Copyright © 1995-1998 Eric Young. All rights reserved.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com). Copyright © 1995-1998 Eric Young. All rights reserved. THIS SOFTWARE IS PROVIDED BY ERIC YOUNG “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Loki Library

Copyright © 2001 by Andrei Alexandrescu. This code accompanies the book:

Alexandrescu, Andrei. “Modern C++ Design: Generic Programming and Design Patterns Applied.” Copyright © 2001 Addison-Wesley. Permission to use, copy, modify, distribute and sell this software for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

Helma Project

Copyright © 1999-2004 Helma Project. All rights reserved.

THIS SOFTWARE IS PROVIDED “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE HELMA PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Helma includes third party software released under different specific license terms. See the licenses directory in the Helma distribution for a list of these license.

Sarissa

Copyright © 2004 Manos Batsis.

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA.

Contents

General Preface

About This PeopleBook xxi

PeopleSoft Application Prerequisites xxi

PeopleSoft Application Fundamentals xxi

Documentation Updates and Printed Documentation xxii

Obtaining Documentation Updates xxii

Ordering Printed Documentation xxii

Additional Resources xxiii

Typographical Conventions and Visual Cues xxiv

Typographical Conventions xxiv

Visual Cues xxv

Country, Region, and Industry Identifiers xxvi

Currency Codes xxvi

Comments and Suggestions xxvi

Common Elements Used in PeopleBooks xxvii

Preface

System and Server Administration Preface xxix

System and Server Administration xxix

Chapter 1

Getting Started with System and Server Administration 1

System and Server Administration Overview 1

PSADMIN 1

Analytic Servers 2

Web Servers 2

Search Indexes 3

PeopleSoft Configuration Manager 4

PeopleTools Utilities 4

Tracing and Debugging 5

Jolt Internet Relay 5

Environment Replication 5

Timeout Settings 6

System and Server Administration Implementation.....6

Chapter 2
Using the PSADMIN Utility.....7
Understanding PSADMIN.....7
Starting PSADMIN.....8
Using PSADMIN.....9
Using Configuration Templates.....9
Using the Quick-Configure Menu.....10
Using the PSADMIN Command-Line Interface.....11
Understanding the PSADMIN Command-Line Interface.....11
Using the Miscellaneous Commands.....11
Using the Application Server Commands.....12
Using the Process Scheduler Commands.....16
Using PSADMIN Executables and Configuration Files.....19
Understanding PSADMIN Executables and Configuration Files.....20
Configuring a Domain.....21
Loading a Configuration.....23
Archiving Application Server Configuration Files.....24
Booting a Domain.....24
Stopping a Domain.....24
Monitoring a Domain.....24
Configuring the Application Server to Handle Cache Files and Replay Files.....25

Chapter 3
Using PSADMIN Menus.....27
Using the Application Server Administration Menu.....27
Accessing the Application Server Options.....27
Administering a Domain.....28
Booting a Domain.....28
Shutting Down a Domain.....28
Performing a Normal Shutdown.....28
Performing a Forced Shutdown.....29
Checking the Domain Status.....29
Purging the Domain Cache.....30
Configuring a Domain.....32
Editing Configuration and Log Files.....33
Creating a Domain.....35

Deleting a Domain....35
Using the Process Scheduler Menu....35
Understanding the Process Scheduler Menu....36
Starting a Process Scheduler Server....36
Stopping a Process Scheduler Server....36
Configuring a Process Scheduler Server....36
Creating a Process Scheduler Server Configuration....37
Deleting a Process Scheduler Server....37
Editing the Process Scheduler Configuration File....37
Using the Process Scheduler Options....38
Using Process Scheduler Command-Line Options....38
Setting Up the PeopleSoft Windows Service....38
Understanding Microsoft Windows Services....38
Configuring the PeopleSoft Service....39
Monitoring the Executables....40
Administering PeopleSoft Services....41
Editing the PSNTSRV.CFG File Manually....41

Chapter 4
Setting Application Server Domain Parameters....43
Startup Options....44
DBName....44
DBType....44
UserID....44
UserPswd....44
Connect ID....44
Connect Password....44
ServerName....45
Database Options....45
SybasePacketSize....45
UseLocalOracleDB....45
EnableDBMonitoring....45
OracleDisableFirstRowsHint....45
Security Options....46
Validate Signon With Database....46
Workstation Listener Options....46
Address....46
Port....46
Encryption....46

Min Handlers....47
Max Handlers....47
Max Clients per Handler....47
Client Cleanup Timeout....47
Init Timeout....47
Tuxedo Compression....47
BEA Jolt Listener Options....48
Address....48
Port....48
Encryption....48
Min Handlers....48
Max Handlers....48
Max Clients per Handler....48
Client Cleanup Timeout....49
Init Timeout....49
Client Connection Mode....49
Jolt Compression Threshold....49
Additional Prompt....50
BEA Jolt Relay Adapter Options....50
Listener Address....50
Listener Port....50
Domain Settings....50
Domain ID....50
Add to PATH....50
Spawn Threshold....51
Restartable....51
Allow Dynamic Changes....51
LogFence....52
AppLogFence....52
Trace-Log File Character Set....52
PeopleCode Debugger Options....52
Trace Options....53
TraceSQL....53
TraceSQLMask....53
TracePC....53
TracePCMask....54
TracePPR and TracePPRMask....54
TraceAE....55
TraceOpt and TraceOptMask....55
TracePPM....56

DumpMemoryImageAtCrash....56
DumpMemoryObjectsAtCrash....56
Log Error Report, Mail Error Report....56
Write Crash Dump to Separate File....56
Cache Settings....56
Cache Settings....57
EnableServerCaching....57
ServerCacheMode....57
CacheBaseDir....57
MaxCacheMemory....57
Remote Call Options....58
RCCBL Redirect....58
RCCBL PRDBIN....58
PSAPPSRV Options....59
Min Instances....59
Max Instances....59
Service Timeout....59
Recycle Count....59
Allowed Consec Service Failures....59
Max Fetch Size....59
Auto Select Prompt....60
Tuxedo Queue Size....60
PSANALYTICSRV Options....60
Min Instances....60
Max Instances....60
Analytic Instance Idle Timeout....60
PSSAMSRV Options....60
Min Instances....61
Max Instances....61
Service Timeout....61
Recycle Count....61
Allowed Consec Service Failures....61
Max Fetch Size....61
PSQCKSRV Options....61
Min Instances....62
Max Instances....62
Service Timeout....62
Recycle Count....62
Allowed Consec Service Failures....62
Max Fetch Size....62

PSQRYSRV Options....62
Min Instances....62
Max Instances....63
Service Timeout....63
Recycle Count....63
Allowed Consec Service Failures....63
Max Fetch Size....63
Use Dirty-Read....63
Messaging Server Processes....63
SMTP Settings....64
SMTPServer....64
SMTPPort....64
SMTPServer1....64
SMTPPort1....65
SMTPSender....65
SMTP BlackberryReplyTo....65
SMTPSourceMachine....65
SMTPCharacterSet....65
SMTPEncodingDLL....65
SMTPGuaranteed....65
SMTPTrace....65
SMTPSendTime....66
SMTPUserName....66
SMTPUserPassword....66
SMTPUserName1....66
SMTPUserPassword1....66
SMTPTimeToWaitForResult....66
SMTP Further Considerations....66
Interface Driver Options....67
SCP_LOCALE....67
PSTOOLS Options....67
EnablePPM Agent....67
Add to CLASSPATH....67
Java VM Options....67
Proxy Host....67
Proxy Port....68
Non Proxy Hosts....68
Character Set (UNIX or USS Only)....68
Suppress App Error Box (Microsoft Windows Only)....69
DbFlags....69

Suppress SQL Error....70
PeopleSoft Integration Broker Options....70
Min Message Size for Compression....70
IB Profile Information....70
Message View Size....70
Search Indexes....70
PSRENSRV Options....71
log-severity_level....71
io_buffer_size....71
default_http_port....71
default_auth_token....71
PSPPMSRV Options....71
Min Instances....71
Max Instances....71
Select Server Process Options....72
Do you want the Publish/Subscribe servers configured?....72
Move quick PSAPPSRV services into a second server (PSQCKSRV)?....72
Move long-running queries into a second server (PSQRYSRV)?....72
Do you want JOLT configured?....72
Do you want JRAD configured?....72
Do you want to enable PeopleCode Debugging?....72
Do you want the Optimization engines configured?....73
Do you want Event Notification configured?....73
Do you want MCF Servers configured?....73
Do you want Performance Collators configured?....73

Chapter 5
Managing Analytic Servers....75
Understanding the Analytic Server Framework....75
Analytic Server Framework Overview....75
Analytic Server Process Flow and Behavior....78
Understanding Batch Processing of Analytic Instances....80
Configuring and Starting Analytic Servers....80
Enabling PSANALYTICSRV....80
Specifying Analytic Server Instance Quantities....81
Starting PSANALYTICSRV....81
Administering Analytic Servers....82
Administering Analytic Server Domains....82
Administering Analytic Server Instances....83

Administering Analytic Tables.....85
Purging Delete Tables.....85
Synchronizing Table Versions.....86
Creating, Deleting, and Copying Analytic Instances.....87
Pages Used to Create, Delete, and Copy Analytic Instances.....88
Creating Analytic Instances.....88
Deleting Analytic Instances.....89
Copying Analytic Instances.....91
Loading and Unloading Analytic Instances.....92
Page Used to Load and Unload Analytic Instances.....93
Loading and Unloading Analytic Instances.....93

Chapter 6
Working with BEA WebLogic.....97
Understanding BEA WebLogic.....97
The PeopleSoft Domain.....97
WebLogic Session Cookie Name Format.....98
Accessing the BEA WebLogic Server Console.....99
Starting BEA WebLogic.....99
Starting BEA WebLogic on Microsoft Windows.....100
Starting BEA WebLogic on UNIX.....100
Stopping BEA WebLogic.....101
Using WebLogic Server 8.1 Console to Monitor PeopleSoft Sessions.....102
Setting Up an RPS.....103
Configuring Microsoft IIS as an RPS.....103
Configuring BEA WebLogic as an RPS.....105
Configuring Sun iPlanet as an RPS.....107
Using the iPlanet Plug-in.....110
Configuring Apache HTTP as an RPS.....111
Setting Up HTTP Session Timeout.....113
Enabling or Disabling HTTP Keep Alive.....113
Changing a WebLogic User's Password.....114
Implementing WebLogic SSL Keys and Certificates.....115
Understanding SSL Encryption with WebLogic 8.1.....116
Obtaining Encryption Keys.....116
Preparing Keys and Certificates for the Keystore.....118
Importing Keys and Certificates Into the Keystore.....120
Configuring WebLogic SSL Encryption Keys.....122
Restricting Access to a Servlet.....123

Adjusting the JVM Heap Size....125
Determining the Service Pack Level....126
Enabling or Disabling HTTP Access Log....127

Chapter 7
Working with IBM WebSphere....129
Installing and Uninstalling WebSphere Features....129
Installing WebSphere 5.1.1 Using Silent Install....129
Uninstalling WebSphere 5.1.1....130
Uninstalling WebSphere Embedded Messaging....131
Setting Up HTTP Server (RPS) with WebSphere 5.1.1....132
Understanding WebSphere Reverse Proxy Servers....133
Setting Up a WebSphere Machine that Requires RPS Support....133
Installing IBM HTTP Server and its RPS Plug-in Together....133
Installing Only the RPS Plug-in for IBM HTTP Server....134
Installing Only the RPS Plug-in for Sun ONE Web Server....135
Installing Only the RPS Plug-in for Microsoft IIS....135
Configuring the HTTP Server....136
Setting Up SSL on WebSphere 5.1.1....137
Generating a Certificate for the Web Container....137
Modifying the Web Container to Support SSL....139
Testing Your Setup....139
Setting Up SSL on IBM HTTP Server....139
Prerequisites for Setting Up SSL....140
Creating Key Files for SSL....140
Adding the Certificate Authority for PeopleTools Testing....141
Creating a Personal Certificate Request....141
Sending the Certificate Request to a Certification Authority for Enrollment....141
Adding the New Certificate from the Enrollment Page....142
Editing the httpd.conf File....142
Testing the Setup....143
Using the WebSphere 5.1.1 Administration Console....143
Starting and Stopping the Administration Console....143
Administering WebSphere 5.1.1....144
Running Application Server as a Non-root User....146
Setting Up IBM Http Server to Run as a Non-root....147
Disabling Directory Browsing in IBM HTTP Server....148
Using the Log Analyzer and Resource Analyzer....149
Using the Log Analyzer....149

Using the Resource Analyzer....149
Uninstalling PIA on WebSphere....150
Installing Additional PIA to Existing WebSphere 5.1.1 Instance....150
Deploying Multiple PIA Instances on WebSphere 5.1.1....151
Understanding Multiple PIA Instances and Virtual Hosting....151
Using Multiple WebSphere Base Instances....152
Using a Single WebSphere Base Instance....154
Using WebSphere ND....156
Reserving WebSphere 5.1.1 PIA Ports....158
Accessing the WebSphere InfoCenter....159

Chapter 8
Building and Maintaining Search Indexes....161
Understanding PeopleSoft Search Indexes....161
Overview of Search Indexes....161
Types of Indexes....162
Components of the Search Architecture....162
Index Building....164
Search Index Limitations....164
User Search Strategies....165
Working with Indexes....166
Understanding Common Controls....166
Understanding Supported MIME Types....166
Opening Existing Collections....167
Creating New Collections....168
Building Record-Based Indexes....168
Modifying Record-Based Index Properties....168
Adding Subrecords to Search Indexes....171
Building File System (Spider) Indexes....171
Setting File System Options....171
Defining What to Index....172
Building HTTP Spider Indexes....174
Defining HTTP Gateway Settings....174
Defining What to Index....175
Administering Search Indexes....175
Specifying the Index Location....176
Administering the Search Index....177
Editing Properties....177
Scheduling Administration....178

Sharing Indexes Between Application Servers and PeopleSoft Process Scheduler....178
Modifying the VdkVgwKey Key....179

Chapter 9
Using PeopleSoft Configuration Manager....181
Understanding PeopleSoft Configuration Manager....181
Common Elements in PeopleSoft Configuration Manager....182
Starting PeopleSoft Configuration Manager....182
Specifying Startup Settings....182
Specifying Display Settings....186
Specifying Crystal Report and Business Interlink Settings....189
Specifying Trace Settings....191
Specifying Workflow Settings....192
Specifying Remote Call/AE Settings....194
Configuring Developer Workstations....195
Importing and Exporting Environment Settings....197
Configuring User Profiles....198
Defining a Profile....198
Specifying Databases and Application Servers....199
Configuring Process Scheduler....202
Configuring nVision....204
Specifying Common Settings....206
Specifying Command Line Options....207
Setting Up the PeopleTools Development Environment....209
Understanding the PeopleTools Development Environment....209
Understanding the Client Setup Process....209
Verifying PS_HOME Access....210
Verifying Connectivity....210
Verify Supporting Applications....210
Using the Configuration Manager Pages....210
Running the Client Setup Process....211

Chapter 10
Using PeopleTools Utilities....213
Understanding the PeopleTools Utilities....213
Using the System Information Page....213
Understanding the System Information Page....214
Viewing the System Information Page....214

Using Administration Utilities....216
PeopleTools Options....217
Message Catalog....223
Spell Check System Dictionary....225
Translate Values....227
Load Application Server Cache....227
Tablespace Utilities....230
Tablespace Management....231
DDL Model Defaults....231
Strings Table....233
XML Link Function Registry....234
Merchant Integration Utilities....235
TableSet IDs....235
Record Group....235
TableSet Control....236
Convert Panels to Pages....237
Update Utilities....240
Remote Database Connection....241
URL Maintenance....242
Copy File Attachments....243
Query Monitor....244
Sync ID Utilities....244
Gather Utility....244
Using Audit Utilities....247
Using the Record Cross Reference Component....247
Performing a System Audit....249
Performing Database Level Auditing....250
Using Debug Utilities....250
Using the PeopleTools Test Utilities Page....250
Using the Trace PeopleCode Utility....251
Using the Trace SQL Utility....251
Using International Utilities....251
Setting International Preferences....251
Setting Process Field Size....252
Administering Time Zones....252
Managing Languages....252
Using Optimization Utilities....254
Using PeopleSoft Ping....254

Chapter 11
Configuring Trace and Debug Settings.....257
Setting Up the PeopleCode Debugger.....257
Debugging for a Two-Tier Connection.....257
Debugging for a Three-Tier Connection.....258
Using the PeopleCode Debugger.....260
Configuring PeopleCode Trace.....260
Configuring SQL Trace.....261

Chapter 12
Setting Up Jolt Internet Relay.....263
Understanding Jolt Internet Relay.....263
Jolt Internet Relay Architecture.....263
A Jolt Internet Relay Example.....264
Implementation Considerations.....265
Configuring JRLY.....266
Configuring JRAD.....268
Running Jolt Relay.....269
Using the JRLY Administration Program.....269
Running Jolt Relay on Windows.....269
Running Jolt Relay on UNIX.....270

Chapter 13
Replicating an Installed Environment.....271
Understanding Environment Replication.....271
Problems Associated with Environment Replication.....272
Replicating a Web Server Environment.....272
Replicating an Application Server Environment.....273
Steps to Replicate PeopleSoft Application Server Using Import Option.....274
Replicating the PeopleSoft Process Scheduler Environment.....274
Reconfiguring Replicated Environment Management Components.....275
Reconfiguring an Environment Management Agent.....275
Reconfiguring the Environment Management Hub.....276
Reconfiguring the Environment Management Viewer.....276

Appendix A
BEA WebLogic 8.1 Managed Server Architecture.....279
Web Applications in PIA.....279
WebLogic Domain Types.....280
Understanding WebLogic Domain Types.....280
Single-Server Domain.....280
Multi-Server Domain.....281
Distributed Managed Server.....284
Common Default Settings.....285
Single-Server and Multi-Server/Distributed Server Analogy.....288
Domain Topology.....289
WebLogic Domain Directory Structure and Files.....292
WebLogic Domain Directory Structure.....292
WebLogic Domain File Listing by Type.....293
J2EE Application Files.....296
PIA Install and Reinstall Options.....298
Administering a WebLogic Server Life Cycle.....299
Understanding the WebLogic Server Life Cycle.....299
Starting and Stopping Single-Server Processes.....299
Starting and Stopping Multi-Server Processes.....300
Starting and Stopping a Distributed Managed Server.....303
Tuning Performance and Monitoring Resources.....303
Managing JVM Heap Size and Execute Thread Usage.....303
Monitoring HTTP Session Count for PeopleSoft Portal.....309
Changing Configuration Settings.....310
Understanding the WebLogic Server Configuration Files.....310
Changing the WebLogicAdmin Server's Listen Ports.....311
Changing Application and Server Deployment Targets.....311
Applying an Example Single-Server Configuration.....312
Basic Machine Configuration.....312
PIA Setup Prompts.....312
Configuration File Settings.....313
Unprompted Configuration File Settings.....314
Starting the Weblogic Server.....314
Applying an Example Multi-Server Configuration.....315
Basic Machine Configurations.....315
PIA Setup Prompts for Multi-Server Configuration.....315
Configuration File Settings for Multi-Server Configuration.....316
Unprompted Multi-Server Configuration File Settings.....317
PIA Setup Prompts for Distributed Managed Server Configuration.....318

Configuration File Settings for Distributed Server Configuration.....319
Unprompted Distributed Server Configuration File Settings.....319
Starting the WebLogic Servers.....320

Appendix B
PeopleSoft Timeout Settings.....323
Web Server Timeouts.....323
Session-Timeout.....325
Application Server Timeouts.....325
Process Scheduler Timeouts.....328
PIA Timeouts.....329

Glossary of PeopleSoft Terms.....331

Index 353

About This PeopleBook

PeopleBooks provide you with the information that you need to implement and use PeopleSoft applications.

This preface discusses:

- PeopleSoft application prerequisites.
- PeopleSoft application fundamentals.
- Documentation updates and printed documentation.
- Additional resources.
- Typographical conventions and visual cues.
- Comments and suggestions.
- Common elements in PeopleBooks.

Note. PeopleBooks document only page elements, such as fields and check boxes, that require additional explanation. If a page element is not documented with the process or task in which it is used, then either it requires no additional explanation or it is documented with common elements for the section, chapter, PeopleBook, or product line. Elements that are common to all PeopleSoft applications are defined in this preface.

PeopleSoft Application Prerequisites

To benefit fully from the information that is covered in these books, you should have a basic understanding of how to use PeopleSoft applications.

You might also want to complete at least one PeopleSoft introductory training course, if applicable.

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

These books do not review navigation and other basics. They present the information that you need to use the system and implement your PeopleSoft applications most effectively.

PeopleSoft Application Fundamentals

Each application PeopleBook provides implementation and processing information for your PeopleSoft applications.

Note. Application fundamentals PeopleBooks are not applicable to the PeopleTools product.

For some applications, additional, essential information describing the setup and design of your system appears in a companion volume of documentation called the application fundamentals PeopleBook. Most PeopleSoft product lines have a version of the application fundamentals PeopleBook. The preface of each PeopleBook identifies the application fundamentals PeopleBooks that are associated with that PeopleBook.

The application fundamentals PeopleBook consists of important topics that apply to many or all PeopleSoft applications across one or more product lines. Whether you are implementing a single application, some combination of applications within the product line, or the entire product line, you should be familiar with the contents of the appropriate application fundamentals PeopleBooks. They provide the starting points for fundamental implementation tasks.

Documentation Updates and Printed Documentation

This section discusses how to:

- Obtain documentation updates.
- Order printed documentation.

Obtaining Documentation Updates

You can find updates and additional documentation for this release, as well as previous releases, on the PeopleSoft Customer Connection website. Through the Documentation section of PeopleSoft Customer Connection, you can download files to add to your PeopleBook Library. You'll find a variety of useful and timely materials, including updates to the full PeopleSoft documentation that is delivered on your PeopleBooks CD-ROM.

Important! Before you upgrade, you must check PeopleSoft Customer Connection for updates to the upgrade instructions. PeopleSoft continually posts updates as the upgrade process is refined.

See Also

PeopleSoft Customer Connection, https://www.peoplesoft.com/corp/en/login.jsp

Ordering Printed Documentation

You can order printed, bound volumes of the complete PeopleSoft documentation that is delivered on your PeopleBooks CD-ROM. PeopleSoft makes printed documentation available for each major release shortly after the software is shipped. Customers and partners can order printed PeopleSoft documentation by using any of these methods:

- Web
- Telephone
- Email

Web

From the Documentation section of the PeopleSoft Customer Connection website, access the PeopleBooks Press website under the Ordering PeopleBooks topic. The PeopleBooks Press website is a joint venture between PeopleSoft and MMA Partners, the book print vendor. Use a credit card, money order, cashier's check, or purchase order to place your order.

Telephone

Contact MMA Partners at 877 588 2525.

Email

Send email to MMA Partners at peoplebookspress@mmapartner.com.

See Also

PeopleSoft Customer Connection, https://www.peoplesoft.com/corp/en/login.jsp

Additional Resources

The following resources are located on the PeopleSoft Customer Connection website:

Resource	Navigation
Application maintenance information	Updates + Fixes
Business process diagrams	Support, Documentation, Business Process Maps
Interactive Services Repository	Interactive Services Repository
Hardware and software requirements	Implement, Optimize + Upgrade, Implementation Guide, Implementation Documentation & Software, Hardware and Software Requirements
Installation guides	Implement, Optimize + Upgrade, Implementation Guide, Implementation Documentation & Software, Installation Guides and Notes
Integration information	Implement, Optimize + Upgrade, Implementation Guide, Implementation Documentation and Software, Pre-built Integrations for PeopleSoft Enterprise and PeopleSoft EnterpriseOne Applications
Minimum technical requirements (MTRs) (EnterpriseOne only)	Implement, Optimize + Upgrade, Implementation Guide, Supported Platforms
PeopleBook documentation updates	Support, Documentation, Documentation Updates
PeopleSoft support policy	Support, Support Policy
Prerelease notes	Support, Documentation, Documentation Updates, Category, Prerelease Notes
Product release roadmap	Support, Roadmaps + Schedules
Release notes	Support, Documentation, Documentation Updates, Category, Release Notes

Resource	Navigation
Release value proposition	Support, Documentation, Documentation Updates, Category, Release Value Proposition
Statement of direction	Support, Documentation, Documentation Updates, Category, Statement of Direction
Troubleshooting information	Support, Troubleshooting
Upgrade documentation	Support, Documentation, Upgrade Documentation and Scripts

Typographical Conventions and Visual Cues

This section discusses:

- Typographical conventions.
- Visual cues.
- Country, region, and industry identifiers.
- Currency codes.

Typographical Conventions

This table contains the typographical conventions that are used in PeopleBooks:

Typographical Convention or Visual Cue	Description
Bold	Indicates PeopleCode function names, business function names, event names, system function names, method names, language constructs, and PeopleCode reserved words that must be included literally in the function call.
Italics	Indicates field values, emphasis, and PeopleSoft or other book-length publication titles. In PeopleCode syntax, italic items are placeholders for arguments that your program must supply. We also use italics when we refer to words as words or letters as letters, as in the following: Enter the letter *O*.
KEY+KEY	Indicates a key combination action. For example, a plus sign (+) between keys means that you must hold down the first key while you press the second key. For ALT+W, hold down the ALT key while you press the W key.
`Monospace font`	Indicates a PeopleCode program or other code example.

Typographical Convention or Visual Cue	Description
“ ” (quotation marks)	Indicate chapter titles in cross-references and words that are used differently from their intended meanings.
. . . (ellipses)	Indicate that the preceding item or series can be repeated any number of times in PeopleCode syntax.
{ } (curly braces)	Indicate a choice between two options in PeopleCode syntax. Options are separated by a pipe (|).
[ ] (square brackets)	Indicate optional items in PeopleCode syntax.
& (ampersand)	When placed before a parameter in PeopleCode syntax, an ampersand indicates that the parameter is an already instantiated object. Ampersands also precede all PeopleCode variables.

Visual Cues

PeopleBooks contain the following visual cues.

Notes

Notes indicate information that you should pay particular attention to as you work with the PeopleSoft system.

Note. Example of a note.

If the note is preceded by *Important!,* the note is crucial and includes information that concerns what you must do for the system to function properly.

Important! Example of an important note.

Warnings

Warnings indicate crucial configuration considerations. Pay close attention to warning messages.

Warning! Example of a warning.

Cross-References

PeopleBooks provide cross-references either under the heading “See Also” or on a separate line preceded by the word *See.* Cross-references lead to other documentation that is pertinent to the immediately preceding documentation.

Country, Region, and Industry Identifiers

Information that applies only to a specific country, region, or industry is preceded by a standard identifier in parentheses. This identifier typically appears at the beginning of a section heading, but it may also appear at the beginning of a note or other text.

Example of a country-specific heading: "(FRA) Hiring an Employee"

Example of a region-specific heading: "(Latin America) Setting Up Depreciation"

Country Identifiers

Countries are identified with the International Organization for Standardization (ISO) country code.

Region Identifiers

Regions are identified by the region name. The following region identifiers may appear in PeopleBooks:

- Asia Pacific
- Europe
- Latin America
- North America

Industry Identifiers

Industries are identified by the industry name or by an abbreviation for that industry. The following industry identifiers may appear in PeopleBooks:

- USF (U.S. Federal)
- E&G (Education and Government)

Currency Codes

Monetary amounts are identified by the ISO currency code.

Comments and Suggestions

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

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

Or send email comments to doc@peoplesoft.com.

While we cannot guarantee to answer every email message, we will pay careful attention to your comments and suggestions.

Common Elements Used in PeopleBooks

As of Date	The last date for which a report or process includes data.
Business Unit	An ID that represents a high-level organization of business information. You can use a business unit to define regional or departmental units within a larger organization.
Description	Enter up to 30 characters of text.
Effective Date	The date on which a table row becomes effective; the date that an action begins. For example, to close out a ledger on June 30, the effective date for the ledger closing would be July 1. This date also determines when you can view and change the information. Pages or panels and batch processes that use the information use the current row.
Once, Always, and **Don't Run**	Select Once to run the request the next time the batch process runs. After the batch process runs, the process frequency is automatically set to Don't Run. Select Always to run the request every time the batch process runs. Select Don't Run to ignore the request when the batch process runs.
Process Monitor	Click to access the Process List page, where you can view the status of submitted process requests.
Report Manager	Click to access the Report List page, where you can view report content, check the status of a report, and see content detail messages (which show you a description of the report and the distribution list).
Request ID	An ID that represents a set of selection criteria for a report or process.
Run	Click to access the Process Scheduler request page, where you can specify the location where a process or job runs and the process output format.
SetID	An ID that represents a set of control table information, or TableSets. TableSets enable you to share control table information and processing options among business units. The goal is to minimize redundant data and system maintenance tasks. When you assign a setID to a record group in a business unit, you indicate that all of the tables in the record group are shared between that business unit and any other business unit that also assigns that setID to that record group. For example, you can define a group of common job codes that are shared between several business units. Each business unit that shares the job codes is assigned the same setID for that record group.
Short Description	Enter up to 15 characters of text.
User ID	An ID that represents the person who generates a transaction.

See Also

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Process Scheduler

Enterprise PeopleTools 8.46 PeopleBook: Using PeopleSoft Applications

System and Server Administration Preface

This preface provides an overview of the contents discussed in the System and Server Administration PeopleBook.

System and Server Administration

This book includes several chapters relating to administration tools for the PeopleSoft application server, web servers including BEA's WebLogic and IBM's WebSphere. It also contains information about building and maintaining search indexes, database level auditing, and PeopleTools utilities.

Note. PeopleSoft supports a number of versions of UNIX and Linux in addition to Microsoft Windows. Throughout this book, there are references to operating system configuration requirements. Where necessary, this book refers to specific operating systems by name (for example, Solaris, HP/UX, Linux, etc.); however, for simplicity the word UNIX is used to refer to all UNIX-like operating systems, including Linux.

CHAPTER 1

Getting Started with System and Server Administration

This chapter provides an overview of system and server administration and discusses system and server administration implementation.

System and Server Administration Overview

This section discusses:

- PSADMIN.
- Analytic servers.
- Web servers.
- Search indexes.
- PeopleSoft Configuration Manager.
- PeopleTools utilities.
- Tracing and debugging.
- Jolt Internet Relay.
- Environment replication.
- Timeout settings.

PSADMIN

PSADMIN is the primary PeopleSoft utility for administering application server domains, as well as PeopleSoft Process Scheduler processes, and the Microsoft Windows *PeopleSoft* service.

PSADMIN also enables you to configure and manage the behavior of servers with respect to a wide range of PeopleTools infrastructure elements, including:

- BEA Tuxedo and Jolt.
- PeopleCode debugging.
- Caching.
- Analytic server framework.
- Transactional SQL requests.
- Performance enhancement.

- PeopleSoft Query.
- PeopleSoft Integration Broker.
- Application messaging.
- Email.
- Real time event notification.
- PeopleSoft Performance Monitor.
- PeopleSoft MultiChannel Framework.

You launch and run PSADMIN using a command line interface.

See Also

Chapter 2, “Using the PSADMIN Utility,” page 7

Chapter 3, “Using PSADMIN Menus,” page 27

Chapter 4, “Setting Application Server Domain Parameters,” page 43

Analytic Servers

When a program doesn’t “maintain state” or when the infrastructure of a system prevents a program from maintaining state, it’s known as a *stateless* program or system. It can’t take information about the last session into the next session, such as settings the user makes or conditions that arise during processing. All session state is maintained by the client and is transferred to the server with each request. As long as an application server is up and running, a user’s session remains active and functional, and any application server can perform requested transactions.

However, with some products, such as PeopleSoft Analytic Calculation Engine or PeopleSoft Optimization Framework, running a calculation on a multi-dimensional model is likely to produce far more data than is reasonable to shuttle between a client and server to maintain a stateless connection. For performance reasons, the calculations are performed completely in memory. If these calculations were to be synchronized and stored in the database so that a stateless connection could be maintained, performance would suffer significantly.

The *analytic server framework* provided by PeopleSoft is a general server infrastructure designed to meet the needs of PeopleSoft products that process large amounts of data in memory. It provides a stateful model of client/server connectivity that these products require to be part of the PeopleTools system, by keeping track of configuration settings, transaction information, and other data for a session. For example, client software could request that an analytic model or optimization model be recalculated in one transaction, then retrieve the results of the calculation on that model at a later time. A server process handles these requests, and maintains the model state and calculated data in memory between the requests. Additional transactions can then modify the model and perform recalculations on it without shuffling all of the data between the client and the server or dumping all the data to a database, thus preserving in-memory performance.

Web Servers

PeopleSoft supports BEA WebLogic and IBM WebSphere web servers, which both provide the same basic functionality to support PeopleSoft Pure Internet Architecture, including a console interface, secure sockets layer (SSL), and reverse proxy servers (RPS).

Each web server has its own way of accomplishing its functionality, and each adds its own extra features that you might find useful to your PeopleSoft system. This PeopleBook provides supplemental information about configuring and administering BEA WebLogic and IBM WebSphere where it has particular relevance to PeopleSoft.

Note. The information in this PeopleBook is not intended to replace any BEA WebLogic or IBM WebSphere documentation. Always refer to the manufacturer's documentation for detailed information about your web server.

See Also

Chapter 6, "Working with BEA WebLogic," page 97

Chapter 7, "Working with IBM WebSphere," page 129

Appendix A, "BEA WebLogic 8.1 Managed Server Architecture," page 279

Search Indexes

A search index is a collection of files that is used during a search to quickly find documents of interest. You build a search index to enable searching on a given set of documents. The set of files that make up the index is a *collection.* This collection contains a list of words in the indexed documents, an internal documents table containing document field information, and logical pointers to the actual document files.

Fields contain metadata about a document. For example, Author and Title might be fields in an index. *VdkVgwKey* is a special field that identifies each document and is unique to all of the documents in the collection.

Every search index can be modified by changing the configuration files that are associated with the index. These configuration files are known as *style* files and reside in the style directory under the database directory. A typical configuration of style files define fields for a particular index.

PeopleSoft software supports three types of search indexes:

- Record-based indexes.

  Record-based indexes are used to create indexes of data in PeopleSoft tables. For example, if the PeopleSoft application has a catalog record that has two fields (Description and PartID), you can create a record-based index to index the contents of the Description and PartID fields.

- HTTP spider indexes.

  HTTP spider indexes index a web repository by accessing the documents from a web server. You typically specify the starting uniform resource locator (URL). The indexer walks through all documents by following the document links and indexes the documents in that repository. You can control to what depth the indexer should traverse.

- File system indexes.

  File system indexes are similar to HTTP spider indexes, except that the repository that is indexed is a file system. You typically specify the path to a file directory, then the indexer indexes all documents within that folder. HTTP spider indexes and file system indexes are sometimes collectively referred to as *spider* indexes. The indexer recognizes a wide variety of document formats, such as Word or Excel documents. Any document in an unknown format is skipped by the indexer.

See Also

Chapter 8, "Building and Maintaining Search Indexes," page 161

PeopleSoft Configuration Manager

PeopleSoft Configuration Manager is a Microsoft Windows application that simplifies workstation administration by enabling you to adjust PeopleSoft registry settings from a central location. You can set up one workstation to reflect the environment at your site, then export the configuration file, which can be shared among all the workstations at your site. You can also define separate profiles for connecting to different PeopleSoft databases.

PeopleSoft configuration parameters are grouped on the Configuration Manager pages according to the function, feature, or tool that they control, including:

- Startup settings.
- Display settings.
- Crystal report and Business Interlink settings.
- Trace settings.
- Workflow settings.
- Remote call settings.
- Developer workstations.
- Importing and exporting environment settings.
- Defining configuration profiles.

See Also

Chapter 9, “Using PeopleSoft Configuration Manager,” page 181

PeopleTools Utilities

The PeopleTools utilities are a set of miscellaneous configuration and administratiion tools that serve as a browser-based complement to PeopleSoft Configuration Manager. These utilities, most of which are available through the PeopleTools Utilities menu, provide the ability to configure, maintain, or launch a wide range of features, including:

- The System Information page.
- The message catalog.
- The spell check dictionary.
- Translate values.
- Application server caching.
- SQR customization.
- Table management and sharing.
- Backward compatibility.
- Remote database connection.
- File attachments.
- Stored URLs.
- Mobile data synchronization.
- Update tracking.

- Platform-specific database features.
- Database auditing.
- International settings.
- Optimization utilities.
- PeopleSoft Ping.

See Also

Chapter 10, "Using PeopleTools Utilities," page 213

Tracing and Debugging

You can use the PeopleCode Debugger to interactively debug a PeopleCode program's configurations of a two-tier connection to the database or a three-tier connection to the database. You can temporarily override the PeopleSoft Configuration Manager trace settings for PeopleCode and SQL programs.

See Also

Chapter 11, "Configuring Trace and Debug Settings," page 257

Jolt Internet Relay

Jolt Internet Relay is a BEA product that's required for web client connections to an application server, in cases where the web server containing PeopleSoft HTML and applets is on a separate machine than the application server. If the web server is on the same machine as the application server, Jolt Internet Relay is not required. PeopleSoft enables configurations in which the web server and application server are on the same or separate machines.

Jolt Internet Relay consists of two parts: Jolt Relay (JRLY) and Jolt Relay Adapter (JRAD).

JRLY consists of a standalone program and configuration file that runs on the same machine as the web server. JRLY receives Jolt messages from a PeopleSoft web client and routes those messages to JRAD on the application server. It receives the Jolt message through one port, the LISTEN port, and connects to the JRAD by using another port, the CONNECT port. JRLY is sometimes referred to as a front-end relay.

JRAD runs on the same machine as the application server. It's configured automatically on the application server domain as part of the PeopleSoft PSADMIN domain configuration procedure. JRAD listens for JRLY messages on its LISTENER port and transfers the message to the JSL or JSH. JRAD is sometimes referred to as a back-end relay.

See Also

Chapter 12, "Setting Up Jolt Internet Relay," page 263

Environment Replication

Environment replication involves taking a working, well-tested environment, and copying the PeopleTools binary and configuration files to a new location to create a new environment by making minor modifications to the new copies.

To further define the term "environment", there are three separate components that can have multiple environment configurations: the web server, the application server, and PeopleSoft Process Scheduler server. You replicate each of these elements using procedures that are appropriate to its architecture.

See Chapter 13, “Replicating an Installed Environment,” page 271.

Timeout Settings

This appendix lists the delivered default timeout settings for the web server, application server, PeopleSoft Process Scheduler, and PeopleSoft Internet Architecture (PIA).

See Also

Appendix B, “PeopleSoft Timeout Settings,” page 323

System and Server Administration Implementation

The functionality of system and server administration for your PeopleSoft applications is delivered as part of the standard installation of PeopleTools, which is provided with all PeopleSoft products.

Several activities must be completed before you administer the system and servers for your implementation:

1. Install your BEA WebLogic or IBM WebSphere web server software according to the documentation provided with that product.
2. Install your PeopleSoft application according to the installation guide for your database platform.

   See *PeopleSoft Installation Guide for your platform and product line.*
3. Establish a user profile that gives you access to PeopleSoft Application Designer and any other tools and processes that you’ll use.

   See *Enterprise PeopleTools 8.46 PeopleBook: Security Administration.*

Other Sources of Information

This section provides information to consider before you begin to engage in system and server administration on your PeopleSoft system. In addition to implementation considerations presented in this section, take advantage of all PeopleSoft sources of information, including the installation guides, release notes, and PeopleBooks.

See Also

“System and Server Administration Preface,” page xxix

Enterprise PeopleTools 8.46 PeopleBook: Getting Started with PeopleTools

CHAPTER 2

Using the PSADMIN Utility

This chapter provides an overview of PeopleSoft Server Administration (PSADMIN) and discusses how to:

- Start PSADMIN.
- Use PSADMIN.
- Use configuration templates.
- Use the PSADMIN command-line interface.
- Use the Quick-Configure menu.
- Use PSADMIN executables and configuration files.
- Configure the application server to handle cache files and replay files.

Understanding PSADMIN

PSADMIN simplifies the process of configuring and administering all of the servers and features that are available on the application server. For example, you use PSADMIN to configure application server domains, process scheduler servers, and Microsoft Windows services.

Note. *PS_HOME* is the directory where you install PeopleTools.

Accessing Network Drives in Microsoft Windows 2003 Server

This section applies only if all of the following are true:

- You've upgraded to the current PeopleTools release, including the required BEA Tuxedo version and rolling patch level, from PeopleTools 8.45 or older.
- You plan to administer your application server domains in Microsoft Windows 2003 (or newer) Server.
- One or more PeopleSoft processes need to directly access a mapped network drive. Activities requiring this can include:
  - Using an instance of PSADMIN that was launched from the network drive.
  - Accessing a database on the network drive.
  - Outputting reports to a location on the network drive.

Any PeopleSoft processes that reference mapped network drives by their drive letters in this environment must be able to find the drives, and must have appropriate permission to access them. In Windows 2003 Server, the operating system does not provide this access directly.

PeopleSoft uses Tuxedo's BEA ProcMGR V8.1 service and an associated environment variable, TM_TUXIPC_MAPDRIVER, to gain access to the network drives. You must configure these elements to provide the appropriate access before you start any PeopleSoft servers or other processes.

To configure access to mapped network drives:

1. Determine which shared network directories your PeopleSoft system will need to access with a drive and directory path.
2. In your Microsoft Windows 2003 system, make sure that the shared network directories are available, and grant domain administrator privileges to access them.
3. Access the Services control panel.
4. Stop the BEA ProcMGR V8.1 service.
5. In the BEA ProcMGR V8.1 service properties, ensure that the "log on as" account is the account with domain administrator privileges for the required shared directories.
6. Define the TM_TUXIPC_MAPDRIVER environment variable to specify the drive mappings and paths of the required shared directories, with the following format:

   drive1:=*machine_name1**dirpath1*[;*drive2*:=*machine_name2**dirpath2*[...]]

   For example:

   ```
   U:=\\myMachine\e$;V:=\\myMachine\PT846
   ```

7. Start the BEA ProcMGR V8.1 service.

   The service uses the value of TM_TUXIPC_MAPDRIVER to create the necessary drive mappings, and uses its own log on settings to provide your PeopleSoft system with access to those locations.

Note. Every time your PeopleSoft system needs to access a new mapped network location, you must repeat this procedure, including the new drive mapping along with the others.

Starting PSADMIN

This section assumes that you have already installed and configured the PeopleSoft application server.

See *PeopleTools 8.46 Installation for your database platform.*

To start the PSADMIN utility:

1. At a command prompt, change to the following directory on the application server machine:

   PS_HOME\appserv

2. Launch the following command:

   psadmin

3. Select the server that you want to configure, administer, or monitor from the PSADMIN menu.

```
---------------------------------
PeopleSoft Server Administration
---------------------------------
 1) Application Server
 2) Process Scheduler
 3) Service Setup
```

```
 q) Quit
Command to execute (1-3, q):
```

Using PSADMIN

Using PSADMIN involves selecting the number of the menu item that reflects the action that you want to take, entering the correct number at the command line, and pressing ENTER. However, in some cases, you may want to take use the command-line options that PSADMIN offers.

See Chapter 2, "Using the PSADMIN Utility," Using the PSADMIN Command-Line Interface, page 11.

Selecting Menu Options

Each PSADMIN menu has the same look and feel. To select a menu item, enter the corresponding number at the prompt and press ENTER. To return to the previous menu enter *q* (quit) at the prompt.

Note. Because the numbers corresponding to the PSADMIN menu commands can change as minor releases and patches provide different features and capabilities to PSADMIN, this documentation generally refers to the names of the commands rather than their menu item numbers.

Using Configuration Templates

The initial values that you see in PSADMIN are derived from the configuration template that you select when you create your domain. The delivered templates provide a range of possible implementations. These are the delivered templates:

- Small

  Use for 1–100 users.

- Medium

  Use for 100–500 users.

- Large

  Use for 500–1000 users.

- Developer

  Use for development and demonstration environments only.

Each configuration template includes a number of server processes, such as PSAPPSRV, that is sufficient for its intended load. You can easily modify and create your own configuration templates to fully include your site's needs. The configuration templates are CFX files in the *PS_HOME*\appserv directory on the application server. To create your own CFX files, save the CFX file with a new name after modifying the template values. The next time PSADMIN prompts you for a configuration template to create a domain, the new CFX file appears in the configuration templates list.

You can modify the CFX files by using any text editor, such as Notepad on Microsoft Windows or vi on UNIX. Use the Save As option to create your own template.

Using the Quick-Configure Menu

When you create a domain for the first time, PSADMIN presents you with the most commonly changed parameters on the Quick-Configure menu, so that you can get up and running quickly. After the initial setup, you may at any time select *Configure this domain* on the PeopleSoft Domain Administration menu to access the Quick-Configure menu.

```
-----------------------------------------------
Quick-configure menu -- domain: MYDOM01
-----------------------------------------------
      Features                      Settings
     ==========                    ==========
  1) Pub/Sub Servers   : No    13) DBNAME     :[QE846DW]
  2) Quick Server      : No    14) DBTYPE     :[MICROSFT]
  3) Query Servers     : No    15) UserId     :[QEDMO]
  4) Jolt              : Yes   16) UserPswd   :[QEDMO]
  5) Jolt Relay        : No    17) DomainID   :[TESTSERV]
  6) PC Debugger       : No    18) AddToPATH  :[C:\apps\db\MSSQL\Binn]
  7) Event Notification: Yes   19) ConnectID  :[people]
  8) MCF Servers       : No    20) ConnectPswd:[peop1e]
  9) Perf Collator     : No    21) ServerName :[]
 10) Analytic Servers  : Yes   22) WSL Port   :[7000]
                               23) JSL Port   :[9000]
                               24) JRAD Port  :[9100]

      Actions
     =========
 11) Load config as shown
 12) Custom configuration
  h) Help for this menu
  q) Return to previous menu

Enter selection (1-24, h, or q):
```

The Quick-Configure menu shows which features are currently set for the newly created domain. The menu contains the values that are most commonly changed when setting up a demonstration or test domain.

To change the value of a parameter under Features, just enter the number corresponding to the feature to toggle the feature on or off.

To change the value of a parameter under Settings, enter the number corresponding to the setting and enter the new value at the prompt.

Note. All of the values that you change remain in effect until you modify them again.

Note. If you select a Settings parameter, then press ENTER without entering a new value at the prompt, the existing value of that parameter is deleted. If you then enter *q* to quit the Quick-Configure menu, your changes are discarded, and the original values remain. However, if you load the conguration as shown, the parameter is saved without a value.

If the parameter is required, you'll see an error message indicating that the configuration could not be completed. The next time you access the Quick-Configure menu, PSADMIN reloads any empty required parameter with its original default value, just as it would appear if you were creating a new domain.

To configure the rest of the parameters that are not presented on the Quick-Configure menu, select *Custom configuration* to view the full list.

The Quick-Configure menu is not intended to replace the series of configuration sections in the custom configuration interface. In most cases, your site requires the parameters and tuning options that are available only through the full custom configuration menu. For this reason, the Quick-Configure menu is provided primarily for situations where you're setting up a demonstration domain for testing or for development purposes.

Note. When you use custom configuration, pressing ENTER instead of entering a new value for a parameter does not delete the parameter's value. PSADMIN interprets ENTER to mean that you want to retain the parameter's existing value. If you want to remove the value, you can edit the parameter in the psappsrv.cfg file directly. You can then boot the domain directly from the PeopleSoft Domain Administration menu without any additional configuration steps.

Using the PSADMIN Command-Line Interface

This section provides an overview of the PSADMIN command-line interface and discusses how to:

- Use the miscellaneous commands.
- Use the application server commands.
- Use the Process Scheduler commands.

Understanding the PSADMIN Command-Line Interface

In some cases, you might want to use the PSADMIN command-line interface rather than starting the PSADMIN interface and navigating to a particular menu. The command line offers a direct method of carrying out certain tasks on the application server and PeopleSoft Process Scheduler. It also enables you to include PSADMIN actions in scripts, and simplifies the task of creating numerous domains that use default server settings.

Note. Before you begin using the PSADMIN commands, you should become familiar with PSADMIN and the components that it controls.

To use the PSADMIN command-line interface, first change at a command prompt to the *PS_HOME*\appserv directory on the application server or PeopleSoft Process Scheduler machine.

PSADMIN has several variations of its basic command-line syntax for miscellaneous activities, application server administration, and Process Scheduler administration, which are described in the following sections.

Using the Miscellaneous Commands

Following are the available miscellaneous PSADMIN commands:

Command	Description
`psadmin -h`	Displays command help and syntax.
`psadmin -v`	Displays the PSADMIN version number, as in *Version 8.46.*
`psadmin -env`	Displays your current environment variables.

Using the Application Server Commands

For application server administration, PSADMIN has two syntax formats — one for creating new application server domains, and the other for administering existing domains.

Using the Application Server Create Command

Use the following syntax to create a new application server domain:

```
psadmin -c create -d domain -t template [ -s s_set [ -p p_set ]]
```

The **create** command creates an application server configuration file with the specified domain name, using the specified configuration template.

The *domain* parameter must be the name of an application server domain that you want to create, for example, HR846DMO.

The *template* parameter must have one of the following values:

- *small*
- *medium*
- *large*
- *developer*

The *s_set* parameter is an optional string of startup values which provide initial configuration settings that you would otherwise specify on the PSADMIN application server Quick-Configure menu. You must enter the startup string as follows:

- In Windows, the values must be separated by slashes —

```
DBNAME/DBTYPE/OPR_ID/OPR_PSWD/DOMAIN_ID/ADD_TO_PATH/⇒
CNCT_ID/CNCT_PSWD/SERV_NAME/{ENCRYPT|NOENCRYPT}
```

- In UNIX, the values must be separated by percent signs —

```
DBNAME%DBTYPE%OPR_ID%OPR_PSWD%DOMAIN_ID%ADD_TO_PATH%⇒
CNCT_ID%CNCT_PSWD%SERV_NAME%{ENCRYPT|NOENCRYPT}
```

Important! You must enter these values in the order shown. You can omit required values only by truncating the string from right to left. For example, you can specify *DBNAME*/*DBTYPE*, but you can't specify *DBNAME*/*DOMAIN_ID*.

These startup settings all have default values if you omit any of them. The default values are generally the values you provided when setting up your PeopleSoft environment, and are the same as they would initially appear on the PSADMIN application server Quick-Configure menu.

The following table describes the startup settings:

Startup (*s_set*) Setting	Description
DBNAME	Enter the name of the database to which the application server will connect. This is the same as the *DBName* parameter in the Startup section of the psappsrv.cfg file. **Note.** If you don't include the *s_set* parameter, the value of this setting is the same as the domain name that you specify in the command.
DBTYPE	Enter the database type. Valid values are ORACLE, INFORMIX, SYBASE, MICROSFT, DB2ODBC, and DB2UNIX. This is the same as the *DBType* parameter in the Startup section of the psappsrv.cfg file. **Note.** Notice the spelling of MICROSFT. DB2ODBC is the database type for DB2 z/OS.
OPR_ID	Enter the user ID, such as QEDMO, for the domain to use to connect to the database. This is the same as the *UserId* parameter in the Startup section of the psappsrv.cfg file.
OPR_PSWD	Enter the user password that is associated with the specified user ID. This is the same as the *UserPswd* parameter in the Startup section of the psappsrv.cfg file.
DOMAIN_ID	Enter a domain ID, such as TESTSRV1, TESTSRV2, and so on. This does not need to match the domain name. This name is important only because the BEA Tuxedo Web Monitor uses it to identify application server domains on each machine. This is the same as the *Domain ID* parameter in the Domain Settings section of the psappsrv.cfg file.
ADD_TO_PATH	(Optional) Enter the directory path that contains your connectivity software or database drivers. This is the same as the *Add to PATH* parameter in the Domain Settings section of the psappsrv.cfg file. **Note.** If this value contains spaces, it must be in double quotes (`" "`). For example: `"c:\Program Files"`. **Important!** If you want this setting to be blank, but you can't truncate the string to this point (you still need to specify a value for *CNCT_ID*), you can specify a value of "_____" (five underscores without the quotes) in this position. PSADMIN interprets this as a blank value.

Startup (*s_set*) Setting	Description
CNCT_ID	Enter the connect ID, which is required for all platforms. This is the same as the *ConnectId* parameter in the Startup section of the psappsrv.cfg file. See *Enterprise PeopleTools 8.46 PeopleBook: Security Administration.*
CNCT_PSWD	Enter the password that is associated with the connect ID. This is the same as the *ConnectPswd* parameter in the Startup section of the psappsrv.cfg file.
SERV_NAME	(Optional) If your RDBMS requires that you specify the server name on which the database resides, enter the appropriate server name. This is the same as the *ServerName* parameter in the Startup section of the psappsrv.cfg file. **Important!** If you want this setting to be blank, but you can't truncate the string to this point (you still need to specify the ENCRYPT setting), you can specify a value of "_____" (five underscores without the quotes) in this position. PSADMIN interprets this as a blank value.
ENCRYPT | NOENCRYPT	Specify ENCRYPT to encrypt the values of the *UserPswd* and *ConnectPswd* parameters in the psappsrv.cfg file. If you specify NOENCRYPT (the default value), these values appear in clear text in the file.

The *p_set* parameter is an optional string of port numbers that you would otherwise specify on the PSADMIN application server Quick-Configure menu. Typically, you include this parameter only if you have more than one domain on the same application server machine or if you need to provide a specific value due to your environment or testing needs. Otherwise, you should accept the defaults for easy configuration.

You must specify the port numbers as follows:

- In Windows, the values must be separated by slashes —

  ```
  WSL_PORT/JSL_PORT/JRAD_PORT
  ```

- In UNIX, the values must be separated by percent signs —

  ```
  WSL_PORT%JSL_PORT%JRAD_PORT
  ```

Important! You must enter these values in the order shown. You can omit values only by truncating the string from right to left. For example, you can specify `WSL_PORT/JSL_PORT`, but you can't specify `WSL_PORT/JRAD_PORT`. These port numbers all have default values if you omit any of them. The default values are the values you provided when setting up your PeopleSoft environment, and are the same as they would initially appear on the PSADMIN application server Quick-Configure menu.

The following table describes the port settings:

Port (*p_set*) Setting	Description
WSL_PORT	Workstation listener port number. This is the same as the *Port* parameter in the Workstation Listener section of the psappsrv.cfg file.
JSL_PORT	BEA Jolt port number. This is the same as the *Port* parameter in the JOLT Listener section of the psappsrv.cfg file. **Note.** Specify this value only if you intend for the domain to support browser deployment.
JRAD_PORT	BEA Jolt internet relay port number. This is the same as the *Listener Port* parameter in the JOLT Relay Adapter section of the psappsrv.cfg file. **Note.** Specify this value only if you intend for the domain to support browser deployment, *and* your web server resides on a separate machine from the application server.

Following is an example of the application server **create** command:

```
psadmin -c create -d HR846DOM -t small -s HR846DB1/MICROSFT/PS/PS/TESTSRV2/⇒
"c:\my apps\db\mssql7\binn"/people/peop1e/_____/ENCRYPT -p 7100/9010/9100
```

When you launch the command, you'll see progress messages similar to the following:

```
Copying application server configuration files...
copying [small.cfx] to [HR846DOM\psappsrv.cfg]
Copying Jolt repository file...
Domain created.
Loading UBBGEN configuration utility with "-s HR846DB1/MICROSFT/PS/PS/TESTSRV2/⇒
c:\my apps\db\mssql7\binn/people/peop1e/_____/ENCRYPT -p 7100/9010/9100"...
 setting DBName=HR846DB1
 setting DBType=MICROSFT
 setting OprId=PS
 setting OprPswd=ICMFyd/wUA0=
 setting ConnectId=people
 setting ConnectPswd=MbVHsgc6/pM=
 setting ServerName=
 setting Port=7100
 setting Port=9010
 setting Listener Port=9100
 setting Domain ID=TESTSRV2
 setting Add to PATH=c:\my apps\db\mssql7\binn
New CFG file written with modified Startup parameters
Log Directory entry not found in configuration file.
Setting Log Directory to the default... [PS_SERVDIR\LOGS]
PSAUTH Spawning disabled because Max Instances <= Min Instances.
Configuration file successfully created.
CFG setting changes completed, loading configuration...
```

Using the Application Server Administration Commands

Use the following syntax to administer an existing application server domain:

```
psadmin -c command -d domain
```

The *domain* parameter must be the name of an application server domain that you want to administer, for example, HR846DMO.

The valid values of the *command* parameter are as follows:

Command	Example	Result of the Example
boot	`psadmin -c boot -d PS846DMO`	Boots an application server domain named PS846DMO.
configure	`psadmin -c configure -d PS846DMO`	Invokes the configuration editor for the PS846DMO domain.
pslist	`psadmin -c pslist -d PS846DMO`	Displays the processes that have been booted for the PS846DMO domain. This includes the system process ID for each process.
shutdown	`psadmin -c shutdown -d PS846DMO`	Shuts down the PS846DMO application server domain, by using a normal shutdown method. In a normal shutdown, the domain waits for users to complete their tasks and turns away new requests before terminating all of the processes in the domain.
shutdown!	`psadmin -c shutdown! -d PS846DMO`	Shuts down the PS846DMO application server domain by using a forced shutdown method. In a forced shutdown, the domain *immediately* terminates all of the processes in the domain.
sstatus	`psadmin -c sstatus -d PS846DMO`	Displays the BEA Tuxedo processes and PeopleSoft server processes that are currently running in the PS846DMO application server domain.
cstatus	`psadmin -c cstatus -d PS846DMO`	Displays the currently connected users in the PS846DMO application server domain.
qstatus	`psadmin -c qstatus -d PS846DMO`	Displays status information about the individual queues for each server process in the PS846DMO application server domain.

Using the Process Scheduler Commands

For Process Scheduler administration, PSADMIN has two syntax formats — one for creating new Process Scheduler configurations, and the other for administering existing configurations.

Using the Process Scheduler Create Command

Use the following syntax to create a new Process Scheduler configuration:

```
psadmin -p create -d database -t template [ -ps ps_set ]
```

The PSADMIN **create** command creates a Process Scheduler configuration file for the specified database, using the specified configuration template.

The *database* parameter must be the name of a database that's associated with a PeopleSoft Process Scheduler Server Agent, for example, HR846DMO.

The *template* parameter must be the name of a .cfx file located in *PS_HOME*\appserv\prcs, without the extension. This represents the operating system platform on which you're running PeopleSoft Process Sheduler. For example, to use the template file called nt.cfx on a Windows machine, specify the value *nt*.

The *ps_set* parameter is an optional string of startup values which provide initial configuration settings that you would otherwise specify on the PSADMIN Process Scheduler Quick-Configure menu. You must enter the startup string as follows:

- In Windows, the values must be separated by slashes —

  DBNAME/*DBTYPE*/*PRCSSERVER*/*OPR_ID*/*OPR_PSWD*/*CNCT_ID*/*CNCT_PSWD*/*SERV_NAME*/*LOGOUT_DIR*/*SQRBIN*/*ADD_TO_PATH*/*DBBIN*/{ENCRYPT|NOENCRYPT}

- In UNIX, the values must be separated by commas —

  DBNAME,*DBTYPE*,*PRCSSERVER*,*OPR_ID*,*OPR_PSWD*,*CNCT_ID*,*CNCT_PSWD*,*SERV_NAME*,*LOGOUT_DIR*,*SQRBIN*,*ADD_TO_PATH*,{ENCRYPT|NOENCRYPT}

Note. The UNIX syntax does not include the DBBIN setting.

Important! You must enter these values in the order shown. You can omit required values only by truncating the string from right to left. For example, you can specify *DBNAME*/*DBTYPE*, but you can't specify *DBNAME*/*LOGOUT_DIR*.

These startup settings all have default values if you omit any of them. The default values are generally the values you provided when setting up your PeopleSoft environment, and are the same as they would initially appear on the PSADMIN Process Scheduler Quick-Configure menu.

Note. Because these PeopleSoft Process Scheduler settings are already documented in the *Enterprise PeopleTools 8.46 PeopleSoft Process Scheduler PeopleBook*, this section provides only a basic overview of the relationship between the settings on the command line and the equivalent settings on the PSADMIN Process Scheduler Quick-Configure menu.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Process Scheduler*, "Using the PSADMIN Utility".

The following table describes the startup settings:

Startup (*ps_set*) Setting	Description
DBNAME	This is the equivalent of the *DBName* parameter on the PSADMIN Process Scheduler Quick-Configure menu. **Note.** If you don't include the *ps_set* parameter, the value of this setting is the same as the database name that you specify in the command.
DBTYPE	This is the equivalent of the *DBType* parameter on the PSADMIN Process Scheduler Quick-Configure menu.
PRCSSERVER	This is the equivalent of the *PrcsServer* parameter on the PSADMIN Process Scheduler Quick-Configure menu.
OPR_ID	This is the equivalent of the *UserId* parameter on the PSADMIN Process Scheduler Quick-Configure menu.
OPR_PSWD	Enter the user password that is associated with the specified user ID. This is the equivalent of the *UserPswd* parameter on the PSADMIN Process Scheduler Quick-Configure menu.
CNCT_ID	This is the equivalent of the *ConnectId* parameter on the PSADMIN Process Scheduler Quick-Configure menu.
CNCT_PSWD	This is the equivalent of the *ConnectPswd* parameter on the PSADMIN Process Scheduler Quick-Configure menu.
SERV_NAME	(Optional) This is the equivalent of the *ServerName* parameter on the PSADMIN Process Scheduler Quick-Configure menu. **Important!** If you want this setting to be blank, but you can't truncate the string to this point (you still need to specify a value for *LOGOUT_DIR*), you can specify a value of "_____" (five underscores without the quotes) in this position. PSADMIN interprets this as a blank value.
LOGOUT_DIR	This is the equivalent of the *Log/Output Dir* parameter on the PSADMIN Process Scheduler Quick-Configure menu. **Note.** If this value contains spaces, it must be in double quotes (`" "`). For example: `"c:\psft app\log_output"`.
SQRBIN	This is the equivalent of the *SQRBIN* parameter on the PSADMIN Process Scheduler Quick-Configure menu. **Note.** If this value contains spaces, it must be in double quotes (`" "`). For example: `"C:\my pt846\bin\sqr\MSS\binw"`.
ADD_TO_PATH	(Optional) This is the equivalent of the *AddToPATH* parameter on the PSADMIN Process Scheduler Quick-Configure menu. **Note.** If this value contains spaces, it must be in double quotes (`" "`). For example: `"%WINDIR%\SYSTEM32;c:\Program Files"`.

Startup (*ps_set*) Setting	Description
DBBIN	(Windows only) This is the equivalent of the *DBBIN* parameter on the PSADMIN Process Scheduler Quick-Configure menu. **Note.** If this value contains spaces, it must be in double quotes (" "). For example: "C:\my apps\db\MSSQL\Binn".
ENCRYPT | NOENCRYPT	Specify ENCRYPT to encrypt the values of the *UserPswd* and *ConnectPswd* parameters in the psprcs.cfg file. If you specify NOENCRYPT (the default value), these values appear in clear text in the file.

Following is an example of the Process Scheduler **create** command:

```
psadmin -p create -d HR846DB1 -t nt -ps HR846DB1/MICROSFT/PSNT/⇒
PS/PS/people/peop1e/_____/"c:\psft app\log_output"/c:\hr846\bin\sqr\MSS\binw/⇒
c:\WINNT\SYSTEM32/c:\apps\db\mssql7\binn/ENCRYPT
```

Using the Process Scheduler Administration Commands

Use the following syntax to administer an existing Process Scheduler configuration:

```
psadmin -p command -d database
```

The *database* parameter must be the name of a database that's associated with a PeopleSoft Process Scheduler Server Agent, for example, HR846DMO.

The valid values of the *command* parameter are as follows:

Command	Example	Result of the Example
start	psadmin -p start -d hr846dmo	Starts a process scheduler.
stop	psadmin -p stop -d hr846dmo	Stops a process scheduler.
configure	psadmin -p configure -d hr846dmo	Configures a process scheduler.
status	psadmin -p status -d hr846dmo	Displays the status of a process scheduler.

See Also

Chapter 3, "Using PSADMIN Menus," Using the Process Scheduler Menu, page 35

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Process Scheduler, "Using the PSADMIN Utility"

Using PSADMIN Executables and Configuration Files

This section provides an overview of PSADMIN executables and configuration files and discusses how to:

- Configure a domain.
- Load a configuration.
- Archive application server configuration files.
- Boot a domain.
- Stop a domain.
- Monitor a domain.

Understanding PSADMIN Executables and Configuration Files

You can create, configure, and boot an application server domain from the PSADMIN interface or through its command-line options.

The executables are:

- PSADMIN.EXE

  This PeopleSoft executable resides in PS_HOME\appserv.

- UBBGEN.EXE

  This PeopleSoft executable resides in *PS_HOME*\bin\server\winx86.

- TMLOADCF.EXE

  This BEA Tuxedo executable resides in TUXDIR\bin.

  Note. TUXDIR is the directory where you install BEA Tuxedo.

- TMBOOT.EXE

  This BEA Tuxedo executable resides in TUXDIR\bin.

- TMSHUTDOWN.EXE

  This BEA Tuxedo executable resides in TUXDIR\bin.

The configuration and data files on which the executables rely all reside in *PS_HOME*\appserv*domain_name*. Each domain has its own set of these files:

- PSAPPSRV.CFG

  This is the catch-all configuration file that contains the entire collection of configuration values for a given application server domain.

- PSAPPSRV.UBX

  This is the template or model file for the PSAPPSRV.UBB file.

- PSAPPSRV.UBB

  This file stores and passes all of the domain values to the BEA Tuxedo load configuration program (tmloadcf.exe).

- PSAPPSRV.PSX

  This is the template or model file specifically for the application messaging server configuration sections, such as PSBRKRSRV, PSSUBSRV, and so on.

- PSAPPSRV.ENV

This contains environment information, such as the PS_HOME on the application server machine.

- PSAPPSRV.VAL

  This contains the format specification for the configuration parameters and, for some parameters, a set of valid values that can assigned. This helps to prevent administrators from entering invalid values.

- PSTUXCFG

  This contains PeopleSoft and BEA Tuxedo information regarding the location of executables, files, and command lines for server processes. This file is required to boot a domain.

- JREPOSITORY

  This file contains a list of the services handled by the Application Server on behalf of the JOLT (web server) client.

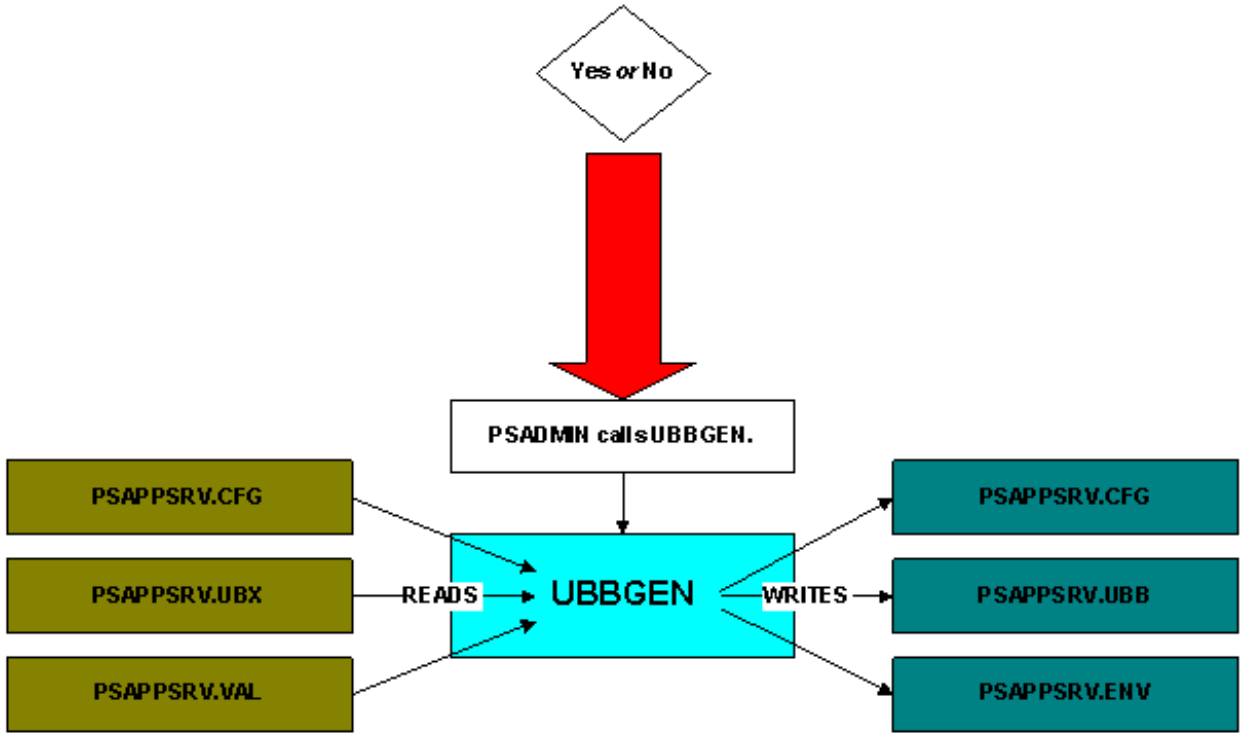
Configuring a Domain

Regardless of how you specify domain values, ultimately you must run PSADMIN to generate some necessary files that include your specific values. In the following example, PSADMIN invokes another PeopleSoft executable, UBBGEN, which reads the values and format in the psappsrv.cfg, psappsrv.val, and psappsrv.ubx files, and generates the psappsrv.ubb and psappsrv.env files:

```
1) Boot this domain
2) Domain shutdown menu
3) Domain status menu
4) Configure this domain
5) TUXEDO command line (tmadmin)
6) Edit configuration/log files menu
7) Messaging Server Administration menu
q) Quit

Command to execute (1-7, q) :4


Removing any existing configuration...
Generating new configuration...
Loading validation table...
Do you want to change any config values (y/n)? [n]:
```



Example of the UBBGEN executable

Where you see *Do you want to change any config values? (y/n),* regardless of what you enter, PSADMIN calls UBBGEN.

If you have already entered values manually in the psappsrv.cfg file and enter n, UBBGEN reads those values and writes to the necessary files.

If you enter y, you see the PSADMIN prompt interface, which is actually a wrapper to UBBGEN. UBBGEN reads the previous values in the psappsrv.cfg, presents those values, and allows you to change them. It presents the values in the format that is derived from reading the PSAPPSRV.UBX file, and it validates selected values based on criteria in the PSAPPSRV.VAL file.

Note. In the previous example, UBBGEN both reads from and writes to the psappsrv.cfg file. It reads the previous values or defaults and, if any values are modified, it writes the new values to the new psappsrv.cfg file.

Here are the scenarios by which you can configure a domain:

- Start PSADMIN, and enter values at all of the prompts.

  This generates all of the necessary files automatically.

- Edit the psappsrv.cfg file.

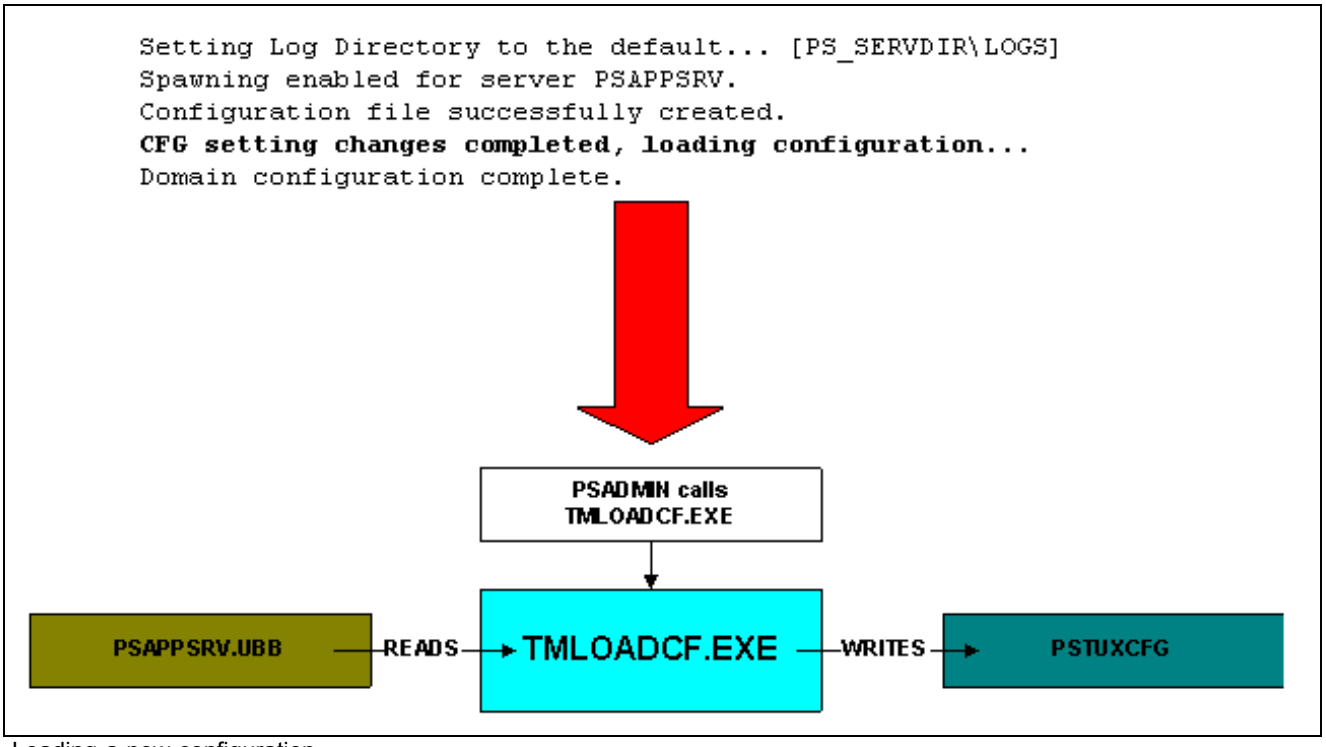
  If you decide not to use PSADMIN you must complete the following tasks in order:

  - From the command line, create a domain based on a particular template.
  - Edit the psappsrv.cfg file in a text editor.
  - Issue the configure command from the PSADMIN command line. This is the command that calls UBBGEN. You see the following after issuing this command:

```
cd ps_home\Appserv>
psadmin -c configure -d 80manual
Loading UBBGEN configuration utility ...
```

Loading a Configuration

After you configure a domain and PSADMIN creates the new configuration file, PSADMIN loads the new configuration settings into PSTUXCFG so that the domain can properly boot. This occurs automatically after you have completed all of the prompts for values in PSADMIN. The following example shows loading a new configuration on the command line:



Loading a new configuration

To load the new configuration, PSADMIN calls the BEA executable, TMLOADCF.EXE, which populates the PSTUXCFG file. TMLOADCF.EXE reads the newly entered values that appear in the PSAPPSRV.UBB file and writes them to the PSTUXCFG file.

Archiving Application Server Configuration Files

To track changes made to the psappsrv.cfg file and the history of the changes, a subdirectory, archive, stores various versions of the CFG file. You can find this subdirectory in the domain name directory, as in C:\pt844\appserv\domain name\archive, where the current version of psappsrv.cfg resides.

When you boot the application server domain for the first time, PSADMIN places a copy of psappsrv.cfg in the archive directory. In subsequent boots, if PSADMIN detects a change in psappsrv.cfg based on the time stamp values, it replaces the current psappsrv.cfg with the latest version. The file name of the new version is then psappsrv_mmddyy_hhmm_ss.cfg, as displayed on the time stamp.

Booting a Domain

When you select Boot this domain, PSADMIN calls the BEA Tuxedo executable, TMBOOT.EXE, which uses the information that resides in the PSAPPSRV.ENV and PSTUXCFG files to boot the appropriate domain.

Stopping a Domain

When you select Domain shutdown menu and select one of the shutdown options, PSADMIN calls the BEA Tuxedo executable, TMSHUTDOWN.EXE, which also uses the information that resides in the PSAPPSRV.ENV and PSTUXCFG files to shut down the appropriate domain.

Following a successful domain shutdown, PSADMIN checks and stops orphaned processes in the domain. If PSADMIN identifies and stops any orphaned server processes, it displays a screen message at the end of the shutdown operation.

Monitoring a Domain

To detect any orphaned application server processes, a server process, PSWATCHSRV, monitors the application server domain. Every two minutes, PSWATCHSRV identifies and stops any hung or orphaned server processes. If any hung or orphaned processes are found, it prints a message to the application server log file. The PSWATCHSRV process is the first process to start when you boot up the domain and the last one to stop when you shut down the domain.

Domain ID Name

To identify orphaned application server processes, all server processes within a server's domain must be uniquely identified. Therefore, the system appends a unique number to the domain ID in the psappsrv.cfg file. If you refer to domain IDs in scripts or processes, you may need to change those to reflect the new naming convention.

The command line varies slightly depending on the application server process, but it looks like this:

```
PSAPPSRV -C dom=pt84_52692 ...
```

Configuring the Application Server to Handle Cache Files and Replay Files

When an application server instance crashes, cache files and replay files are generated automatically. Over time, the size of these files can consume a large amount of disk space if there are recurring crashes in a domain. To minimize the buildup of cache files and replay files, you can modify the psappsrv.cfg file based on the following rules:

- When a crash occurs, the system creates a directory in the domain's LOGS directory.
- The dump file is saved in a directory within the domain's LOGS directory.
- The DumpMemoryImageATCrash setting in the Trace section of the psappsrv.cfg file saves the memory image of the failed process in Microsoft Windows.

  This functionality is not available to non-Windows platform. If the value of DumpMemoryImageAtCrash is MINI, a miniature memory image (with a size less than or equal to 64K) is generated. If the value is FULL, then a full memory image is created. Depending on how much memory is consumed by the application, this full memory image can be quite large. The location of the memory image is the same as the replay file.
- The settings for DumpManagerObjectsAtCrash and DumpMemoryImageAtCrash are dynamic.

  That is, the application server doesn't need to be restarted for these settings to be effective.
- If DumpManagerObjectsAtCrash is set to Y, then the application server instance:
  1. Generates the replay file.
  2. Dumps the customized objects being used by the current service request into the special cache directory.

     The cache directory resides in the same location as the replay file.
  3. If the value of DumpMemoryImageAtCrash is NONE and the platform is set to MS Windows (Win NT or Win 2000), a miniature memory image is created.
- There is no separate setting for generating the replay file.

  This file is generated as mentioned previously.
- Irrespective of the setting in DumpManagerObjectsAtCrash, a summary report of objects in each managed type for which at least one object is loaded in memory is written to the dump file or application log file.

  The summary report resembles the following example:

```
PDM Definitions: Total=36   Customized=0   In-Use=10
 RDM Definitions: Total=53   Customized=52   In-Use=50
 MDM Definitions: Total=1   Customized=0   In-Use=0
 PCM Definitions: Total=199  Customized=0   In-Use=3
 PGM Definitions: Total=1   Customized=0   In-Use=1
 CRM Definitions: Total=67   Customized=0   In-Use=0
 SSM Definitions: Total=1   Customized=0   In-Use=1
 CLM Definitions: Total=1   Customized=0   In-Use=0
 UPM Definitions: Total=1   Customized=0   In-Use=0
```

  Total indicates the total number of in-memory definitions being used by the current service. Customized indicates how many of those objects are customized, and In-Use indicates how many of those objects were being used at the time of the crash.

If DumpManagerObjectsAtCrash is set to Y, the summary for each managed object type follows the list of configured objects that are being dumped as part of the crash information gathering. If a configured object is in use, its name is prefixed with an asterisk.

A sample report for a managed object type follows:

```
RDM(PSOPTIONS/ENG)
 *RDM(PSTREEDEFNLABLS/ENG)
 RDM Definitions: Total=10   Customized=2   In-Use=1
```

Note. The asterisk that precedes the object name indicates that this object is being used by the current service request.

CHAPTER 3

Using PSADMIN Menus

This chapter discusses how to:

- Use the Application Server Administration menu.
- Use the Process Scheduler menu.
- Set up the PeopleSoft Windows service.

Using the Application Server Administration Menu

This section discusses how to:

- Access the application server options.
- Administer a domain.
- Boot a domain.
- Shut down a domain.
- Perform a normal shutdown.
- Perform a forced shutdown.
- Check the domain status.
- Purge the domain cache.
- Configure a domain.
- Edit configuration and log files.
- Create a domain.
- Delete a domain.

Accessing the Application Server Options

To access the menu options for configuring and administering an application server, select *Application Server* from the PeopleSoft Server Administration (PSADMIN) menu.

The PeopleSoft Application Server Administration menu appears.

The menu options and parameters within the Create a domain and Delete a domain menus are straightforward, one-time tasks (per domain). The Administer a domain menu offers numerous configuration, administration, and logging parameters that you may access frequently.

Administering a Domain

To administer a domain, you must have already created a domain. After you have created a domain, specify environment-specific settings for the application server to function correctly with your system. The following sections describe all of the menus and menu options that you use to administer and configure an application server domain.

To administer a domain:

1. Select *Administer a domain* from the PeopleSoft Application Server Administration menu.
2. In the Select domain number to administer command line, enter the number that corresponds to the previously created domain that you want to administer that appears in the BEA Tuxedo domain list.
3. Select the option that you want to perform from the PeopleSoft Domain Administration menu.

PSADMIN transparently sets several environment variables before invoking any Tuxedo administrative commands. You don't need to set these variables manually. These environment variables are:

- TUXCONFIG = *PS_HOME*/appserv/*domain_name*/PSTUXCFG
- APPDIR = *PS_HOME*/appserv/*domain_name*
- PATH = TUXDIR/bin; *PS_HOME*/bin/server/winx86; PATH
- APP_PW = Application Password (initialize)

The following sections describe each option that appears in the PeopleSoft Domain Administration menu

Booting a Domain

This boots the BEA Tuxedo domain (the application server) by using the tmboot command. This command will start all of the server processes that have been configured for your domain.

Shutting Down a Domain

The PeopleSoft Domain Shutdown menu offers two options: a normal shutdown and a forced shutdown.

```
------------------------------
PeopleSoft Domain Shutdown Menu
------------------------------
   Domain Name: ps800dmo

 1) Normal shutdown
 2) Forced shutdown
 q) Quit

Command to execute (1-2, q) [q]:
```

Performing a Normal Shutdown

A normal shutdown is a quiescent shutdown that waits for users to complete their tasks and turns away new requests before terminating all of the processes in the domain.

Performing a Forced Shutdown

A forced shutdown is a nonquiescent shutdown that *immediately* terminates all of the processes in the domain. Normally, you use the forced shutdown only when a Bulletin Board Liaison (BBL) process encounters errors and cannot be shut down by using a normal shutdown.

Note. The BBL is a primary BEA Tuxedo process that controls the domain.

Checking the Domain Status

Use the PeopleSoft Domain Status menu to view the status of the server, queues, or clients connected to the domain.

```
-----------------------------
PeopleSoft Domain Status Menu
-----------------------------
   Domain Name: ps800dmo

 1) Server status
 2) Client status
 3) Queue status
 q) Quit

Command to execute (1-3, q) [q]:
```

Server Status

Select *Server status* to invoke the BEA Tuxedo tmadmin psr subcommand (print server processes), which displays the BEA Tuxedo processes and PeopleSoft server processes that are currently running. For example:

```
Prog Name      Queue Name  Grp Name      ID RqDone Load Done Current Service
---------      ----------  --------      -- ------ --------- ---------------
BBL.exe        43054       MJOHNST+       0     10       500 (  IDLE )
PSMONITORSRV.e MONITOR     MONITOR        1      0         0 (  IDLE )
PSAPPSRV.exe   APPQ        APPSRV         1      0         0 (  IDLE )
PSWATCHSRV.exe WATCH       WATCH          1      0         0 (  IDLE )
PSAPPSRV.exe   APPQ        APPSRV         2      8       400 PortalRegistry
PSPPMSRV.exe   PPMQ2       PPMGRP       100      0         0 (  IDLE )
PSSAMSRV.exe   SAMQ        APPSRV       100      0         0 (  IDLE )
PSRENSRV.exe   RENQ1       RENGRP       101      0         0 (  IDLE )
WSL.exe        00001.00020 BASE          20      0         0 (  IDLE )
JSL.exe        00095.00200 JSLGRP       200      0         0 (  IDLE )
JREPSVR.exe    00094.00250 JREPGRP      250      6       300 (  IDLE )
```

The number of items appearing depends on the number of server processes that you have configured.

Client Status

Select *Client status* to invoke the BEA Tuxedo tmadmin pclt subcommand (printclient), which displays connected users. For example:

```
    LMID         User Name       Client Name    Time    Status  Bgn/Cmmt/Abrt
--------------- --------------- --------------- -------- ------- -------------
MJOHNST2040403  NT              WSH              0:03:56 IDLE    0/0/0
MJOHNST2040403  NT              JSH              0:03:55 IDLE    0/0/0
MJOHNST2040403  PTWEBSERVER     MJOHNST2040403   0:01:25 IDLE/W  0/0/0
MJOHNST2040403  QEDMO           mjohnst2032202+  0:01:09 IDLE/W  0/0/0
MJOHNST2040403  NT              tmadmin          0:03:54 IDLE    0/0/0
```

Queue Status

Examining the status of the individual queues for each server process provides valuable tuning information. Check the queues by using the Queue status option. In the following example, the results of the Queue status option show the individual server processes, the associated queue, the number of server processes currently running, and the number of requests waiting to be processed:

```
Prog Name      Queue Name  # Serve Wk Queued  # Queued  Ave. Len    Machine
---------      ------------------ ---------  --------  --------    -------
JSL.exe        00095.00200       1          -         0         - MJOHNST20+
JREPSVR.exe    00094.00250       1          -         0         - MJOHNST20+
PSMONITORSRV.e MONITOR           1          -         0         - MJOHNST20+
PSSAMSRV.exe   SAMQ              1          -         0         - MJOHNST20+
BBL.exe        43054             1          -         0         - MJOHNST20+
PSWATCHSRV.exe WATCH             1          -         0         - MJOHNST20+
PSPPMSRV.exe   PPMQ2             1          -         0         - MJOHNST20+
WSL.exe        00001.00020       1          -         0         - MJOHNST20+
PSRENSRV.exe   RENQ1             1          -         0         - MJOHNST20+
PSAPPSRV.exe   APPQ              2          -         1         - MJOHNST20+
```

The results alert you to any bottlenecks that may be occurring on your application server. With this information, you can make more informed performance decisions. For instance, if the bottlenecks appear to be persistent, it may indicate that you need to add more instances of a particular server process, such as PSAPPSRV for example. Or the results may indicate that you need to start either a PSQCKSRV or a PSQRYSRV.

Purging the Domain Cache

A proven technique for resolving problem application server environments is to purge the application server domain cache located in *PS_HOME*\appserv*domain_name*\CACHE.

Important! You should purge the cache only after due consideration, and in consultation with PeopleSoft.

Please keep the following in mind:

- You can purge only non-shared cache.
- You can purge the cache regardless of whether the application server domain is running; there's no need to shut it down and reboot. However, the procedure is less disruptive and runs more quickly if the domain is shut down or its activity level is low.
- Purging the cache can take five minutes or more on a large or busy domain, depending on the domain configuration.

To purge the domain cache:

1. On the PeopleSoft Domain Administration menu, select Purge Cache.

   If the cache is currently empty, the purge operation is cancelled, and the PeopleSoft Domain Administration menu reappears.

   If the cache is not empty, the following prompt appears:

   ```
   Enter log comments about this purge, if any (maximum 256 characters):
   ```

2. Enter any information (up to 256 characters) that you want recorded explaining the circumstances of this cache purge operation, and press ENTER. Your comments will be saved to a purge log file.

   The following prompt appears:

   ```
   Do you wish to archive the contents of the current cache? (y/n) [n] :
   ```

3. Enter *y* to archive the cache contents, or *n* to delete them permanently. The default response is *n*.

   If you enter *y*, the following prompt appears:

   ```
   Cache contents will be archived to⇒

    PS_HOME\appserv\domain_name\Archive\CACHE_mmddyy_hhmm_ss.
   Hit Enter to continue or provide a new location:
   ```

   Note. At runtime, *PS_HOME* and *domain_name* are replaced with values appropriate to your system, and *mmddyy_hhmm_ss* represents the date and time of the cache purge operation.

4. (If you chose to archive the cache contents) Enter a different archive location if desired, and press ENTER.

   If the location you enter is rejected, the following message appears, and you're prompted to continue:

   ```
   Failed to archive cache to location.
   ```

   Note. Continuing this procedure with an invalid location will purge the cache without archiving.

   The default location is a unique directory name. Keep in mind that the location you enter might have been rejected for the following reasons:

   - The directory can't be created due to an invalid drive or network mount.
   - The directory can't be created due to insufficient user privileges.
   - The directory has insufficient space for the cache files.

5. When prompted to continue, enter *y* to continue the purge operation, or *n* to cancel the operation and return to the PeopleSoft Domain Administration menu.

   Note. Archiving the cache increases the time required to complete the purge, because the cache files must be copied to the archive location.

If the application server domain is running, you might see messages such as the following, which are normal and don't require any action:

```
INFO: BEA Tuxedo, Version 8.1, 32-bit, Patch Level 118
INFO: Serial #: 650522264137-1048416937507, Expiration NONE, Maxusers 1000000
INFO: Licensed to: PeopleSoft - ISV
```

When the cache is successfully purged, the following message appears:

```
Purge Cache operation completed successfully.

You may notice that the cache directory is non-empty.⇒
 Cache files have been invalidated and will be refreshed from the database.
```

If the cache was archived, you'll also see the following:

```
You may also have noticed a number of Sharing Violation messages⇒
 during the Cache Purge. These messages are no cause for alarm⇒
 and are expected as part of the cache archival.
```

If the application server domain is running, an entry is written to the application server log file to indicate that the cache has been purged.

The purge log file is saved (including any comments you entered in step 2) as *PS_HOME*\appserv*domain_name*\LOGS\PurgeCache_*mmddyy_hhmm_ss*.log.

Note. At runtime, *PS_HOME* and *domain_name* are replaced with values appropriate to your system, and *mmddyy_hhmm_ss* represents the date and time of the cache purge operation.

Following is an example of the purge log file contents:

```
Date:02/17/05 11:47
User Explanation: Processes appeared to take a long time to recycle.
Cache Contents archived to C:\ptdvl\appserv\Q846RET\Archive\CACHE_021705_1147_01
```

Configuring a Domain

This option prompts you with a model configuration file to gather such parameters as port numbers, the number of various server processes that are needed, encryption enabling, and so forth. PSADMIN then invokes a subprogram, UBBGEN, which takes the configuration parameters, builds the file *PS_HOME*/appserv/*domain-name*/psappsrv.ubb, and carries out the *tmloadcf - y psappsrv.ubb* command to generate the following binary file: *PS_HOME*/appserv/*domain-name*/pstuxcfg.

The following topics describe all of the parameters that you encounter while configuring an application server. Either read this section before you fine tune the configuration of your application server or have it available while you are doing it.

To configure a domain:

1. Select *Configure this domain* from the PeopleSoft Domain Administration menu.

   Enter *n* (No), if you do not want to continue. This returns you to the previous menu. Otherwise, enter *y* (Yes).

2. When prompted to change configuration values, enter y.

   If you don't need to change any of the values, enter n. By doing so, you create a new configuration file with the same values that were previously specified. Enter n, or elect not to modify the PSADMIN parameters, if:

   - You have changed only the location of TUXDIR.
   - You would rather edit the PSAPPSRV.CFG file manually.
   - You installed a new BEA Tuxedo patch.

Editing Configuration and Log Files

Use the Edit Configuration/Log Files menu to view the application server and BEA Tuxedo log files. You can also manually edit the PSAPPSRV.CFG file if you do not want to use the PSADMIN interface.

To have PSADMIN start your text editor (such as Notepad or KEDIT) so that you can manually edit or view application server configuration and log files, you must specify the text editor in the environment settings. For example, to use KEDIT, the editor environment setting should look like this:

```
set EDITOR=c:\apps\kedit\keditw32.exe
```

To use Notepad, it should look like this:

```
set EDITOR=c:\Windows\Notepad.exe
```

Note. You can view and edit a domain's PSAPPSRV.CFG file while the domain is running, but the changes that you specify do not take effect until the next time you reconfigure the domain.

For the following options, you must enter your operator ID to view and edit the files:

```
Edit PSAPPSRV.tracesql (PSAPPSRV SQL trace file)
Edit PSSAMSRV.tracesql (PSSAMSRV SQL trace file)
```

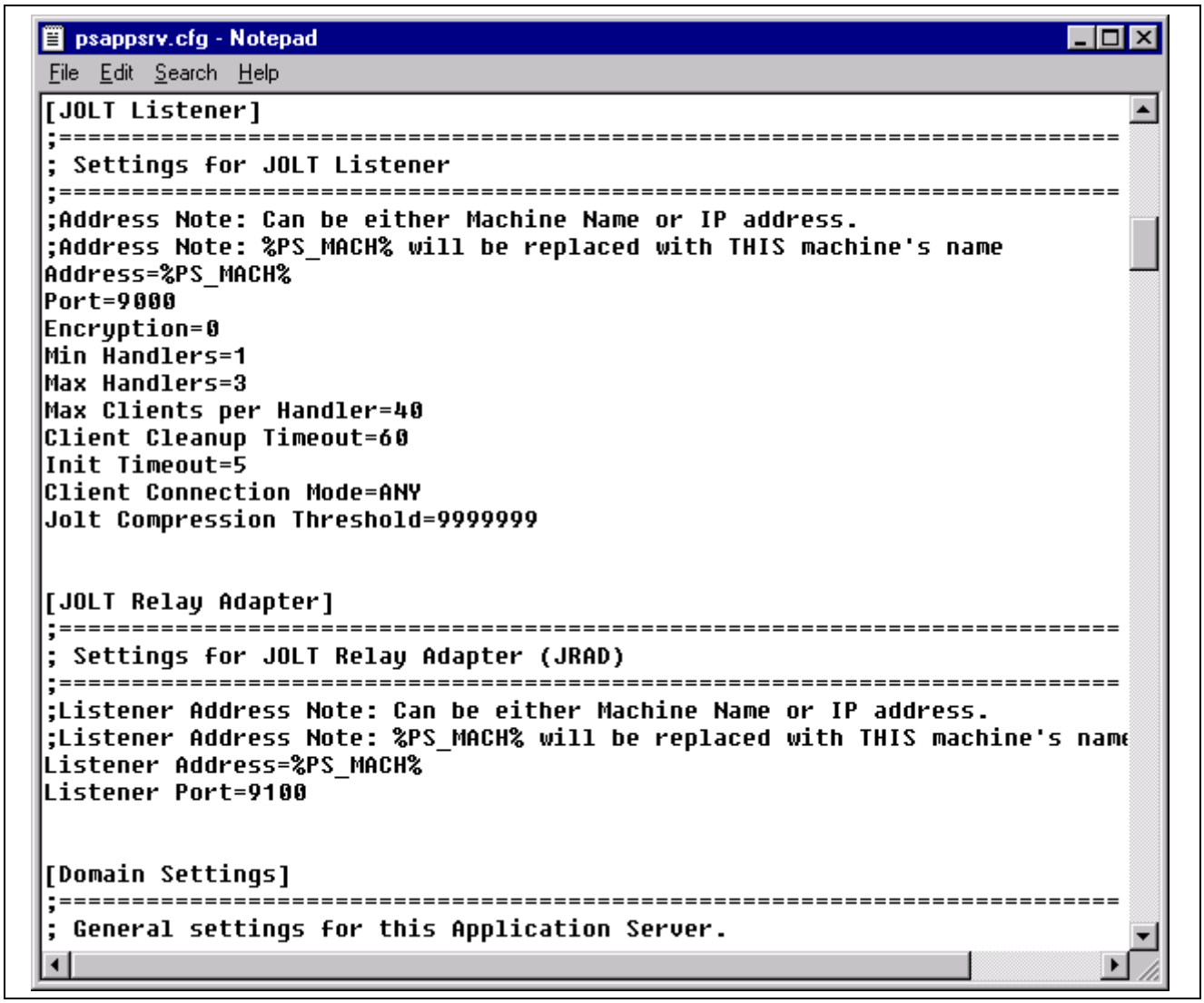
For example:

```
Command to execute (1-7, q) [q]: 5
Enter the operator ID : PTXYZ
```

Note. PeopleSoft secures the Structured Query Language (SQL) traces because, in some instances, the SQL that is traced may involve sensitive information.

Edit PSAPPSRV.CFG

The PSAPPSRV.CFG file contains all of the configuration settings for an application server domain. The PSADMIN interface provides prompts so that you can edit and modify this file within a structured format. In many cases, and perhaps due to personal preference, you may opt to edit the PSAPPSRV.CFG file manually. When editing this configuration file manually, note that it is similar to editing an INI file, because all of the parameters are grouped in sections.



PSAPPSRV.CFG file in a text editor

Edit APPSRV.LOG

This log file contains PeopleTools specific logging information.

Edit TUXLOG

The TUXLOG file enables you to trace the BEA Tuxedo component for troubleshooting information.

Edit PSAPPSRV.tracesql

You can specifically trace the activity of the PSAPPSRV server process by setting the PSAPPSRV.tracesql option.

Edit PSSAMSRV.tracesql

You can specifically trace the activity of the PSSAMSRV server process by setting the PSSAMSRV.tracesql option.

Creating a Domain

Use the Create a domain option to create a subdirectory under *PS_HOME*/appserv by using the domain name that the user specifies and to copy model files to that directory.

To create an application server domain:

1. Select *Create a domain* from the PeopleSoft Application Server Administration menu.
2. Enter the name of the domain that you want to create; the name must not exceed 8 characters.
3. Select a configuration template from the Configuration template list.

   The configuration templates are preconfigured sets of application server processes.

Note. If you are responsible for routinely creating many domains, you may want to either modify the CFX files to reflect your environment or create your own. You can manually edit any CFX file in the *PS_HOME*\appserv directory with any text editor, such as Notepad. To create your own CFX files, just save the CFX file to a new name after modifying the template values. The next time PSADMIN prompts you for a configuration template to create a domain, your new CFX file appears in the configuration templates list.

Deleting a Domain

Use the Delete a domain option to shut down the domain, if running, and delete the domain's subdirectory.

Note. Before you delete a domain, make sure that it is not running.

To delete a domain:

1. Select *Delete a domain* from the PeopleSoft Application Server Administration menu.
2. From the BEA Tuxedo domain list, select the number that corresponds to the domain that you want to delete.
3. When prompted to continue, enter y and press ENTER.

Using the Process Scheduler Menu

This section provides an overview of the Process Scheduler menu and discusses how to:

- Start a Process Scheduler server.
- Stop a Process Scheduler server.
- Configure a Process Scheduler server.
- Create a Process Scheduler server configuration.
- Delete a Process Scheduler server.
- Edit the Process Scheduler configuration file.
- Use the Process Scheduler options.
- Use Process Scheduler command-line options.

Understanding the Process Scheduler Menu

Use the PSADMIN utility to configure and administer PeopleSoft Process Scheduler. PeopleSoft Process Scheduler is used to run batch processes. You only need to configure PeopleSoft Process Scheduler on a server where you intend to run batch processes.

The following sections describe the menus and options within the PSADMIN utility that are related to PeopleSoft Process Scheduler in the order that they appear in the PeopleSoft Process Scheduler Administration menu—not in the order that you would access them the first time you configure the Process Scheduler server.

To access the PeopleSoft Process Scheduler Administration menu:

1. Select 2 from the PSADMIN menu.
2. Select the option from the PeopleSoft Process Scheduler Administration menu that corresponds to the action that you need to perform.

   The following sections explain the options for PeopleSoft Process Scheduler within PSADMIN. Those options that pertain to UNIX only are marked accordingly.

Starting a Process Scheduler Server

To start a Process Scheduler server:

1. Select 1 from the PeopleSoft Process Scheduler Administration menu.
2. To start the Process Scheduler server for a specific database, enter the number in the database list that corresponds to the appropriate database.

Stopping a Process Scheduler Server

You can stop a Process Scheduler server that is running on an application server by using PSADMIN or the Process Monitor.

To stop a Process Scheduler server:

1. Select 2 from the PeopleSoft Process Scheduler Administration menu.
2. To stop the Process Scheduler server for a specific database, enter the number from the database list that corresponds to the appropriate database.

Configuring a Process Scheduler Server

Configuring a Process Scheduler server is similar to configuring application servers and web servers. From the PeopleSoft Process Scheduler Administration menu, you invoke a text-driven interface that prompts you for parameter values. All of the Process Scheduler server configuration information for a specific database is contained in the PSPRCS.CFG file, and the PSADMIN provides an interface for and prompts you to edit the PSPRCS.CFG file.

Note. The PSPRCS.CFG file supports environment variables. For example, the TEMP setting in the Process Scheduler section can look like this: TEMP=%TEMP%.

For Microsoft Windows, although you edit the PSPRCS.CFG file through PSADMIN, you can find the PSPRCS.CFG file in the following directory: *PS_HOME*\APPSERV\PRCS*database_name*.

For UNIX, although you edit the PSPRCS.CFG file through PSADMIN, you can find the PSPRCS.CFG file in the following directory: *PS_HOME*/appserv/prcs/*database_name*.

To configure a Process Scheduler server by editing the PSPRCS.CFG file:

1. Select 3 from the PeopleSoft Process Scheduler Administration menu.
2. Select the number in the database list that corresponds to the server that you want to configure.
3. Specify the appropriate values for your site in the following configuration section prompts.

Creating a Process Scheduler Server Configuration

You must add or create a Process Scheduler server before you can configure it.

To add a Process Scheduler server configuration on the application server:

1. Select 4 from the PeopleSoft Process Scheduler Administration menu.
2. Enter the name of the database that the Process Scheduler server will access.
3. Enter *Y* to configure the Process Scheduler.
4. Update the settings as appropriate for your environment. For example, select *9* to change the UserID that the Process Scheduler uses to log on to the database.
5. When all of the settings are correct, select *4* to load the configuration.

Deleting a Process Scheduler Server

To delete a Process Scheduler server configuration:

1. Select 5 from the PeopleSoft Process Scheduler Administration menu.
2. Select the number in the database list that corresponds to the database to which the server has access.
3. Enter y when PSADMIN prompts you to continue.

Editing the Process Scheduler Configuration File

You can edit the Process Scheduler server configuration file manually instead of using the prompts in the PSADMIN interface to specify environment variables. This enables you to edit the configuration file in your preferred editor. You must set the EDITOR environment variable to point to the editor. For example:

```
set EDITOR=c:\apps\utils\kedit\keditw32.exe
```

Use this example if you use Notepad:

```
set EDITOR=c:\Windows\Notepad.exe
```

Note. When editing the PSPRCS.CFG file, make sure that there are no spaces between the equals sign and the entries. Also, make sure that there are no trailing spaces.

To manually edit the psprcs.cfg file:

1. Select 6 from the PeopleSoft Process Scheduler Administration menu.
2. Select the database that is associated with the file that you want to edit.
3. Enter the variables for the parameters that you need to specify.

Note. The system invokes the text editor that you have set as the EDITOR environment variable, such as Notepad or KEDIT, on the particular machine.

Using the Process Scheduler Options

You can have the Process Scheduler server run as a standalone component, or you can have the Process Scheduler server be controlled by BEA Tuxedo, which enables automatic restarts if the server goes down.

Using Process Scheduler Command-Line Options

You can bypass the PSADMIN menus to start and stop the Process Scheduler server.

Starting the Process Scheduler Server

To start the Process Scheduler server from the command line, enter the following:

```
psadmin -p start -d database_name
```

Stopping the Process Scheduler Server

To stop the Process Scheduler server from the command line, enter the following:

```
psadmin -p stop -d database_name
```

Setting Up the PeopleSoft Windows Service

This section provides an overview of Microsoft Windows services and discusses how to:

- Configure the PeopleSoft service.
- Monitor the executables.
- Administer PeopleSoft services.
- Edit the PSNTSRV.CFG file manually.

Note. This section applies only to Microsoft Windows servers. It involves setting up both the application server and Process Scheduler server agent as PeopleSoft Windows services. There is no equivalent feature for UNIX servers.

Understanding Microsoft Windows Services

A Microsoft Windows service is a Microsoft-standard package that automatically starts and stops a process when you boot or shut down the system. You can also start and stop Microsoft Windows services manually through the Service Control Manager (SCM), which you can access through the Control Panel. A service uses a standard application programming interface (API) so that it can interact with the Control Panel and log messages to the standard event log.

For PeopleSoft, the service starts in an environment that is separate from any users who are signed in to the system (or to the machine). This means that administrators no longer need to log on to a machine, start the command prompt, and enter the proper commands to start the server process. In addition, if you use the PeopleSoft service, an administrator's logon session does not need to remain open while the Process Scheduler server or the application server runs.

If you have multiple application server domains and Process Scheduler servers on the same machine, you can start them all by using the same service setup.

Note. The PeopleSoft service supersedes the method that is provided in the Microsoft Windows resource kit. Do not use SRVANY.EXE or AT commands to start the Process Scheduler or the application server.

You can start application server domains and Process Scheduler servers as Microsoft Windows services. The PeopleSoft service, if configured, automatically starts the application server or Process Scheduler server when you boot the server machine. This means that administrators do not need to manually boot each application server or Process Scheduler server after you reboot a Microsoft Windows server.

Note. When you configure a domain to run as a Windows service, you must set the JavaVM option to *–Xrs* in the PSAPPSRV.CFG or PSPRCS.CFG file.

Configuring the PeopleSoft Service

The following procedure assumes that you have already installed and configured an application server domain or Process Scheduler server agent on the Microsoft Windows server.

After completing this procedure, the specified application server domains or Process Scheduler servers start and shut down automatically when the operating system recycles.

To set up the Microsoft Windows service for an application server or Process Scheduler server:

1. Open the System utility within the Control Panel, and set the following variables on the Environment tab:

Variable	Value
TEMP	Specify the location of the TEMP directory on the Microsoft Windows server, as in C:\TEMP.
TUXDIR	Specify the location of the BEA Tuxedo directory on the Microsoft Windows server, as in C:\bea\tuxedo81.

   These settings must appear in the System Variables section.

2. Run the PSADMIN utility, and select Service Setup from the main menu.
3. Select Configure a Service from the PeopleSoft Services Administration menu.
4. Enter *y* to indicate that you want to change configuration values.
5. Enter the names of the application server domains and the Process Scheduler databases that you want to include as part of the Microsoft Windows service.

   To add multiple domains or databases, delimit each value with a comma and a space.

   Note. The Windows Services section of PSADMIN modifies the psntsrv.cfg file in the *PS_HOME*\appserv directory. You can edit this file manually by selecting Edit a Service Configuration File from the PeopleSoft Services Administration menu.

6. Select Install a Service from the PeopleSoft Services Administration menu.

   Note. All of the domains and databases that you specified are now part of a single Windows service called PeopleSoft *PS_HOME*, where *PS_HOME* is the root directory of the PeopleSoft system from which you ran PSADMIN (for example, *PeopleSoft C:\pt846*).

7. Return to the Windows Control Panel, and start the Services utility.
8. In the Services dialog box, scroll to find the entry that adheres to the PeopleSoft *PS_HOME* naming convention and double-click it to access its properties.

   Note. The default startup mode is *Manual*.

9. On the General tab of the service properties, select a startup type of *Automatic*.
10. On the Log On tab, the Log On As setting must match the Log On As setting that's defined for the BEA ProcMGR V 8.1 service, which was created when you installed BEA Tuxedo.

    Both services should either be configured to *Log On As Local System Account*, or to *Log On As This Account* (referring to the same account).

    Important! If the PeopleSoft service includes a Process Scheduler server agent, you must select *Log On As This Account* for both the PeopleSoft service and the BEA ProcMGR V 8.1 service, to prevent problems when running Crystal Reports.

11. On the General tab of the service properties, click Start.

    The application and Process Scheduler servers are now running, and will start automatically whenever you boot the server.

Service Start Failure

It's possible that one or more of the domains or databases that are configured as part of the PeopleSoft *PS_HOME* service will fail to start, for reasons unrelated to the service.

The service is marked as started even if only one of its assigned domains or databases starts. A message is written to the Windows event log for each domain and database, indicating whether it has started or not. If you experience problems with any domain or database, check the event log to see if it started successfully.

If all of the assigned domains and databases fail to start, the service is marked as stopped, and the following message is written to the event log:

```
Unable to start any of the domains configured for service service_name.
```

Monitoring the Executables

To test the Microsoft Windows service, reboot the server, and make sure that the appropriate server executables are running.

For the application server, use the Microsoft Windows Task Manager or the Server status option from the Domain status menu to see that the following executables are running:

- PSAPPSRV.EXE
- PSSAMSRV.EXE
- BBL.EXE
- WSL.EXE

Also make sure that any additional server processes that you have configured, such as PSQCKSRV.EXE, are running.

For PeopleSoft Process Scheduler, use the Microsoft Windows Task Manager or the Process Monitor to make sure that PTPURCS.EXE is running. If you've changed the name of PTPURCS.EXE, look for your name instead.

Administering PeopleSoft Services

you can specify three options that are related to the PeopleSoft service setup by using PSADMIN or by editing the PSNTSRV.CFG file manually.

The following sections describe each parameter.

Service Start Delay

When an application server or Process Scheduler server resides on the same machine as the database server, consider using the Service Start Delay setting. By using this feature, you can avoid the situation where the database server is booting and is not ready to process requests at the time that the service attempts to boot the application server domain or Process Scheduler server. In this scenario, without a delay set, the connection fails.

You can configure a Service Start Delay parameter in the PSNTSRV configuration file that specifies a delay, in seconds, that elapses before a service attempts to start any application server domains or Process Scheduler servers. This allows the RDBMS enough time to boot and become available to accept requests.

The default is 60 seconds.

Application Server Domains

Specify the names of the domains that you want to start automatically when you boot the application server machine.

If you specify multiple domains, separate each domain with a comma and a space.

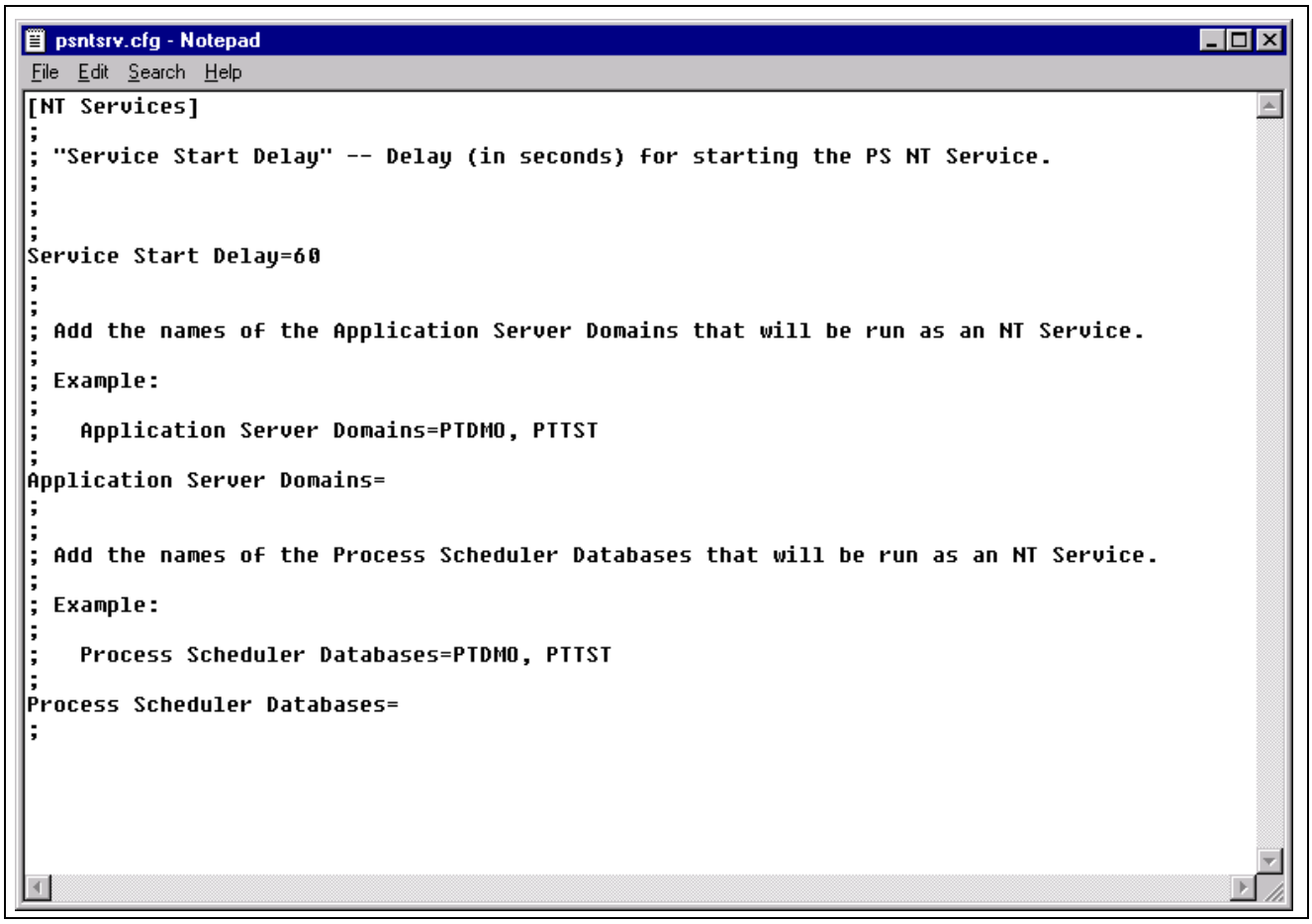
Process Scheduler Databases

Enter the databases with which a Process Scheduler server is associated. For each database that you specify, the associated Process Scheduler server starts when you boot the Microsoft Windows server.

If you specify multiple databases, separate each database with a comma and a space.

Editing the PSNTSRV.CFG File Manually

You can edit the file directly by selecting 4 (Edit a Service Configuration File) from the main menu. This opens the PSNTSRV.CFG file in a text editor, where you can enter and save your changes.



PSNTSRV.CFG file

CHAPTER 4

Setting Application Server Domain Parameters

This chapter describes all of the configuration options that are related to an application server domain. Generally, the documentation reflects the order in which the configuration sections appear in the PSADMIN interface or the PSAPPSRV.CFG file.

This chapter discusses:

- Startup options.
- Database options.
- Security options.
- Workstation listener options.
- BEA Jolt listener options.
- BEA Jolt relay adapter options.
- Domain settings.
- PeopleCode Debugger options.
- Trace options.
- Cache settings.
- Remote call options.
- PSAPPSRV options.
- PSANALYTICSRV options.
- PSSAMSRV options.
- PSQCKSRV options.
- PSQRYSRV options.
- Messaging server processes.
- Simple Mail Transfer Protocol (SMTP) settings.
- Interface driver options.
- PSTOOLS options.
- PeopleSoft Integration Broker options.
- Search indexes.
- PSRENSRV options.
- PSPPMSRV options.
- Select server process options.

Note. The application server dynamically *spawns* server processes according to the volume of transaction requests. There is no explicit parameter that you must set to enable spawning. In the following configuration section descriptions, some servers enable you to specify a minimum and maximum number of server processes. To enable spawning, the maximum value must exceed the minimum value by an increment of at least one. As needed, the application server spawns server processes up to the maximum value. As the volume of transactions decreases, the number of spawned server processes decreases, or decays, until the minimum value is reached.

Startup Options

Set database sign-in values in the Startup section.

DBName

Enter the PeopleSoft database name, such as FSDMO80 or HRDMO80. This parameter is case sensitive.

DBType

Enter the PeopleSoft database type, such as DB2ODBC, DB2UNIX, INFORMIX, MICROSFT, ORACLE, or SYBASE. If you enter an invalid database type, PSADMIN prompts you with a valid list.

UserID

Enter the PeopleSoft user ID that is authorized to start the application server. Use Maintain Security to add this property to a permission list, which is applied to the user profile by way of a role. The Can Start Application Server permission must be set in the permission list. For the application server to boot, the appropriate user ID with the correct authorizations must be assigned to this parameter. This is the ID that the application server passes to the database for authentication and connection. The user ID that you enter here is not related to the actual user (administrator) that carries out the boot command.

The authorization to start an application server does not (directly or indirectly) grant any authorizations or privileges beyond the ability to start the application server. Each user who attempts to sign in enters a unique user ID and password, which the application server uses to authenticate each user.

UserPswd

Enter the password that is used by the specified user ID that will gain access to the database. The value that you enter must be specified in uppercase to simplify administration of the system.

Connect ID

Required for all database platforms. Enter the database-level ID that the PeopleSoft system uses to make the initial connection to the database. This user name must have authority to select from PSACCESPRFL, PSLOCK, PSOPRDEFN, and PSSTATUS.

Connect Password

Enter the password for the connect ID. For instance, this might be the UNIX user's password (either uppercase or lowercase).

ServerName

Required for Sybase and Informix. Enter the name of the server on which the PeopleSoft database is installed. This value is case sensitive.

Database Options

Use the Database Options section to specify environment variables that may improve the performance of the system. These options do not apply to every database.

SybasePacketSize

Enter a Transmission Control Protocol (TCP) packet size. The minimum value is 512 and the maximum value is 65538. The default packet size is 512. If you change the packet size, make the corresponding changes to the Sybase database server.

See Your Sybase documentation.

UseLocalOracleDB

Use this option to enable a batch program to initiate a local connection to a PeopleSoft database that is running on the same machine. You should use this option for all PeopleSoft Process Scheduler (batch) and application server configurations that are local (on the same server) to the PeopleSoft Oracle instance. This type of connection enables batch processes to complete significantly quicker. Enter 1 to enable this option, and enter 0 to disable it.

Note. Using the local Oracle connection disables the Query Kill function.

EnableDBMonitoring

Required for database-level auditing. How this works varies slightly, depending on the platform. Use this option to view more information regarding the clients that are connected to a database server through the application server. For instance, with this enabled, you can view the client machine name or user ID that is associated with a particular connection. Without this option enabled, all connections appear somewhat anonymously, as in PSFT or APPSERV.

The default value is *1* (enabled). Enter *0* to disable it.

Note. This parameter isn't supported on Informix or DB2 LUW platforms.

OracleDisableFirstRowsHint

This parameter enables or disables the inclusion of the Oracle FIRST_ROWS hint on PeopleSoft search pages that use the %First_Rows meta-SQL variable.

For certain SQL constructs, the use of the FIRST_ROWS hint can severely degrade performance to an unacceptable degree. This performance reduction occurs primarily on search pages that are ultimately based on views with multi-table joins.

Specify a value of *1* to apply this parameter and disable the FIRST_ROWS hint. SQL statements generated by search page PeopleCode that includes the %First_Rows meta-SQL variable won't contain FIRST_ROWS.

The default value of this parameter is *0*, which enables SQL containing the FIRST_ROWS hint to be generated.

Security Options

Use the Security section to set an additional layer to the sign-in process.

Validate Signon With Database

Use this option to set an additional level of authorization-checking to be performed at the database level. Enter 1 to enable this option, and enter 0 to disable it.

With this option disabled, if a PeopleSoft user attempts to connect to an application server, the application server ensures that the user's PeopleSoft user ID and password exist on PSOPRDEFN. If it does not exist, the request to connect fails. This is PeopleTools-level authentication.

With this option enabled, the application server first attempts to connect to the database by using the user ID and password as part of the database connection string. If the authorization is successful, it disconnects, and then the normal PeopleSoft sign-in procedure occurs.

With this option enabled, to connect successfully to the database, the user must be defined on either the operating system or the database and within PeopleSoft.

Note. For DB2 z/OS (MVS), the user ID and password must be defined as z/OS user logon IDs.

Workstation Listener Options

The workstation listener is the component to which PeopleSoft native Windows clients send BEA Tuxedo messages.

Address

%PS_MACH% resolves automatically to the machine name that PSADMIN obtains by using a system application programming interface (API) call. You can also specify the machine's Internet Protocol (IP) address (dotted notation) or its resolvable name (domain name server [DNS] name).

You should not change this value except in the following rare cases. If you are configuring files to run an application server on another machine (that is, you plan to copy PSAPPSRV.CFG and PSAPPSRV.UBB to a domain on another machine), you must overlay %PS_MACH% with the other machine's name.

Port

Enter the 4-digit port number to assign to the WSL. Port numbers are arbitrary numbers between 1000 and 64 K and must not already be in use by another service. The default value is 7000.

Encryption

Use this option to enable the encryption of data messages between client workstations and the application server. Specify one of the following values:

- *0* — No encryption.

Important! This is the default value.

- *40* — 40-bit encryption.
- *128* — 128-bit encryption.

Note. Because this is a dynamic parameter, you must modify it by selecting Custom Configuration on the Quick-Configure menu, and reboot the application server domain for it to take effect.

Min Handlers

Enter the number of workstation handlers (WSHs) to be started at boot time. The default for small and large application server configuration templates are 1 and 10, respectively.

Max Handlers

Enter the maximum number of WSHs that can be started for a domain. If the Min Handlers value equals the Max Handlers value, BEA Tuxedo does not automatically spawn incremental WSHs.

Max Clients per Handler

Enter the maximum number of client workstation connections that each WSH can manage. Each WSH allows up to around 60 client connections. Numbers vary depending upon the resources of the server. In most cases, you should decrease the default as opposed to increasing it. The default is 40.

Client Cleanup Timeout

Enter the amount of time, in minutes, that a client connection can remain idle (no work requested) before BEA Tuxedo terminates the client connection. Client disconnects are transparent to a client, and a user just clicks the mouse to cause a reconnection. The default value for this setting is *60* minutes.

Init Timeout

This value, when multiplied by SCANUNIT (a UBB parameter value that is defined in the PSAPPSRV.UBB file) specifies the amount of time, in seconds, that BEA Tuxedo allows for a client connection request to bind to a WSH before terminating the connection attempt.

Tuxedo Compression

Enter the minimum length of a data message for which the application server initiates data compression. While compression results in favorable performance gains for transactions over a wide area network (WAN), testing reveals that compression can degrade performance slightly over a local area network (LAN) due to the compression and decompression overhead.

You should use the default threshold of 5000, which sets a balance between WAN and LAN environments. This means that only network request and response messages over 5000 bytes are compressed, and those 5000 and under are uncompressed. If you support both WAN and LAN users, you can configure a hybrid environment by configuring two application servers: one to support WAN users (with compression set to 100) and another to support LAN users (with compression set to 100000, effectively turning compression off).

BEA Jolt Listener Options

Use this section to enable PeopleSoft Internet Architecture connections. The BEA Jolt listener enables BEA Tuxedo to exchange messages with the web server.

Address

See the equivalent parameter for the workstation listener.

Port

Enter the port number that is used for the BEA Jolt server listener (JSL). This value can be any port number that is not already in use by another service on the machine that runs the application server domain. The port number is not used unless you answer Yes to the prompt that asks whether you want to start BEA Jolt.

Encryption

Use this option to enable the encryption of data messages between client workstations and the web server. Specify one of the following values:

- *0* — No encryption.

  Important! This is the default value. Incoming Jolt requests from the web server (portal, PIA, and Integration Broker) are not encrypted.

- *40* — 40-bit encryption.
- *128* — 128-bit encryption.

Note. Because this is a dynamic parameter, you must modify it by selecting Custom Configuration on the Quick-Configure menu, and reboot the application server domain for it to take effect.

Min Handlers

Enter the number of BEA Jolt server handlers (JSH) to be started at boot time. Each JSH multiplexes up to 50 connections.

Max Handlers

Enter the maximum number of JSHs.

Note. JSHs spawn by using successive port numbers starting at the port number for the JSL in the PSAPPSRV.CFG file. Make sure that the additional ports are free before configuring spawning.

Max Clients per Handler

Enter the maximum number of client connections that each JSH can manage.

Client Cleanup Timeout

Enter the amount of time, in minutes, that a client connection can remain idle (no work requested) before BEA Tuxedo terminates the client connection. Client disconnects are transparent to a client, and a user just clicks the mouse to cause a reconnection. The default value for this setting is *10* minutes.

Init Timeout

See the equivalent parameter for the workstation listener.

Client Connection Mode

Enter one of these options to control the allowed connection modes from clients:

- RETAINED: The network connection is retained for the full duration of a session.
- RECONNECT: The client establishes and brings down a connection when an idle timeout is reached and reconnects for multiple requests within a session. The reconnection is transparent to the user.
- ANY: (Default) The server allows client code to request either a RETAINED or RECONNECT type of connection for a session. Whereas, with the other two options, the server dictates from which type of client it accepts connections. This option translates to the -c Connection Mode parameter for the JSL section in the PSAPPSRV.UBB file.

Jolt Compression Threshold

BEA Jolt compression can significantly improve performance. BEA Jolt compression enables messages that are transmitted through a BEA Jolt connection to be compressed as they flow over the network. You are likely to see the most significant performance improvements over a WAN.

For compression, the configuration files contain a default compression threshold. This default value should provide the best results for most situations. However, your application server administrator can adjust this value to suit your implementation.

The compression threshold indicates to the server how large a packet must be to require compressing. In other words, the value that you set is the minimum number of bytes that a single packet must be before the server compresses it.

Many of the XML messages being sent around the system are greater than 100,000 bytes. These messages contain HTML in compressed states, so it’s generally not required that these messages be compressed. Because of this, the PeopleSoft default is set to 1,000,000 bytes.

Be careful when adjusting compression settings. If you set the threshold too high, then no packets will be large enough to be compressed. If you set the threshold too low, you may greatly reduce network traffic, but be aware that the server will have an increased workload from compressing numerous packets. Typically, you should decrease the threshold according to the bandwidth of the workstation hardware as described in the following paragraphs.

If you are handling only LAN connections, you may want to disable compression by setting the threshold to 99999999 so that only packets larger than 99,999,999 bytes are compressed. Of course, such a large value effectively disables compression so that no packets are compressed. This means no extra work for the server compressing packets.

Alternatively, if you have mostly low bandwidth, as in 56-kilobyte (KB) modem connections over a WAN, then you most likely want to compress the packets as much as possible. When decreasing the compression threshold, keep in mind that the law of diminishing returns applies. Setting the threshold much below 1000 puts an increasing load on the server, and this can nullify any performance increases that you may have gained from reduced network traffic.

Additional Prompt

After you finish all of the configuration sections, PSADMIN prompts you to configure BEA Jolt which is on by default.

If you are using the PeopleSoft Internet Architecture, you must configure BEA Jolt

BEA Jolt Relay Adapter Options

The BEA Jolt relay adapter is disabled by default. Unless you have a specific need for JRAD, you should skip this section.

Listener Address

The default is %PS_MACH%. Enter the machine on which the application server is running. See the equivalent parameter for the workstation listener.

Listener Port

This option is for advanced configurations requiring the BEA Jolt internet relay (JRLY). The listener port listens for JRLY requests and must match the JRLY "OUT" port setting in the JRLY configuration file of the sending machine. The port number, as in 9100, is not used unless you enter y at the prompt that asks if you want to configure JRAD.

Domain Settings

Use this section to specify general settings for the entire domain—not just for a specific component of the domain.

Domain ID

Enter the name of the application server domain. It does not need to match the name that you specified when you created the domain. This name is important only because the BEA Tuxedo Web Monitor and PeopleSoft Watch Server (PSWATCHSRV) use it to identify application server domains and the processes associated with each machine. It should not exceed 8 characters. Generally, you should use the database name in lowercase.

Add to PATH

Enter the directory that contains your database connectivity software, as in /apps/db/oracle/bin, in the path. If the database connectivity directory is not already specified in the path, you can set it by specifying this parameter. The value is added to the path.

On Microsoft Windows, if you don't enter a value, it uses the current path.

On UNIX, if you don't enter a value, it uses the current directory—not the current path. To have it set by default to the current path, enter a period (.).

Note. On Windows, entries that contain a space must be surrounded by quotes.

Spawn Threshold

Enter a parameter that's supplied to BEA Tuxedo for control of process spawning by using the -p command-line option for all server processes. The default setting (*1,600:1,1*) rarely needs to be changed.

This setting enables the dynamic decay of spawned server processes as the transaction volume decreases. The value can be loosely translated to mean that if, in 600 seconds, there is less than or equal to one job in the queue, the decay process begins. This is described in more detail in the timeout settings appendix of this PeopleBook.

For more information, see servopts(s) in the reference manual of the BEA TUXEDO online documentation.

Note. This parameter applies only if, for PSAPPSRV, the value of *Max Instances* is greater than that of *Min Instances*.

See Also

Appendix B, "PeopleSoft Timeout Settings," Application Server Timeouts, page 325

Restartable

Enter y to have BEA Tuxedo restart server processes (except the BBL process) if the server dies abnormally, as in a kill on UNIX or through the Task Manager on Microsoft Windows. Otherwise, enter n.

Allow Dynamic Changes

Often, administrators must set a trace or performance parameter while the domain is up and running. If you enable this option, then you don't need to reboot the domain for the modified parameter value to take effect.

Enter y or no to enable or disable dynamic changes. When disabled, you must reboot (or cycle the processes) for changes to take effect.

When enabled, the server checks an internal time stamp for a particular service request to see if any values have changed for the parameters for which dynamic changes are valid. If values have changed, the system uses the modified parameter value.

You should enable this option in your test and development domains. For production environments, you should enable dynamic changes selectively.

Only these parameters allow dynamic changes:

- Recycle Count.
- Consecutive service failures.
- Trace SQL and Trace SQL Mask.
- Trace PC and Trace PC Mask.
- Trace PPR and Trace PPR Mask.
- Log Fence.

- Enable DB Monitoring.
- Enable Debugging.
- Dump Memory Image at Crash.
- Dump Managed Objects at Crash.
- Log Error Report.
- Mail Error Report.
- SMTP Settings (all except SMTPGuaranteed, SMTPTrace, and SMTPSendTime).

Note. The parameters that allow dynamic changes are also identified through comments in the PSAPPSRV.CFG file. Look for the phrase "Dynamic changes allowed for X," where *X* is the parameter name. This option does not apply to configuration parameters that BEA Tuxedo relies on, such as the number of processes, whether restart is enabled, the port numbers, the amount of handlers, and so on.

LogFence

Enter a level of network tracing, ranging from –100 (suppressing) to 5 (all). The default is 3.

The trace file is generated in *PS_HOME*\appserv*domain*\LOGS\psappsrv.log.

AppLogFence

This setting is not available through the PSADMIN interface, but can be entered directly into the PSAPPSRV.CFG file.

You can use this parameter to conditionally determine whether you want to do certain logging from your application. You can implement this parameter from PeopleCode using the %AppLogFence system variable, and it's more fully documented in the PeopleCode Developer's Guide.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleCode Developer's Guide*, "Debugging Your Application," Using Application Logging.

Trace-Log File Character Set

Enter the character set (ANSI or UNICODE) of the machine to which you typically write and read the traces and log files. If the character sets are not matched between the file and the machine, the file is unreadable.

PeopleCode Debugger Options

Use this section to enable and configure the PeopleCode debugging environment. Configuring PeopleCode debugging is discussed in detail in another section of this PeopleBook.

See Also

Chapter 11, "Configuring Trace and Debug Settings," Setting Up the PeopleCode Debugger, page 257

Trace Options

This section enables you to specify the tracing options that you can enable on the application server to track the Structured Query Language (SQL) and PeopleCode of the domains. You can also set all of the trace parameters from the PeopleSoft sign-in page. Just beneath the Sign In button, click the link that opens the trace flags page. This enables you to set the trace options and then sign in to the system.

TraceSQL

Enter the logging level for SQL tracing for all clients. Traces are written to *PS_HOME*/appserv/*domain*/LOGS/*domain_user_ID_servername*.tracesql. See TraceSQLMask for trace options.

Enter 0 to disable tracing; enter 7 to enable a modest tracing level for debugging. For other levels of tracing, set this option to a value that equals the sum of the needed options. For example, to trace only SQL, enter 1; to trace SQL statements and connect statements enter 7 (1+ 2 + 4 = 7). A setting of 7 is recommended for troubleshooting connection and other basic problems. Tracing can consume large amounts of disk space over time, so be sure to reset this option to 0 when you finish troubleshooting.

Important! The trace file stores elapsed times for SQL events to a precision of one microsecond (six decimal places). However, due to limitations of the operating system, Windows precision is actually in milliseconds (three decimal places), so the last three digits in a Windows trace will always be zero. Elapsed times in UNIX are accurate to one microsecond.

TraceSQLMask

Enter the logging level ceiling for SQL tracing for individual clients. Traces are written to *PS_HOME*/appserv/*domain*/LOGS/*client_user_ID_servername*.tracesql. Clients must specify the necessary SQL tracing level by using the PeopleSoft Configuration Manager on the Trace tab. To prevent clients from turning on the application server trace and consuming resources, the application server uses TraceSQLMask as an administrative control facility.

If a client transmits a request to trace SQL, the application server compares the value that is transmitted to the TraceSQLMask value. If the client value is less than or equal to the TraceSQLMask value, the application server enables the trace. However, if the client value is greater, the application server enables the trace up to the TraceSQLMask value. Trace files are written on the application server; no trace shows up on the client workstation.

Trace values are set in the PSAPPSRV.CFG file. Output files are written to *PS_HOME*/appserver/winx86/*domain*/logs.

TracePC

Enter a level for PeopleCode tracing for activity that is generated by all clients on a domain. Eligible values are defined in the configuration file. TracePC values are displayed in the PeopleSoft Configuration Manager on the Trace tab. You can find the results in *PS_HOME*/appserv/*domain*/LOGS/*domain*.log.

Important! The trace file stores elapsed times for PeopleCode events to a precision of one microsecond (six decimal places). However, due to limitations of the operating system, Windows precision is actually in milliseconds (three decimal places), so the last three digits in a Windows trace will always be zero. Elapsed times in UNIX are accurate to one microsecond.

TracePCMask

Enter which PeopleCode trace options that are requested by client machines will be written to the trace file. You can find the results in *PS_HOME*/appserv/*domain*/LOGS/*client_machine.domain*.log.

TracePPR and TracePPRMask

Use these options to trace the activity in the page processor. Typically, these options are used internally only by PeopleSoft developers; however, you may need to view the results of this trace when troubleshooting.

Tracing-related display processing is useful for seeing when and if related displays are being updated and if they are updating successfully. Some sample output in the log file from setting this flag includes:

```
Starting Related Display processing
Related Display processing - PPR_RELDSPLVALID not set
Related Display processing - All Rows
  Starting Related Display processing for - PSACLMENU_VW2.MENUNAME
  Related Display processing for - PSACLMENU_VW2.MENUNAME - completed successfully
Finished Related Display processing
```

By using the keylist generation tracing in addition to the related display tracing, you can determine why the related displays have the wrong value. It shows where the keys are coming from. The following is a sample of keylist generation tracing:

```
 Starting Keylist generation
       Keylist generation - FIELDVALUE is a key
       FIELDVALUE is low key
       Low key value was supplied =
       Key FIELDVALUE =
       Keylist generation - FIELDNAME is a key
       Keylist generation - Finding value for USRXLATTABLE_VW.FIELDNAME
         Not Found in key buffer
           Seaching for field FIELDNAME in component buffers
           Scanning level 1
             Scanning record DERIVED_USROPTN for field FIELDNAME
             Field FIELDNAME found in record DERIVED_USROPTN
         Found in component buffers, value = PT_TIME_FORMAT
       Key FIELDNAME = PT_TIME_FORMAT
       Keylist generation - USEROPTN is a key
       Keylist generation - Finding value for USRXLATTABLE_VW.USEROPTN
         Not Found in key buffer
           Seaching for field USEROPTN in component buffers
           Scanning level 1
             Scanning record DERIVED_USROPTN for field USEROPTN
             Scanning record PSUSROPTLIST_VW for field USEROPTN
             Field USEROPTN found in record PSUSROPTLIST_VW
         Found in component buffers, value = TFRMT
       Key USEROPTN = TFRMT
     Keylist Generation complete
     FIELDNAME = PT_TIME_FORMAT
     FIELDVALUE =
     USEROPTN = TFRMT
```

In this example, you can see how the system builds the keylist by first searching in the current record (key buffer), then searching the buffers in the current level, and then searching up a level, and so on. It also indicates exactly what record the key value is taken from. This is useful on complex components where there are often several instances of a particular field; a common problem is that the value is derived from an unexpected location.

Combining the keylist tracing and the related display tracing provides a good view of the system behavior. For example:

```
Starting Related Display processing
Related Display processing - All Rows
  Starting Related Display processing for - PSACLMENU_VW2.MENUNAME
    Starting Keylist generation
      Keylist generation - MENUNAME is a key
      MENUNAME is low key
      Low key value was supplied = APPLICATION_ENGINE
      Key MENUNAME = APPLICATION_ENGINE
    Keylist Generation complete
     MENUNAME = APPLICATION_ENGINE
  Related Display processing for - PSACLMENU_VW2.MENUNAME - completed successfully
```

Each related display goes through the keylist generation process, and you can see exactly what key values are used to populate the related displays and where those key values came from.

The work record flag is a performance feature. If every field in a level-0 record has a value from the keylist and is display-only, then it is marked as a work record because the values can't be changed. After it is marked as a work record, that affects how the record behaves. For example, PeopleCode for fields in the record but not in the component don't run, data isn't saved, and so on. By enabling this tracing option, you can see which records are flagged as work records. The output looks like this:

```
Work flag cleared for record PSCLASSDEFN_SRC
 Work flag cleared for record PSCLASSDEFN_SRC
 Work flag cleared for record PSCLASSDEFN
 Work flag cleared for record PSPRCSPRFL
 Work flag cleared for record SCRTY_QUERY
 Work flag set for record PSCLASSDEFN
 Work flag set for record PSPRCSPRFL
 Work flag set for record SCRTY_QUERY
```

Because the flag is turned on and off at various points, the last setting (set or cleared) is the most important. In the previous trace, PSCLASSDEFN, which is marked as a work record, is cleared and then set again.

TraceAE

Use this parameter to activate specific PeopleSoft Application Engine traces for tracing Application Engine programs.

TraceOpt and TraceOptMask

The bits enable logging for Optimization Engine components beyond the standard LogFence setting. For example, a value of 3510 sets full trace on all components.

TracePPM

The Performance Monitor agent is a thread that reports performance metrics for each instrumented server if monitoring is enabled for the database. Select *1* to enable and *0* to disable.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Performance Monitor*

DumpMemoryImageAtCrash

This parameter determines whether or not a memory image of the failing process is created when a crash occurs. By default, the value is 'NONE' which means that a memory image will not be written during a crash. This value can be set to 'MINI' or 'FULL'. MINI means that a shorter memory image is written. This may be a better option if you are leaving this option turned on permanently. FULL may be appropriate when you are debugging a known issue. Typically, this option is used internally only by PeopleSoft developers.

DumpMemoryObjectsAtCrash

This parameter assists PeopleSoft in debugging any crash problems at your site by providing insight into the customized object definitions to reproduce the crash on another database.

Log Error Report, Mail Error Report

If you enter y (enabled) and runtime errors are detected (nonfatal error conditions), the system writes a message and information regarding the runtime error to the current log file. If you assign the MailErrorReport parameter an email address, an individual, such as a system administrator, can be alerted whenever the system writes an error to the log. If MailErrorReport is enabled but LogErrorReport is set to n, the system still sends a message for application server crashes, which always cause data to be written to the log. The following is an example of setting this parameter to send notifications to an email address: MailErrorReport=tom.sawyer@bigcompany.com.

Write Crash Dump to Separate File

If the application server shuts down abnormally, you can view the log information that is related to the shutdown. However, because this information can be lengthy, this option enables you to write the information to a file other than the appserv.log file. To enable this option, enter y.

The system writes the crash dump file to *PS_HOME*\appserv*domain*\logs. The system names the crash dump file according to the following convention: *server_process_name*.*process_ID*.dmp.

The following example shows what appears in the appserv.log in the event of a crash:

```
(0) Unhandled exception occurred. Writing crash dump to PSAPPSRV.213.dmp
(3) Switching to new log file b:\appserv\test\logs\PSAPPSRV.213.dmp
```

To disable this option, enter n. If you do not enable this option, crash information appears in the appserv.log by default.

Cache Settings

Use this section to specify how to handle caching at your site. Enabling caching on the application server improves performance.

Cache Settings

This section has no configuration parameters that need adjusting. In the PSAPPSRV.CFG, you notice that the settings for this section have been commented out. These settings should only be re-introduced pending recommendation from PeopleSoft Support or Engineering.

EnableServerCaching

With EnableServerCaching, you specify what objects the system stores in cache on the application server. To enable application server disk caching the value must be set to 1 or 2.

If you enter 1 the system caches only the most used classes of objects, and if you enter 2, the system caches all object types regardless of the frequency of use. Which option you select depends on internal testing at your site.

To disable application server caching, set this value to 0. In most cases there is no reason to disable server caching. Doing so significantly degrades performance, because it requires the application server to retrieve an object from the database each time the system needs it.

ServerCacheMode

If server caching is enabled on the application server, which is usually the case, there are two modes of caching from which to choose: shared and nonshared cache files.

If you use the nonshared cache mode, each PSAPPSRV server process that starts within a domain maintains its own separate cache file. In this mode, there is one cache file per PSAPPSRV server process.

To set one cache directory and file per server process, enter 0 at the Set ServerCacheMode prompt. By default, nonshared cache files are enabled. With this option enabled, you can find cache files in *PS_HOME*\appserv*domain*\cache*n-1**n*.

In the preceding path, *n* refers to the number of PSAPPSRV server processes that are configured to start within the domain. For example, if you have two PSAPPSRV processes, the system creates two cache directories, \1 and \2, beneath the cache directory.

To set shared caching for the domain, enter 1 at the Set ServerCacheMode prompt. With this option enabled, you can find the cache files in *PS_HOME*\appserv*domain*\cache\share.

The system assumes that a preloaded cache exists in the share directory. The preloaded cache contains most instances of the managed object types that are cached to file. When you boot the application server, if shared cache files are enabled but no cache files exist, the system reverts to unshared caching.

CacheBaseDir

This setting is the location where cache files will be written and stored for this domain.

Note. You must preload your shared cache before you enable shared caching on the application server.

Application Engine processes are independent from application server domains, directories, and configuration files. Therefore, Application Engine processes do not share cache with application server domain processes.

MaxCacheMemory

PeopleTools stores application data in a memory cache to increase system performance. However, too large a cache can leave insufficient available memory on your system, which leads to reduced performance.

Use this setting to specify the maximum size of the memory cache. Every time you use an object, its LastUsedDate value is updated. When your system reaches the memory cache threshhold, the system prunes the oldest objects in the cache first — that is, the ones with the oldest LastUsedDate values — and places the pruned data in a disk cache instead. It prunes the cache to keep it 10% below the specified threshhold.

Because using a disk cache can also reduce performance, the default setting might not be optimal for your application. You can adjust this setting to achieve the best trade-off between speed and available memory.

Enter an integer value to specify the maximum size of the memory cache in megabytes. By specifying a value of 0 megabytes you disable pruning altogether, which allows for an unlimited memory cache. The default value of this setting is 10 megabytes.

Remote Call Options

There are two significant Remote Call domain parameters.

RCCBL Redirect

You must set the RCCBL Redirect option for remote call through PSADMIN.

Enter *0* to disable redirection and*1* to enable redirection. Redirection causes the server process to retain intermediate work files that are used to pass parameter values between the server process and a RemoteCall/COBOL program for debugging purposes. Redirect should always be used, except for debugging. Work files are written to the /LOGS directory with .in and .out extensions.

RCCBL PRDBIN

Use this parameter to specify where RemoteCall can find the COBOL executables. By default, RemoteCall looks in a predefined location. This might not be the location where you've installed them on your system:

- In UNIX, RemoteCall looks in $PS_HOME/cblbin.
- In Windows, RemoteCall looks in %PS_HOME%\cblbin%PS_COBOLTYPE%. The %PS_COBOLTYPE% variable contains a single letter that indicates the character encoding for the database platform. It's automatically set to one of the following values when the application server starts:
  - *U* — Unicode.
  - *A* — Non-Unicode.
  - *E* — EBCDIC.

To override this default behavior, set RCCBL PRDBIN to the absolute path of your COBOL executables, for example:

- In Windows: `RCCBL PRDBIN=c:\pscobol\MYDOMAIN\cblbin`
- In UNIX: `RCCBL PRDBIN=/app/psoft/MYDOMAIN/cblbin`

Note. This parameter doesn't appear in the PSADMIN custom configuration interface if it's not already set. You must define it by editing the application server configuration file directly. On the PeopleSoft Domain Administration menu, select *Edit configuration/log files menu*, then select *Edit psappsrv.cfg (current configuration file)* to open psappsrv.cfg in a text editor. Define the RCCBL PRDBIN parameter in the RemoteCall section of the file.

PSAPPSRV Options

The PSAPPSRV server process performs the functional requests, such as building and loading panel groups. It also provides the in-memory-caching feature for PeopleTools objects on the application server. Each server process maintains its own cache.

Min Instances

Enter the minimum number of application server instances that start when you boot the domain. There's always at least this number of instances running. This translates to the PSAPPSRV server's -m (min) parameter in the UBB file.

Max Instances

Enter the maximum number of server instances that can be started. This translates to the PSAPPSRV server's -M (Max) parameter in the UBB file.

Service Timeout

Enter the number of seconds that a PSAPPSRV waits for a service request, such as MgrGetObj or PprLoad, to complete before timing out. Service timeouts are recorded in the TUXLOG and APPSRV.LOG. In the event of a timeout, PSSAPSRV terminates itself and BEA Tuxedo automatically restarts this process.

Recycle Count

Enter the number of service requests that each server has carried out before being terminated (intentionally) and then immediately restarting. Servers must be intermittently recycled to clear buffer areas. The time that is required to recycle a server is negligible, occurring in milliseconds. The recycle count does not translate into a native BEA Tuxedo parameter in the PSAPPSRV.UBB file. Instead, the value is stored in memory and is managed by a PeopleSoft server.

Allowed Consec Service Failures

Enter a number greater than 0 to enable dynamic server processes to restart for service failures. To disable this option, enter 0. The default is 2. The value that you enter is the number of consecutive service failures that will cause a recycle of the server process. This is a catchall error handling routine that enables PSAPPSRV, PSQCKSRV, and PSAMSRV to terminate themselves if they receive multiple, consecutive, fatal error messages from service routines. Such errors should not occur consecutively, but if they do, the server process must be recycled or cleansed. A retry message appears on the client browser when this occurs.

Max Fetch Size

The default is 5000 (K). Enter the maximum memory that is used by the server to store fetched rows for a transaction before sending the result set back to a client. If the memory limit is exceeded, the client receives the rows retrieved with a memory buffer exceeded warning. You should use the default value. PSAPPSRV supports nonconversational transactions, so this parameter provides a way to balance high-volume throughput with the needs of users working with large volumes of data. A value of 0 means unlimited memory is used. The memory is not preallocated, but it is acquired as needed for each transaction.

Auto Select Prompt

Enter 1 (the default) to enable automatic prompting on lookup pages. When the user selects the prompt lookup button, the application server automatically returns all values for that field, up to 300 rows. If necessary, the user can refine the search further by entering partial data in the Search By field.

Enter 0 to require the user to enter a partial value before the automatic prompt list appears.

Tuxedo Queue Size

This parameter determines the threshold for the BEA Tuxedo queue size and is used for Pub/Sub processing only.

See *Enterprise PeopleTools 8.46 PeopleBook: Integration Broker*, "Tuning Messaging System Performance," Throttling Dispatched Messages Through the Messaging System.

PSANALYTICSRV Options

PSANALYTICSRV relates to the server processes that are associated with the *analytic server framework*.

Min Instances

Enter the minimum number of analytic server instances that start when you boot the application server domain. There's always at least this number of instances running. The default value of this parameter is *3*.

Max Instances

Enter the maximum number of analytic server instances that can result from dynamically spawning new processes. The default value of this parameter is *3*.

See Chapter 5, "Managing Analytic Servers," Configuring and Starting Analytic Servers, page 80.

Analytic Instance Idle Timeout

Enter the number of minutes of inactivity before the analytic instance times out and is unloaded.

This value takes effect only if the PeopleCode AnalyticInstance class Load method specifies a value of *-1* for its IdleTimeOut parameter when loading an analytic instance. This includes Load PeopleCode that's launched from an analytic grid, which enables you to avoid having to explicitly specify a timeout.

The default value of this parameter is *0* (no timeout limit) for domains that are configured with a developer template, and *30* minutes for other domains.

PSSAMSRV Options

The PSSAMSRV server process communicates through the BEA Tuxedo conversational mode. It performs transactional SQL requests (updates).

Min Instances

Enter how many servers are started at boot time. This translates to the PSSAMSRV server's -m (min) parameter in the UBB file.

Max Instances

Enter the maximum number of servers that can be started. This translates to the PSSAMSRV server's -M (Max) parameter in the UBB file.

Service Timeout

Enter the number of seconds that the server processes waits for a request before timing out. This stops runaway processes, like an rccbl timeout.

Recycle Count

Enter the number of service requests that each server carries out before being terminated (intentionally). Tuxedo immediately restarts the server. Servers must be intermittently recycled to clear buffer areas. The time that is required to recycle a server is negligible, occurring in milliseconds. The recycle count does not translate into a native BEA Tuxedo parameter in the PSAPPSRV.UBB file. Instead, the value is stored in memory and is managed by a PeopleSoft server.

Allowed Consec Service Failures

Enter a number greater than zero to enable dynamic server process restarts for service failures. To disable this option, enter 0. The default is 2. The value that you enter is the number of consecutive service failures that cause a recycle of the server process. This is a catchall error handling routine that enables PSAPPSRV, PSQCKSRV, and PSSAMSRV to terminate themselves if they receive multiple, consecutive, fatal error messages from service routines. Such errors should not occur consecutively, but if they do, the server process must be recycled or cleansed. A retry message appears on the client browser when this occurs.

Max Fetch Size

The default is 32 (K). Enter the maximum memory that is used by the server to store fetched rows for a transaction before sending results to the client and refilling the memory buffer. When the memory limit is reached, the server sends rows to the client, but then resumes refilling the buffer and sending results to the client until the query is complete. You should leave the default value unchanged.

PSSAMSRV supports conversational transactions, so this parameter enables users to tune performance by adjusting the number of network round-trips that are required for the average transaction. A value of 0 causes unlimited memory to be used, which means one round-trip no matter how large the result set. The memory is not preallocated, but is acquired as needed.

PSQCKSRV Options

The PSQCKSRV is an optional server process to improve performance. Essentially, the PSQCKSRV, or quick server, is a copy of the PSAPPSRV. It performs quick requests, such as nontransactional (read-only) SQL requests. The PSQCKSRV improves overall performance by enabling the PSAPPSRV process to direct a portion of its workload to PSQCKSRV.

Min Instances

Enter how many servers are started at boot time. This translates to the PSQCKSRV server's –m (min) parameter in the UBB file.

Max Instances

Enter the maximum number of servers that can be started. This translates to the PSQCKSRV server's –M (Max) parameter in the UBB file.

Service Timeout

Enter the number of seconds that a PSQCKSRV waits for a request before timing out. This stops runaway processes, like an rccbl timeout. This applies to incremental PSQCKSRV servers that are dynamically started by the Max Instances parameter.

Recycle Count

Use the PSAPPSRV specifications.

Allowed Consec Service Failures

Enter a number greater than zero to enable dynamic server process restarts for service failures. To disable this option, enter 0. The default is 2. The value that you enter is the number of consecutive service failures that will cause a recycle of the server process. This is a catchall error handling routine that enables PSAPPSRV, PSQCKSRV, and PSAMSRV to terminate themselves if they receive multiple, consecutive, fatal error messages from service routines. Such errors should not occur consecutively, but if they do, the server process must be recycled or cleansed. A retry message appears on the client browser when this occurs.

Max Fetch Size

Use the PSAPPSRV specifications.

PSQRYSRV Options

PSQRYSRV handles the SQL that is generated by PeopleSoft Query (PSQED.EXE). With PSQRYSRV configured, SQL-intensive, complicated, user-defined queries are offloaded to a dedicated server process, thus freeing PSAPPSRV and PSQCKSRV to handle the SQL requests for which they are more suited.

PSQCKSRV also processes SQLRequest services; however, if PSQRYSRV is configured, it processes all SQLRequests that are initiated specifically by PSQuery (SQLQuery:SQLRequest).

Like the PSQCKSRV server process, PSQRYSRV is an optional server process. However, if you allow users to initiate queries from PeopleSoft Query, you should take advantage of this server process.

Min Instances

Enter how many servers are started at boot time. This translates to the PSQRYSRV server's –m (min) parameter in the UBB file.

Max Instances

Enter the maximum number of servers that can be started. This translates to the PSQRYSRV server's –M (Max) parameter in the UBB file.

Service Timeout

Enter the number of seconds that PSQRYSRV waits for a request before timing out. This stops runaway processes.

Recycle Count

Enter the number of service requests that each server carries out before being terminated (intentionally) by BEA Tuxedo and then immediately restarted. Servers must be intermittently recycled to clear buffer areas. The time that is required to recycle a server is negligible, occurring in milliseconds.

If the recycle count is set to 0, PSQRYSRV is never recycled.

Allowed Consec Service Failures

Enter a number greater than 0 to enable dynamic server process restarts for service failures. To disable this option, enter 0. The default is 2. The value that you enter is the number of consecutive service failures that will cause a recycle of the server process. This is a catchall error handling routine that enables PSAPPSRV, PSQCKSRV, PSQRYSRV, and PSSAMSRV to terminate themselves if they receive multiple, consecutive, fatal error messages from service routines. Such errors should not occur consecutively, but if they do, the server process must be recycled or cleansed. A retry message appears on the client browser when this occurs.

Max Fetch Size

Enter the maximum size (in KB) of a result set that is returned from a SELECT query. The default is 10000 KB. Use 0 for no limit.

Use Dirty-Read

Enter 1 to enable PSQRYSRV to read uncommitted data from a table. It is usually acceptable to use this parameter for general reporting or queries.

Enter 0 to disable dirty reads.

Note. Dirty reads are not recommended if you are reading data and doing subsequent processing based on the disposition of the data at the time that it is read. Between the time the data is read by a subsequent process and the time the unit of work is completed by the first process, any activity affecting the table data at the time a subsequent process read could be rolled back, invalidating the accuracy of the data that a subsequent process read.

Messaging Server Processes

A variety of server processes are devoted to application messaging. If you are not implementing the application messaging technology, skip through these delivered, default server processes:

- Publish & Subscribe
- PSBRKDSP

- PSBRKHND
- PSPUBDSP
- PSPUBHND
- PSSUBDSP
- PSSUBHND

These server processes act as brokers, dispatchers, and handlers of the messages in the messaging system.

See Also

Appendix B, "PeopleSoft Timeout Settings," Application Server Timeouts, page 325

Enterprise PeopleTools 8.46 PeopleBook: Integration Broker, "Administering Messaging Servers for Asynchronous Messaging"

SMTP Settings

You can send electronic mail requests, issued with workflow or PeopleCode, to the application server, and the application server, in turn, passes the requests to the specified mail server (SMTPServer). By having the application server submit the email request, you avoid having to install mail connectivity software on each client, just as you avoid having to install database connectivity software on each client in a three-tier connection. To specify the appropriate SMTP server and port to receive the email requests, you must edit the SMTP Settings section.

When set in the PSAPPSRV.CFG file, these three SMTP settings are not dynamic: SMTPGuaranteed, SMTPTrace, SMTPSendTime. They require a domain reboot to take effect.

Note. You can also control most of these settings using the PeopleCode SMTPSession class, which temporarily overrides them without changing their values in PSAPPSRV.CFG. You use this class primarily when you want to send multiple emails in a single session of the SMTP server, instead of having to change the permanent SMTP settings for every email.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleCode API Reference*, "Mail Classes," SMTPSession Class.

SMTPServer

Enter the host name and IP address of the mail server machine.

SMTPPort

Enter the port number on the mail server machine.

SMTPServer1

Enter the host name and IP address of the failover mail server machine in case the other specified server is down.

SMTPPort1

Enter the port number on the failover mail server machine.

SMTPSender

Enter the sender's internet address. This must be a valid address, such as user1@xyzcorp.com.

SMTP BlackberryReplyTo

Enter the internet address that you want to be the reply to address for Blackberry email responses. This must be a valid address such as user1@xyzcorp.com.

SMTPSourceMachine

Enter the sender's source machine name and internet address in the form of MACHINE.XYZCORP.COM. This value is required in some, but not all environments.

SMTPCharacterSet

Enter the character set that is used on the sender's machine.

SMTPEncodingDLL

Enter the name of a dynamic-link library (DLL) that is used to translate the mail message from the sender's character set (such as latin1, sjis, big5, gb, ks-c-5601-1987, or ks-c-5601-1992) to a 7-bit safe character set for transmission.

SMTPGuaranteed

Set this parameter to *1* if you want TriggerBusinessEvent email PeopleCode to be delivered through the messaging system, which provides some additional administration capabilities for ensuring delivery of the message.

If the application server isn't able to make a connection to the SMTP mail server, it attempts to resend the message up to the number of times specified by the Max Retries parameter of the PSSUBHND_dflt or PSMBHND messaging server process.

The system doesn't try to resend the message indefinitely. When the Max Retries limit is reached without success, the subscription contract for the message is set to a status of *Error*. You can then manually resubmit the message from the Application Message Monitor.

By enabling this feature you implement a mechanism to ensure that emails get routed to the appropriate place in case SMTP mail fails for reasons such as network timeouts, downed mail servers, invalid parameters, and so on.

SMTPTrace

Enter *1* to enable the tracing of all email details to the log file when LogFence is set to *5*. Enter *0* to disable it. With this option, you can reduce the log file size for high-volume email users.

SMTPSendTime

Enter 1 to have messages contain a send time that is populated by the application server. Enter 0 to leave the send time blank and have it populated by the receiving gateway (depending on the gateway).

SMTPUserName

Enter the user name to log in to the SMTP server. This applies only when authentication is enabled on the SMTP server.

SMTPUserPassword

Enter the password for the user specified by *SMTPUserName* to access the SMTP server. This applies only when authentication is enabled on the SMTP server.

SMTPUserName1

Enter the user name to log in to the failover SMTP server. This applies only when authentication is enabled on the failover SMTP server.

SMTPUserPassword1

Enter the password for the user specified by *SMTPUserName1* to access the failover SMTP server. This applies only when authentication is enabled on the failover SMTP server.

SMTPTimeToWaitForResult

Enter the time in milliseconds for the mail system to wait for the result of sending each email. If the time is set to *0*, the system doesn't wait, and the returned result will be always be *%ObEmail_SentButResultUnknown (= -1)*. If you set this parameter to *-1*, the system will wait for the completion of the send process. The default value of this setting is 10000 (ten seconds).

SMTP Further Considerations

Keep in mind the following considerations:

- PeopleSoft mail integration is on the application server only.

  Currently, PeopleSoft software does not support VIM/MAPI, because this option is client-side-only integration, and PeopleSoft Internet Architecture applications run on the server-side.

- The application server communicates directly with an SMTP server through telnet by using standard SMTP commands with Multipurpose Internet Mail Extensions (MIME) 1.0 messages.
- PeopleSoft software currently supports UTF-8 encoding of the email messages out-of-the-box, and you can encode email messages in other ways.

  With server-side integration, you do not have to certify any specific email client application. You can use any application to read email.

- You can send email using the PeopleCode Mail classes, as well as the SendMail and TriggerBusinessEvent PeopleCode built-in functions. PeopleSoft recommends using the Mail classes for all email sent from a PeopleSoft application.
- Outside of PeopleSoft applications, you use PSMAIL.EXE, which is an executable that is for use by advanced developers. PSMAIL.EXE can send email messages through SMTP based on data that is passed

as parameters to the executable or from an input file. This executable is primarily used for PeopleSoft Process Scheduler programs.

Interface Driver Options

Set the following parameter for configuring the interface driver for business interlinks.

SCP_LOCALE

Enter the RPS_LOCALE string, which the driver sends to the Supply Chain Planning (SCP) server.

PSTOOLS Options

You may need to set the following parameters in advanced configurations.

EnablePPM Agent

Enter*0* to disable the Performance Monitor Agent. This setting overrides the value for this parameter that is set in the database. The default value of *1* enables the performance monitor agent.

Add to CLASSPATH

The CLASSPATH environment variable tells the Java Virtual Machine (JVM) and other Java applications where to find the Java class libraries, including any user-defined class libraries. Because PeopleTools automatically generates CLASSPATH entries for core, delivered class libraries, use this field to specify additional class libraries that the PeopleSoft software needs to access.

Java VM Options

Specify additional options to be passed to the JVM that's loaded by the PSAPPSRV process. Separate the options with spaces, for example:

-Xrs -Xmx256m -Xms128m

If the domain will run as a Windows service, you must specify at least the default option, -Xrs.

Refer to your JRE documentation for valid JVM options.

Proxy Host

If the HTTP destination, such as the application messaging gateway or business interlink remote host, is behind a proxy server for security reasons, enter the distinguished name of the proxy server, as in proxy.peoplesoft.com.

Proxy Port

Enter the port number on which the proxy server is listening for transmissions. For instance, 80 is a typical default port number.

Non Proxy Hosts

Enter a list of the hosts that should be connected to directly, not through a proxy server. Separate the hostnames with a pipe symbol (|). You can use an asterisk (*) as a wildcard character to specify a pattern of similar hostnames.

For example, `localhost|*.peoplesoft.com` indicates that the local host and all hosts with names ending in .peoplesoft.com will be accessed directly.

Note. The length of this string cannot exceed 1024 characters.

Character Set (UNIX or USS Only)

Enter the character set to use for processing external data on the Unix application server, such as configuration file and log file name and body, and attachment filenames.

The default value is *latin1*, which supports all Western European languages, including English. If this application server will be used to process only Western European data, you should accept the default for this parameter.

Otherwise, select a character set from the following list corresponding to the languages that this application server will process:

UNIX Character Set	USS CCSID	Description (Similar Windows Codepage)
latin1	ccsid1047	Western Europe (CP1252) (Default)
latin2	ccsid870	Central Europe (CP1251)
sjis	ccsid930	Japanese Shift-JIS (CP932)
big5	ccsid937	Traditional Chinese (CP950)
gb2312	ccsid935	Simplified Chinese (CP936)
ks-c-5601-1987	ccsid933	Korean Wansung (CP949)
ks-c-5601-1992	ccsid933	Korean Johab (CP1361)
utf8	(No USS equivalent)	Unicode (No Windows equivalent)

Note. The character set of the application server and the character set of any Microsoft Windows workstations connecting to that application server must match.

The *utf8* option is valid only when the locale character set is UTF-8.

Suppress App Error Box (Microsoft Windows Only)

Enter *y* to suppress an application error box or message from appearing after an application error occurs. Enter *n* to view error dialogs and message boxes.

Note. If the system generates an error box for an application server process and this parameter is set to *n*, BEA Tuxedo can't restart the down process until you close the error box.

DbFlags

The following values are valid for the DbFlags parameter:

Value	Description
0	Enable the %UpdateStats meta-SQL construct.
1	Disable the %UpdateStats meta-SQL construct.
2	Ignore the Truncate command for DB2 UNIX/NT. Use Delete instead.
4	Disable a secondary database connection (used with the GetNextNumberWithGapsCommit PeopleCode function). This prevents the creation of a secondary database connection, bundling all SQL into a single unit of work. Without the additional database connection, the database row lock is held for a longer time, reducing concurrency in a multiple-user environment. **Note.** Analytic instance processing requires a secondary database connection, so if you're using analytic servers, ensure that this value is not set.
8	Disable a persistent second database connection (used with the GetNextNumberWithGapsCommit PeopleCode function). This creates a second database connection in each GetNextNumberWithGapsCommit call, then immediately closes the second connection. This keeps the number of database connections to a minimum, but requires each call to create a new database connection on demand. **Note.** The performance impact of making a new database connection is significant, especially in high volume user production environments. Don't use this setting without carefully considering its effect.

DbFlags uses a bit mask so that you can specify one or more of these values. You set this parameter to the total of the values that you want to apply. For example, to disable %UpdateStats and ignore the Truncate command, set DbFlags to *3* (setting bits one and two).

The default is value is *1*.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Application Engine*, "Using Meta-SQL and PeopleCode," Using PeopleCode in Application Engine Programs and *Enterprise PeopleTools 8.46*

PeopleBook: PeopleCode Language Reference, "PeopleCode Built-in Functions," PeopleCode Built-in Functions and Language Constructs.

Suppress SQL Error

This option is not available through the PSADMIN interface, but it exists in the PSTOOLS section of the PSAPPSRV.CFG file for small, medium, and large configurations.

For security purposes, this option has a default value of *1* to prevent SQL error details from being displayed to users. Any SQL errors that occur don't display details, but refer users to consult the system log. The details that were in the SQL message are written to the log file. This helps to prevent SQL injection vulnerabilities.

If you want SQL error details to be visible to users, set this property as follows:

```
Suppress SQL Error=0
```

Note. In developer configurations, the Suppress SQL Error option doesn't exist in PSAPPSRV.CFG, and the system assumes a value of *0*.

See Also

Enterprise PeopleTools 8.46 PeopleBook: PeopleCode Developer's Guide, "Improving Your PeopleCode," Preventing SQL Injection

PeopleSoft Integration Broker Options

The following parameter applies to PeopleSoft Integration Broker.

Min Message Size for Compression

Use this parameter to configure the threshold of a message before the system compresses the message.

IB Profile Information

Enter *Y* to turn on profiling for both synchronous and asychronous Integration Broker processing. The default value of *N* disables profiling for Integration Broker processing.

Message View Size

You can provide the maximum size message that the Message Monitor will display through a PIA page. Messages larger than this size will be downloaded to a file.

Search Indexes

Use this option to specify the location of all the files pertaining to the search index. Index name is same as the location.

See Chapter 8, "Building and Maintaining Search Indexes," Specifying the Index Location, page 176 and Chapter 8, "Building and Maintaining Search Indexes," Sharing Indexes Between Application Servers and PeopleSoft Process Scheduler, page 178.

PSRENSRV Options

PSRENSRV is a modified web server designed for real time event notification. The primary purpose of PSRENSRV is to publish events to the browser.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft MultiChannel Framework*, "Configuring REN Servers".

log-severity_level

This is the log severity level for the PSRENSRV process. Settings are Error, Warning, Notice or Debug. Default is Warning.

io_buffer_size

This is the TCP buffer size when serving content. This should not exceed a value of *65536*.

default_http_port

This is the REN servers http port. The default value is *7180*.

default_auth_token

The fully qualified domain name of the application server. This value should match the value of the web server's authentication token domain.

PSPPMSRV Options

PSPPMSRV servers subscribe to performance metrics published by the web service at the PPMI URL (entered into the Performance Monitor administration pages) and insert them into the database. If you select *Y* when you are asked whether you want Performance Collators configured, then the number of PSPPMSRVs specified in Min Instances=1 will be started. Min and Max instances should not be set to the same value, as new servers are not spawned on demand.

Min Instances

The number of servers started at boot time. This translates to the PSPPMSRV server's –m (min) parameter in the UBB file.

Max Instances

The maximum number of servers that can be started. This translates to the PSPPMSRV server's –M (max) parameter in the UBB file.

Select Server Process Options

After you enter all of the previous parameter values for the application server, PSADMIN prompts you for the following server process options. You can use these prompts to reduce the number of server processes that start when the domain boots. This, in turn, makes your configuration simpler while conserving system resources.

For instance, if you enter n for any of the following prompts, the corresponding server process (or a set of server processes) is not configured for the domain. If you enter n to all of the prompts, the domain will contain only the required server processes.

Do you want the Publish/Subscribe servers configured?

If you want the application messaging server processes to be configured and booted, enter y. If you are not implementing the application messaging technology, enter n.

Note. In addition to setting this option, in PeopleSoft Integration Broker you must also activate the domain on which the pub/sub server resides before you can use the pub/sub system.

See *Enterprise PeopleTools 8.46 PeopleBook: Integration Broker*, "Using Integration Broker Monitor," Working with Pub/Sub Server Domains.

Move quick PSAPPSRV services into a second server (PSQCKSRV)?

Enter n if very few clients access the domain and concurrency is not an issue. Enter y to enable the PSQCKSRV in situations where concurrency and optimal transaction throughput are needed.

Move long-running queries into a second server (PSQRYSRV)?

If you want all user-generated queries to be initiated by PSQuery and handled by a dedicated server process, enable this option to improve overall performance.

Do you want JOLT configured?

The BEA Jolt listener is required to support the PeopleSoft Internet Architecture. If you are not going to deploy the PeopleSoft Internet Architecture, there is no need to configure BEA Jolt.

Do you want JRAD configured?

JRAD supports specific configurations. Accept the default unless you are attempting to configure JRAD for use with the BEA Jolt internet relay.

Do you want to enable PeopleCode Debugging?

Enter y to debug PeopleCode programs with the current domain.

Do you want the Optimization engines configured?

If you want the optimization engine server processes configured and booted, enter y. If you are not implementing the optimization technology on this domain, enter n. Some PeopleSoft applications utilize optimization engine to run computation-intensive algorithms to find the optimal recommendations for business decisions. If you install such applications, optimization engines must be configured to enable this functionality.

Do you want Event Notification configured?

Select *Y* to start the PSRENSRV servers.

See Chapter 4, "Setting Application Server Domain Parameters," PSRENSRV Options, page 71.

Do you want MCF Servers configured?

Select *Y* to start the Multi Channel Framework servers.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft MultiChannel Framework*, "Configuring MCF Queues".

Do you want Performance Collators configured?

If the domain is servicing a Performance Monitor database, select *Y* to start the PSPPMSRV servers.

See Chapter 4, "Setting Application Server Domain Parameters," PSPPMSRV Options, page 71.

CHAPTER 5

Managing Analytic Servers

This chapter provides an overview of the analytic server framework and batch processing of analytic instances and discusses how to:

- Configure and start analytic servers.
- Administer analytic servers.
- Administer analytic tables.
- Create, delete, and copy analytic instances.
- Load and unload analytic instances.

Understanding the Analytic Server Framework

This section discusses:

- Analytic server framework overview.
- Analytic server process flow and behavior.

Analytic Server Framework Overview

When a program doesn't "maintain state" or when the infrastructure of a system prevents a program from maintaining state, it's known as a *stateless* program or system. It can't take information about the last session into the next session, such as settings the user makes or conditions that arise during processing.

For example, the HTTP protocol is stateless. Additional schemes, such as cookies, are necessary to maintain state in the HTTP (web) environment.

PeopleTools is architected primarily around a stateless model of client/server connectivity. This model enables users' application sessions to be preserved even if servers are shut down or rebooted. All session state is maintained by the client and is transferred to the server with each request. As long as an application server is up and running, a user's session remains active and functional, and any application server can perform requested transactions.

However, with some products, such as PeopleSoft Analytic Calculation Engine or PeopleSoft Optimization Framework, running a calculation on a multi-dimensional model is likely to produce far more data than is reasonable to shuttle between a client and server to maintain a stateless connection. For performance reasons, the calculations are performed completely in memory. If these calculations were to be synchronized and stored in the database so that a stateless connection could be maintained, performance would suffer significantly.

The *analytic server framework* provided by PeopleSoft is a general server infrastructure designed to meet the needs of PeopleSoft products that process large amounts of data in memory. It provides a *stateful* model of client/server connectivity that these products require to be part of the PeopleTools system, by keeping track of configuration settings, transaction information, and other data for a session.

For example, client software could request that an analytic model or optimization model be recalculated in one transaction, then retrieve the results of the calculation on that model at a later time. A server process handles these requests, and maintains the model state and calculated data in memory between the requests. Additional transactions can then modify the model and perform recalculations on it without shuffling all of the data between the client and the server or dumping all the data to a database, thus preserving in-memory performance.

A large model might take a long time to load. In the event that a user's session times out and is terminated, the loading and calculation of the model continues, and enables the user to return to the model at a later time in a new session.

The elements of the analytic server framework are:

- PSANALYTICSRV server.

  PSANALYTICSRV is a Tuxedo managed PeopleSoft application server process, like PSAPPSRV. It contains both the analytic calculation engine and the optimization engine. Multiple instances of PSANALYTICSRV can run in an application server domain. The current condition of each PSANALYTICSRV instance is tracked in system tables.

- Analytic server administration pages.

  The Analytic Domain Summary page provides current information about the application server domains with PSANALYTICSRV running that are attached to the current database.

  The Analytic Servers page enables you to inspect the individual analytic server instances within the running domains, with information about their analytic types and analytic instances, operations, and timeout intervals. You can also halt processes individually on this page.

- Analytic table administration pages.

  The Purge Delete Tables page displays the names of delete tables relevant to an analytic type or analytic instance, and enables you to clear the data from the tables.

  The Synchronize Table Versions page enables you to resynchronize versions of analytic type or analytic instance data and the PSOPTSYNC table that are out of synchronization after you use PeopleSoft Data Mover to move data from one database to another.

- Pages for creating, deleting, and copying analytic instances.

  The Create Analytic Instance, Delete Analytic Instance, and Copy Analytic Instance pages enable you to define and manage analytic instances that you can then load to inspect and debug your analytic models.

- The Analytic Instance Load/Unload page.

  The Analytic Instance Load/Unload page enables you to load analytic instances so you can view them within the Analytic Model Viewer, then unload the analytic instances that you no longer need.

- Various supporting enhancements in several PeopleTools products.

  These products include PeopleSoft Analytic Calculation Engine, PeopleSoft Optimization Framework, Crystal Reports, PeopleSoft Performance Monitor, PeopleSoft Process Scheduler, PeopleSoft Application Engine, PSADMIN, and PeopleCode.

Note. Information about the role that the analytic server framework plays in these products can be found in the documentation for each product.

Analytic Server Terms

The following terms are useful in understanding analytic server technology:

Analytic type	A description of a data set to be loaded and the calculations to be performed on the data set in the analytic server framework. Multiple calculation engines such as the analytic calculation engine or the optimization engine can be associated with an analytic type.
Analytic instance	One instance of an analytic type. You can create multiple instances of the same analytic type.
Analytic server	The primary functional element of the analytic server framework, called PSANALYTICSRV. Each application server domain can include zero or more analytic servers.
Analytic server instance	One running instance of an analytic server. You can run multiple instances of PSANALYTICSRV for a given domain. Each running analytic server instance can hold one analytic instance.
Analytic engine	The portion of the analytic server framework that's responsible for managing analytic instances.
Analytic engine type	One of the following: • Analytic Server. • Application Engine Server. • Application Engine.

Analytic Server Features

The analytic server framework has the following features:

- It's dedicated to the storage and management of large models.
- It's supported on all PeopleTools application server platforms.
- It runs PSANALYTICSRV as a Tuxedo managed server.
- You can configure the minimum and maximum number of analytic server instances per domain that are running at one time.
- You can specify a timeout for a loaded analytic instance. If the analytic instance isn't referenced within the timeout interval, it's discarded.
- Multiple domain environments are supported, in which an analytic instance can be loaded in one Tuxedo domain, and a user can access the analytic instance from another domain.
- You install, configure, and administer analytic servers using the same facilities as with other servers.
- You can shut down an analytic server and discard any loaded analytic instances.
- You use the standard PeopleTools mechanisms to troubleshoot, trace, log and debug analytic servers.

See Also

Chapter 2, “Using the PSADMIN Utility,” Using the Quick-Configure Menu, page 10

Chapter 4, “Setting Application Server Domain Parameters,” page 43

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Analytic Calculation Engine

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Optimization Framework

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Performance Monitor

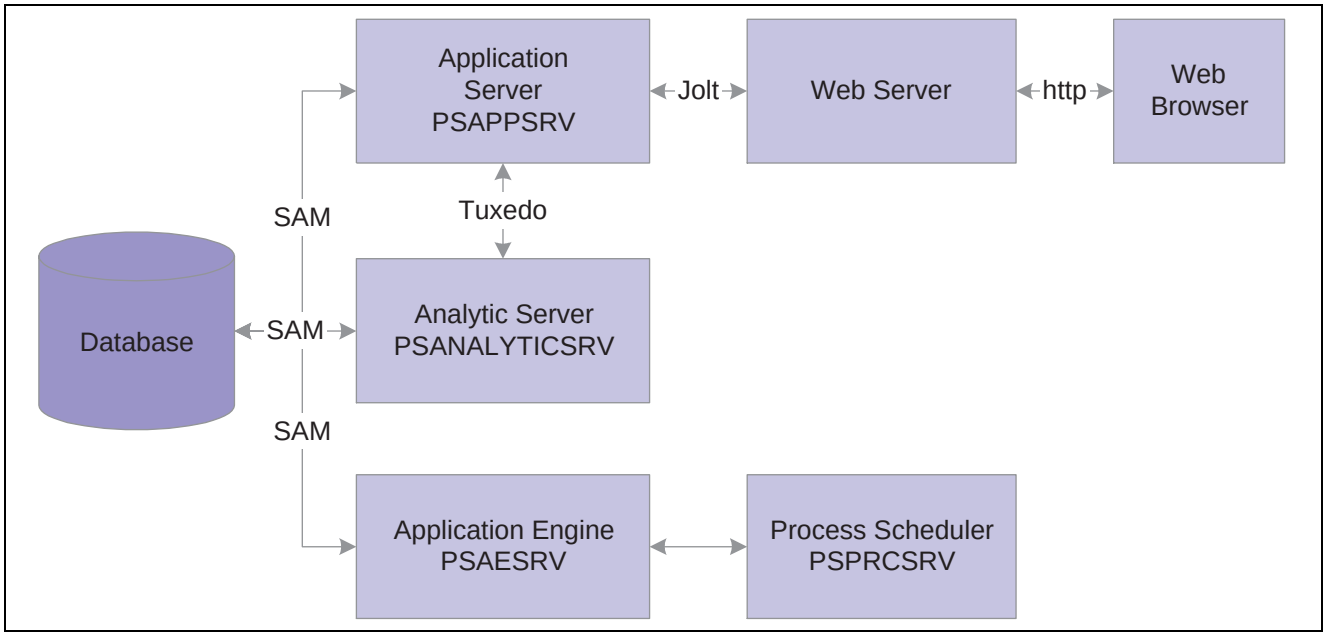
Analytic Server Process Flow and Behavior

PeopleSoft session activity, such as a user action, a component interface operation, or a message subscription, launches PeopleCode that requires the application server to invoke the analytic calculation engine or the optimization engine to process an analytic instance.

The database maintains a list of all the available PSANALYTICSRV (analytic server) instances, their status, and any analytic instances currently loaded, so it can properly select analytic server instances for new analytic instances, and direct subsequent requests to the proper analytic server instance. When an analytic server instance starts, the database is updated.

When a running program requests the creation of an analytic instance, the analytic server framework considers all available PSANALYTICSRV instances in the same application server domain and allocates one of them from the pool of idle server instances to handle this particular analytic instance. Any further load or recalculate operations requested by the application for this analytic instance are directed back to the same analytic server instance for processing.

Note. If there are no idle server instances, the analytic server framework can spawn additional server instances up to a maximum limit that you can define in the application server domain configuration. If this maximum is reached, the system attempts to allocate a server instance from a running analytic server in another domain.



Analytic server architecture

Once an analytic server instance takes over processing the analytic instance, that processing becomes independent of the status of the application server. The core functionality provided by the analytic server framework is the ability to host analytic instances for an indeterminate amount of time in an environment where that data can persist across multiple sessions, and where that data can be accessed without requiring its entire content to be transferred from server to client and back.

Analytic Instance Access

Access to the analytic instances maintained by the analytic server environments is supported only through PeopleCode programs. The environments in which PeopleCode can run include:

- The application server (PSAPPSRV).
- PeopleSoft Application Engine accessed by PeopleSoft Process Scheduler (PSAESRV or psae)
- PeopleSoft Application Engine run from the command line (psae).

An allocated server instance doesn't need to be running in the same Tuxedo domain or on the same server machine as the application server. Once it's allocated, the initiating user is redirected to an application server that's running on the same domain and server machine as the analytic server.

PeopleCode that's running in any PSAPPSRV process can access analytic instances that were loaded by any other PSAPPSRV process, regardless of the Tuxedo domain in which it's running. When the program requests access to such an analytic instance, the running PeopleCode program is restarted and the web server is notified to redirect the request back to an application server within the Tuxedo domain where the analytic instance is loaded. This application server is then able to directly contact the PSANALYTICSRV server with the loaded analytic instance.

Note. PeopleCode that's running in a given Application Engine environment can access only analytic instances that were loaded in the same process.

Secondary Database Connection

A secondary database connection is used to prevent unexpected table locks when you run an analytic calculation. The secondary connection isn't opened until an analytic instance is referenced in a PeopleCode program. A secondary connection is required regardless of whether the analytic calculation is run by an application engine program as a batch process or directly by an online application.

By default, the secondary connection is persistent for improved performance. If you find that the persistent connection imposes too much overhead, you can change it to an on-demand connection by setting bit eight of the DbFlags application server and process scheduler domain parameter.

Note. A non-persistent connection can significantly affect system performance, so consider this setting carefully.

You can use DbFlags bit four to disable the secondary connection altogether, but analytic instance processing requires it, so ensure that DbFlags does not have bit four set.

Errors and Abnormal Process Termination

Any errors that occur while processing an analytic server request result in the PeopleCode program returning an error code or throwing a PeopleCode exception.

If an analytic server instance that's hosting an analytic instance terminates unexpectedly, the loaded analytic instance is lost and unrecoverable. However, the analytic instance status still appears on the Analytic Server Administration pages. The domain monitor (PSMONITORSRV) discovers the unexpected termination and cleans up the status information.

See Also

Chapter 5, “Managing Analytic Servers,” Administering Analytic Servers, page 82

Chapter 4, “Setting Application Server Domain Parameters,” DbFlags, page 69

Enterprise PeopleTools 8.46 PeopleBook: PeopleCode API Reference, “Analytic Calculation Engine Classes,” Load

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Process Scheduler

Understanding Batch Processing of Analytic Instances

The analytic server framework integrates with and works with PeopleSoft Process Scheduler using PeopleSoft Application Engine, because PeopleSoft Application Engine can access the analytic calculation engine and the optimization engine directly.

When PeopleSoft Process Scheduler launches an Application Engine job to process an analytic instance, PeopleSoft Application Engine handles the entire job directly by loading the analytic engine within its own process rather than using a server.

This is true whether PeopleSoft Process Scheduler submits the job to the PSAESRV process, or launches PeopleSoft Application Engine using the *psae* command.

See Also

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Analytic Calculation Engine

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Optimization Framework

Configuring and Starting Analytic Servers

This section discusses how to:

- Enable PSANALYTICSRV.
- Specify analytic server instance quantities.
- Start PSANALYTICSRV.

You can specify whether an application server domain includes the PSANALYTICSRV Tuxedo managed application server process, and specify the maximum number of analytic server instances that you want the domain to support. You use the Quick-Configure menu of the PSADMIN utility to enable, configure, and start analytic server instances.

Enabling PSANALYTICSRV

You access the Quick-Configure menu of PSADMIN by selecting *Configure This Domain* from the Domain Administration menu.

On the Quick-Configure menu, enter the menu item number for *Analytic Servers* to toggle the setting for that entry to *Yes*, so the domain will include instances of PSANALYTICSRV when it boots.

See Also

Chapter 2, “Using the PSADMIN Utility,” Using the Quick-Configure Menu, page 10

Specifying Analytic Server Instance Quantities

Before you boot the application server domain, specify the appropriate minimum and maximum number of allowed analytic server instances. The values you specify depend on your assessment of how many users you expect to be using applications that process analytic instances.

Consider the typical number of analytic instances in a domain that are being processed at any given moment as your minimum, and the possible total number of analytic instances that might simultaneously require processing as your maximum. The most appropriate values produce the fastest system response without unused server instances consuming memory unnecessarily.

To specify analytic server instances:

1. On the Quick-Configure menu for the domain, enter the menu number for *Custom Configuration*.

   The Custom Configuration environment launches, and prompts you to indicate whether you want to change any configuration values.

2. Enter *y* to indicate that you want to change configuration values.

   Custom Configuration prompts you to decide whether to change any values for each configuration item in turn.

3. Press ENTER to accept the default answer for each item presented, until the following entry appears:

   ```
   Values for config section - PSANALYTICSRV
   ```

4. Enter *y* to change the values for PSANALYTICSRV.

   You’re prompted for each value in turn.

5. Specify the minimum number of instances.

   This defines the number of analytic server instances that start when you boot the application server domain. There are always at least this number of instances running. The default value of this parameter is *3*.

6. Specify the maximum number of instances.

   This defines the maximum number of analytic server instances that can result from spawning new processes. The default value of this parameter is *3*.

7. Press ENTER to accept the default answer for each subsequent item presented. When you respond to the last item, PSADMIN loads the new configuration and the PeopleSoft Domain Administration Menu appears.

   You now can boot the domain normally.

See Also

Chapter 2, “Using the PSADMIN Utility,” Using the Quick-Configure Menu, page 10

Chapter 4, “Setting Application Server Domain Parameters,” PSANALYTICSRV Options, page 60

Starting PSANALYTICSRV

If you enabled analytic servers on the Quick-Configure menu, when you boot the application server domain, the PSANALYTICSRV process starts with the minimum number of instances that you specified.

When an application running under this domain requests an analytic instance, the analytic server framework allocates an available idle analytic server instance for that analytic instance. If no idle server instance is available, the framework spawns and allocates an additional server instance, up to the maximum that you defined.

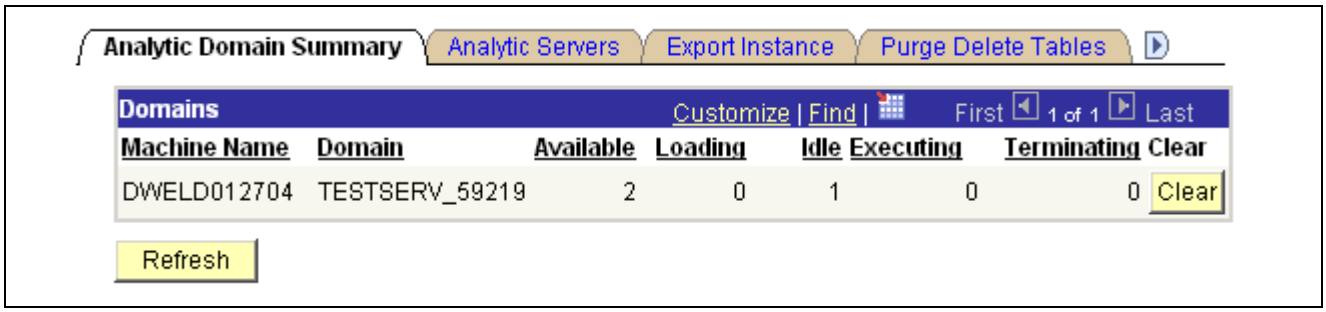
Administering Analytic Servers

This section discusses how to:

- Administer analytic server domains.
- Administer analytic server instances.

Administering Analytic Server Domains

In a browser, select PeopleTools, Utilities, Administration, Analytic Server Administration to access the Analytic Server Administration - Analytic Domain Summary page.



Analytic Server Administration - Analytic Domain Summary page

This page displays the current status of the application server domains with PSANALYTICSRV running that are attached to the current database. Each active domain is listed, along with the following information:

Machine Name Displays the network name of the computer on which the listed domain is running.

Domain Displays the name of each active domain.

Note. If a domain has been unexpectedly terminated, it might still be listed here. You can click the Clear button to remove the outdated information from the display.

Available Displays the total number of analytic server instances running in the domain.

Loading Displays the number of available analytic server instances in the domain that are currently being loaded.

Idle Displays the number of available analytic server instances in the domain that are allocated to analytic instances, but aren't actively processing them.

Executing Displays the number of available analytic server instances in the domain that are allocated to analytic instances, and are actively processing them.

Terminating Displays the number of analytic server instances in the domain that are marked as terminated, but haven't yet been shut down.

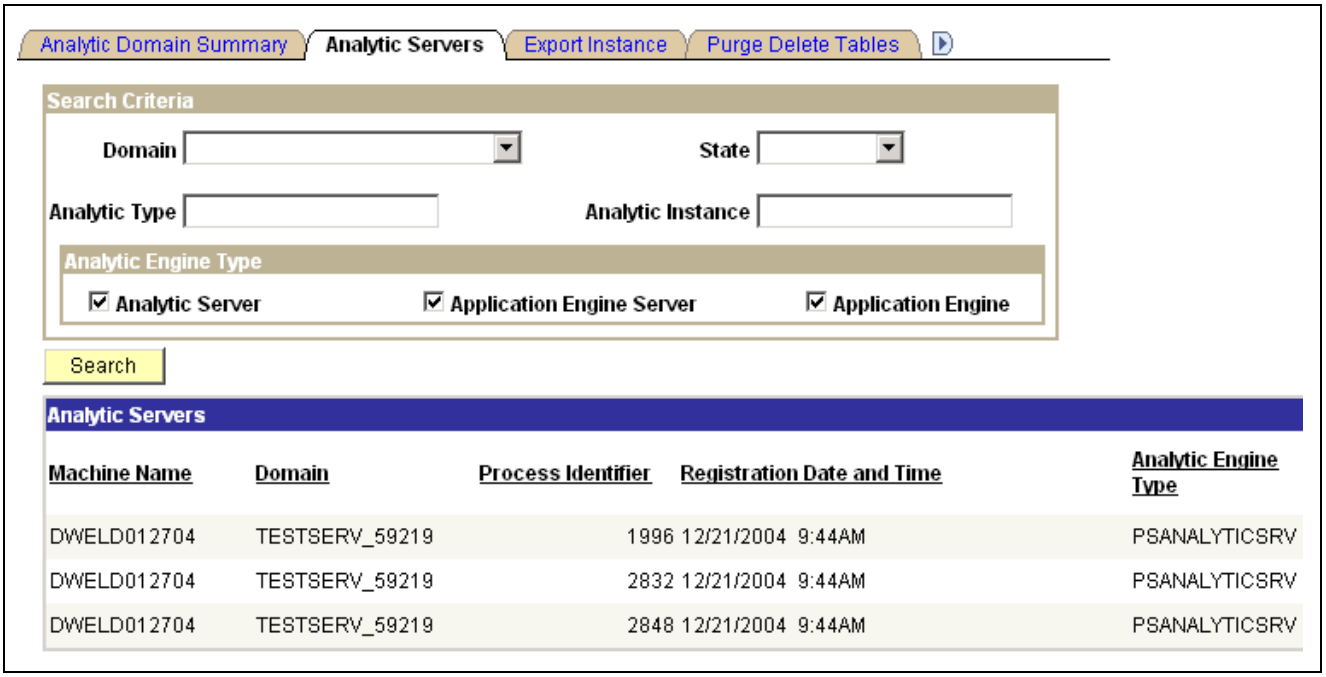
Click a domain's Clear button to remove that row from the display when the domain has been unexpectedly terminated.

Warning! Use the Clear button with caution, and only if you're certain that the domain has been unexpectedly terminated. Clearing the domain information for servers that are still running produces errors in those servers.

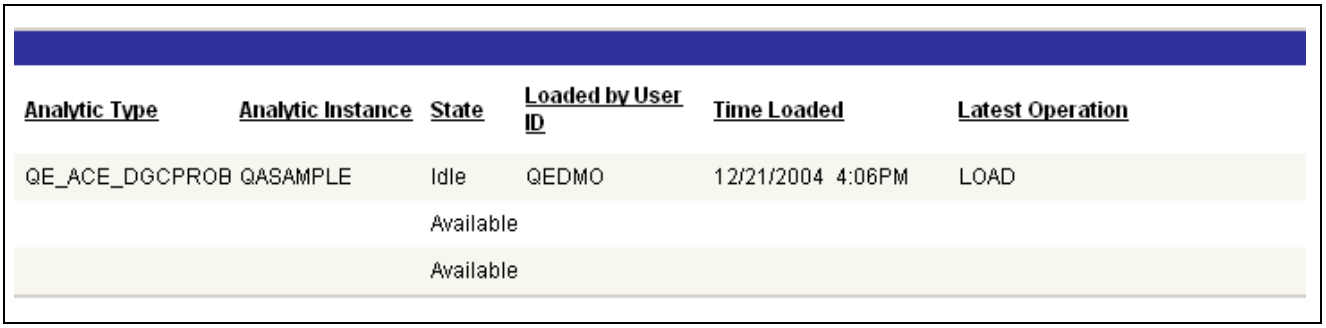
Click the Refresh button to retrieve information about any newly started domains that have PSANALYTICSRV running.

Administering Analytic Server Instances

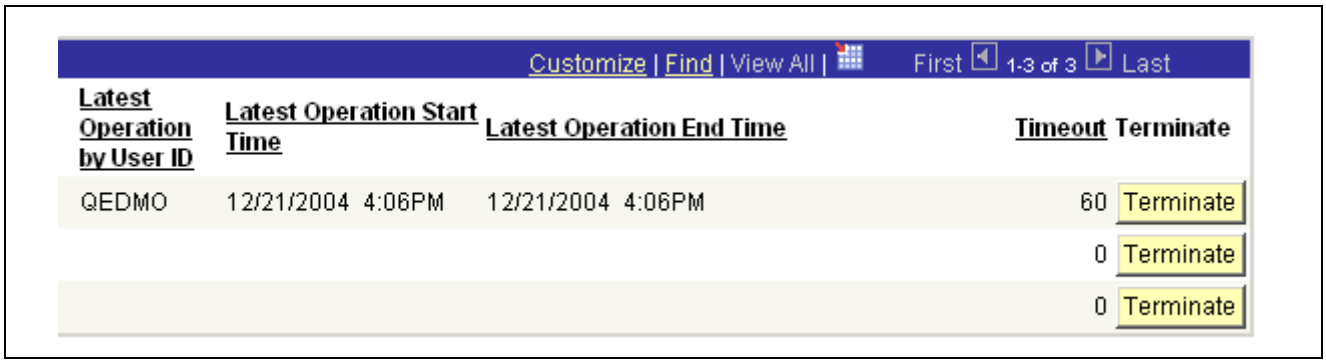
In a browser, select PeopleTools, Utilities, Administration, Analytic Server Administration, Analytic Servers to access the Analytic Server Administration - Analytic Servers page.



Analytic Server Administration - Analytic Servers page (1 of 3)



Analytic Server Administration - Analytic Servers page (2 of 3)



Analytic Server Administration - Analytic Servers page (3 of 3)

Click the Search button to retrieve status information about all analytic server instances that are running in application server domains that are attached to the current database. You can use the Search Criteria section to limit the information returned based on various criteria.

Search Criteria

Domain Select the name of an active application server domain for which you want to retrieve information.

State Select a server state to limit the search to analytic server instances with the selected state. You can select from the following states:

- *Available*
- *Registered*
- *Loading*
- *Idle*
- *Executing*
- *Terminate*

Analytic Type Select an analytic type from the set of analytic types defined in the current database. This limits the search to analytic server instances that have analytic instances of the selected analytic type loaded.

Analytic Instance Select an analytic instance from the set of analytic instances defined in the current database. This limits the search to analytic server instances that have the selected analytic instance loaded.

Analytic Engine Type Specify the process types for which you want to get status information. Select one or more of the following:

- *Analytic Server*
- *Application Engine Server*
- *Application Engine*

All three types are selected by default.

Analytic Servers

This section displays a row of status information retrieved for each analytic server instance that's returned by the search. In addition to the fields documented in the previous section, each row displays the following information:

Machine Name	Displays the network name of the computer on which the listed analytic server instance is running.
Process Identifier	Displays the operating system process ID for the listed analytic server instance.
Registration Date and Time	• Analytic server type: Displays the date and time that this analytic server booted. • Application Engine types: Displays the date and time that the application engine process loaded this analytic instance.
Loaded by User ID	Displays the user ID of the user whose activity resulted in the allocation of this analytic server instance.
Time Loaded	Displays the date and time that this analytic server instance loaded its analytic instance.
Latest Operation	Identifies the last operation that was applied to this analytic instance.
Latest Operation By User ID	Displays the user ID of the last user to access this analytic instance.
Latest Operation Start Time	Displays the date and time that the last operation on this analytic instance started.
Latest Operation End Time	Displays the date and time that the last operation on this analytic instance completed.
Timeout	Displays the timeout interval in minutes that's specified for this analytic instance. Timeout values are defined for analytic instances by the TimeOut parameter of the PeopleCode AnalyticInstance class Load method. A value of *0* indicates an unlimited lifespan for this analytic instance.
Terminate	Click to indicate that the server instance should be shut down.

Administering Analytic Tables

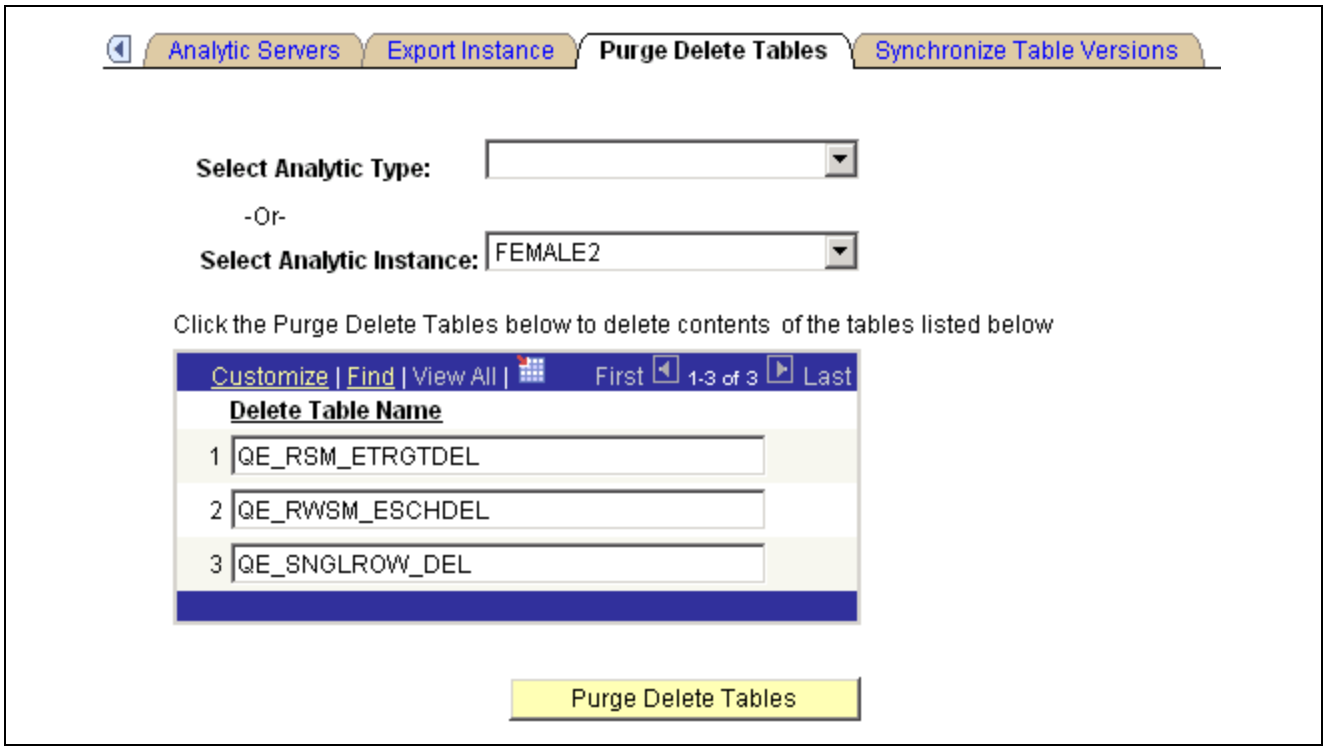
This section discusses how to:

- Purge delete tables.
- Synchronize table versions.

Purging Delete Tables

In a browser, select PeopleTools, Utilities, Administration, Analytic Server Administration, Purge Delete Tables to access the Analytic Server Administration - Purge Delete Tables page.

Note. Shut down all running analytic server processes before using this page.



Analytic Server Administration - Purge Delete Tables page

Delete tables contain rows that have been deleted from analytic instance working data. These tables accumulate data when you use triggers for database level auditing, and they're not always cleaned up after the deletes have been completed. You use this page to accomplish the cleanup manually.

Select Analytic Type -Or- Select Analytic Instance These drop-down lists are mutually exclusive. Select either an analytic type or an analytic instance for which you want to purge delete tables.

Delete Table Name Displays the names of the delete tables relevant to the analytic type or analytic instance that you selected.

Purge Delete Tables Click to clear the data from the displayed delete tables.

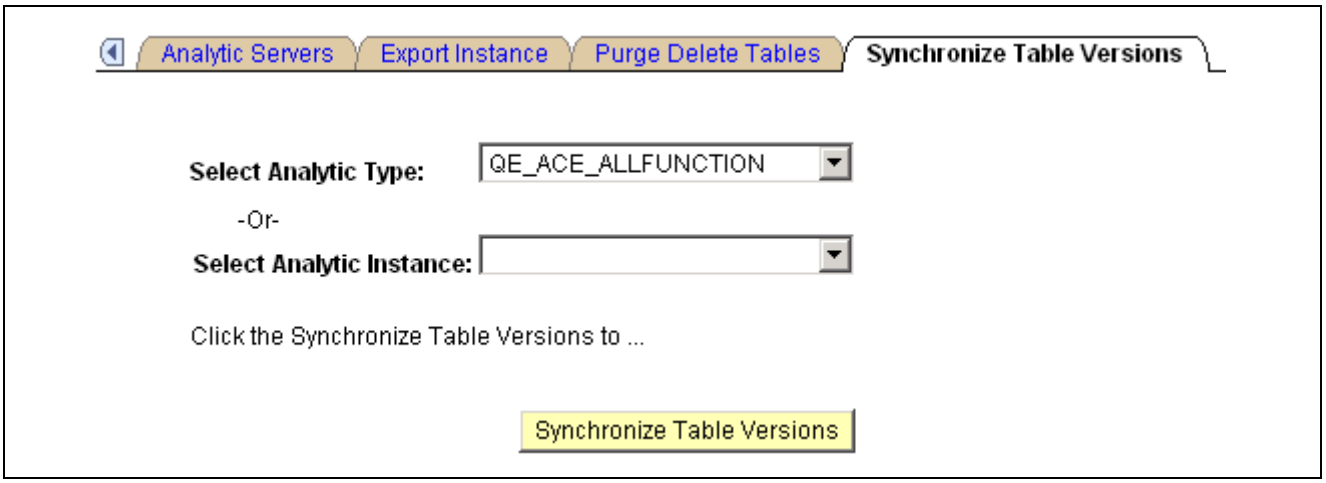
See Also

Enterprise PeopleTools 8.46 PeopleBook: Data Management, "Employing Database Level Auditing"

Synchronizing Table Versions

In a browser, select PeopleTools, Utilities, Administration, Analytic Server Administration, Synchronize Table Versions to access the Analytic Server Administration - Synchronize Table Versions page.

Note. Shut down all running analytic server processes before using this page.



Analytic Server Administration - Synchronize Table Versions page

Some scenario-managed optimization tables used with an analytic type have a version number field. The analytic server framework maintains a list of the tables and their version numbers. After an upgrade, the version numbers in the upgraded tables might not match the version numbers on this list. You use this page to update the list so the version numbers match.

Select Analytic Type -Or- Select Analytic Instance These drop-down lists are mutually exclusive. Select either an analytic type or an analytic instance for which you want to synchronize table versions.

Synchronize Table Versions When you use PeopleSoft Data Mover to move data from one database to another, it's often the case that the versions of analytic type or analytic instance data and the PSOPTSYNC table are out of synchronization. Click this button to synchronize the PSOPTSYNC table with the analytic instance tables.

Creating, Deleting, and Copying Analytic Instances

This section discusses how to:

- Create analytic instances.
- Delete analytic instances.
- Copy analytic instances.

Note. You can create, delete and copy analytic instances for use with both Analytic Calculation Engine and PeopleSoft Optimization Framework.

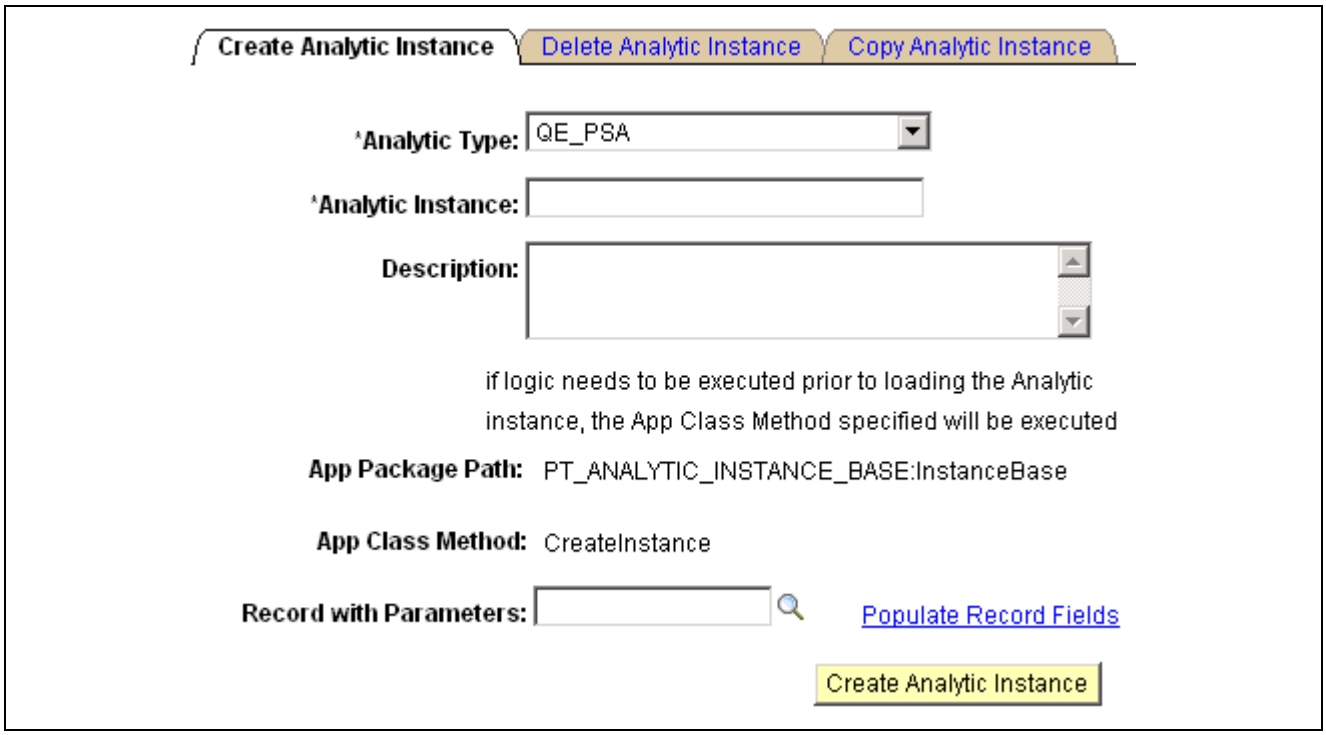
Pages Used to Create, Delete, and Copy Analytic Instances

Page Name	Object Name	Navigation	Usage
Create Analytic Instance	PTACECRTINST	PeopleTools, Utilities, Administration, Analytic Inst. Create/Del/Copy.	Create an analytic instance.
Delete Analytic Instance	PTACEDELINST	PeopleTools, Utilities, Administration, Analytic Inst. Create/Del/Copy, Delete Analytic Instance.	Delete an analytic instance.
Copy Analytic Instance	PTACECPYINST	PeopleTools, Utilities, Administration, Analytic Inst. Create/Del/Copy, Copy Analytic Instance.	Copy an analytic instance.

Creating Analytic Instances

Use the Create Analytic Instance page to create an analytic instance that you can then load and view within the Analytic Model Viewer to inspect and debug your analytic model.

Access the Create Analytic Instance page.



Create Analytic Instance page

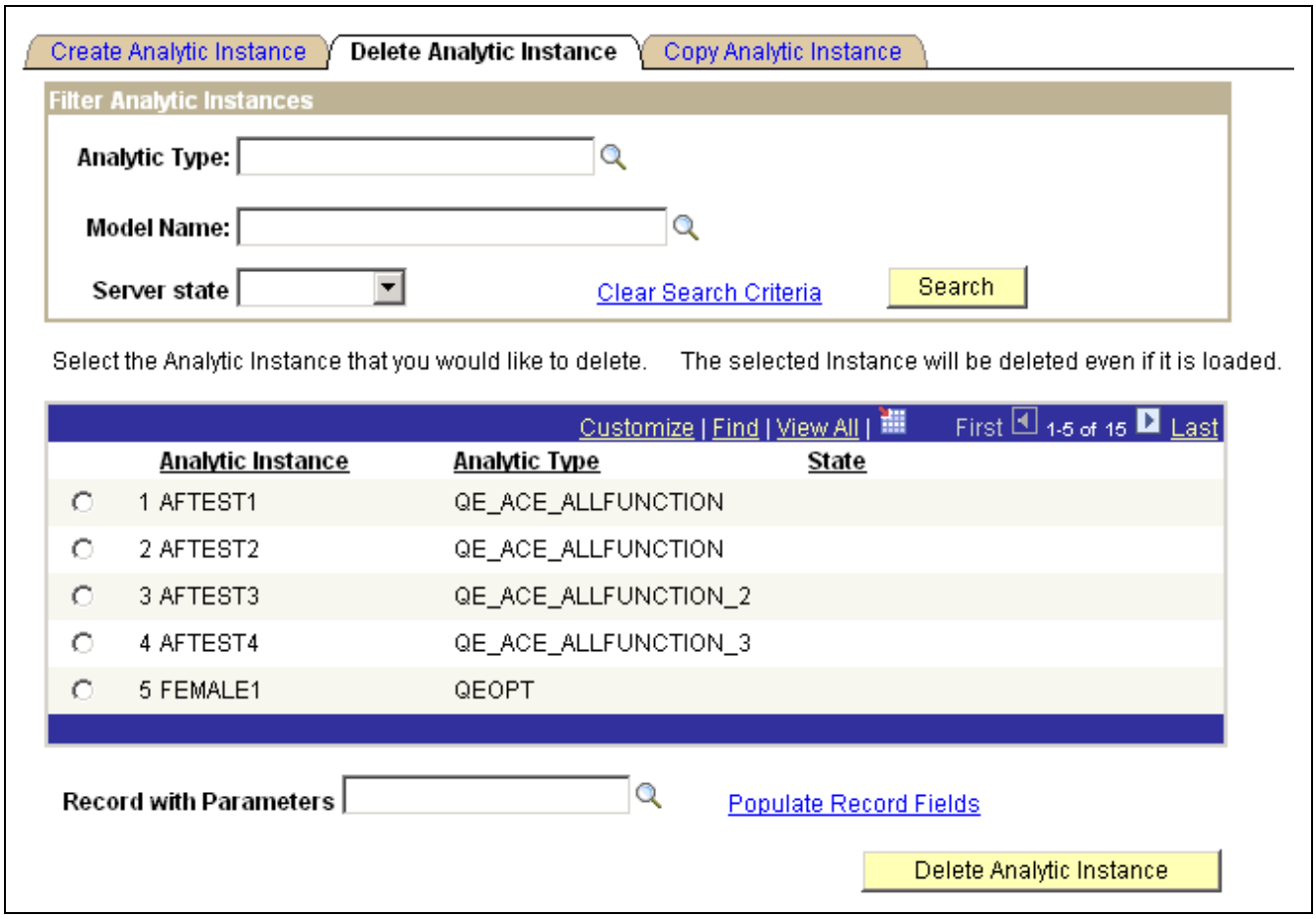
Analytic Type Select an analytic type from the drop-down list.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Analytic Calculation Engine*, "Working with Analytic Types".

	See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Optimization Framework*, "Designing Analytic Type Definitions," Creating Analytic Type Definitions.
Analytic Instance	Enter a name for the analytic instance. Analytic instance names should consist of alphanumeric characters, can be up to 20 characters long, and cannot include spaces.
App Package Path and **App Class Method**	The App Package Path field displays the full name of an application class (application package name, subpackage names if applicable, and class name) that's used to execute logic before loading the analytic instance. The App Class Method field displays the name of the method in the displayed class that's called at creation time to populate the new analytic instance with data. You establish the application package class and method to use when you define the analytic type. See *Enterprise PeopleTools 8.46 PeopleBook: PeopleCode Language Reference*, "PeopleCode Built-in Functions," CreateAnalyticInstance.
Record with Parameters	Look up and select parameters to be passed to the application class method. Click the lookup button to display a list of records. Selecting any record generates a standalone record.
Populate Record Fields	Displays a secondary page that lets you populate the fields of the standalone record; the values of these fields will serve as parameters passed into the App Class Method.
Create Analytic Instance	Create the analytic instance. After the analytic instance has been successfully created, you receive a notification to that effect.

Deleting Analytic Instances

Access the Delete Analytic Instance page.



Delete Analytic Instance page

You can specify search criteria to filter the display of returned analytic instances based on a combination of analytic type, model name, and server state.

Analytic Type Look up and select the analytic type upon which the analytic instance is based.

Model Name Look up and select the analytic model upon which the analytic instance is based.

Server State Select one of the following:

- *Idle*
- *Loading*
- *Executing*
- *Terminating*

See Chapter 5, "Managing Analytic Servers," Administering Analytic Servers, page 82.

Clear Search Criteria Click to delete any current search criteria so you can begin a search from scratch.

Search Click to display all analytic instances that meet the specified search criteria.

Select one of the displayed analytic instances to delete.

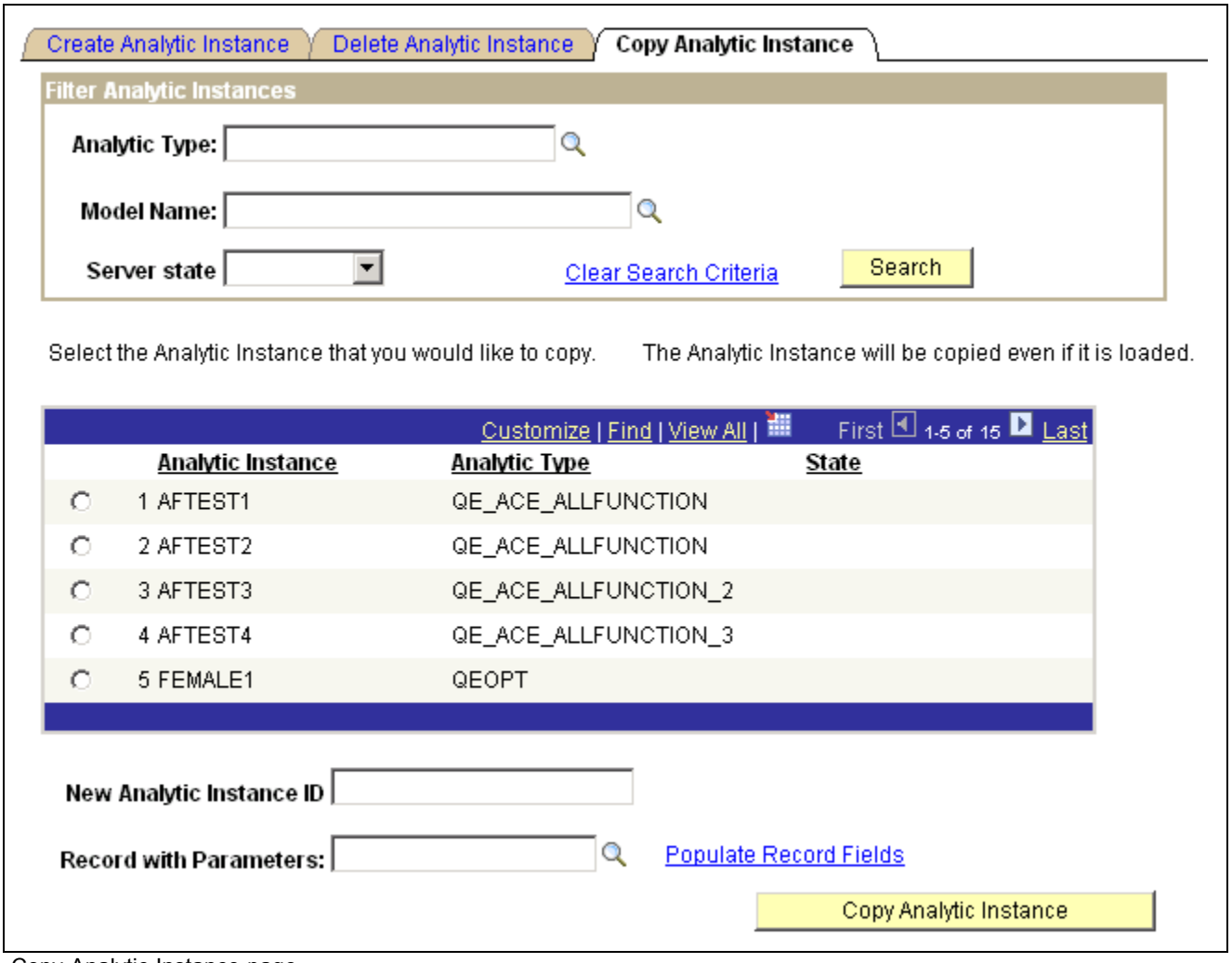
Record with Parameters Look up and select parameters to be passed to the application class method. Click the lookup button to display a list of records. Select a record, the first row of which will consist of parameters that you want to pass to the application class method. Selecting any record generates a populated, standalone record.

Populate Record Fields Displays a secondary page that lets you populate the fields of the standalone record; the values of these fields will serve as parameters passed into the App Class Method.

Delete Analytic Instance Unload the selected analytic instance and delete the data associated with it.

Copying Analytic Instances

Access the Copy Analytic Instance page.



Copy Analytic Instance page

You can specify search criteria to filter the display of returned analytic instances based on a combination of analytic type, model name, and server state.

Analytic Type Look up and select the analytic type upon which the analytic instance is based.

Model Name Look up and select the analytic model upon which the analytic instance is based.

Server State Select one of the following:

- *Idle*
- *Loading*
- *Executing*
- *Terminating*

See Chapter 5, "Managing Analytic Servers," Administering Analytic Servers, page 82.

Clear Search Criteria Click to delete any current search criteria so you can begin a search from scratch.

Search Click to display all analytic instances that meet the specified search criteria.

Select one of the displayed analytic instances as the source instance to copy.

New Analytic Instance ID Enter a name for the new analytic instance; this analytic instance will be a copy of the selected source instance.

Record with Parameters Look up and select parameters to be passed to the application class method that will copy the source analytic instance. Click the lookup button to display a list of records. Select a record, the first row of which will consist of parameters that you want to pass to the application class copy method. Selecting any record generates a populated standalone record.

Populate Record Fields Displays a secondary page that lets you populate the fields of the standalone record; the values of these fields will serve as parameters passed into the application class copy method.

Copy Analytic Instance Copy the selected analytic instance and its associated data. If a tree is attached to the selected analytic instance, all tree data is also copied to the new analytic instance, if all of the following are true:

Note. The analytic instance data and tree data are copied only if the record with parameters that you specified is populated with the source analytic instance ID.

Loading and Unloading Analytic Instances

This section discusses how to load and unload analytic instances:

Note. You can load and unload analytic instances for use with both the PeopleSoft Analytic Calculation Engine and PeopleSoft Optimization Framework.

Page Used to Load and Unload Analytic Instances

Page Name	Object Name	Navigation	Usage
Analytic Instance Load/Unload	PTACEMDLLOAD	PeopleTools, Utilities, Administration, Analytic Instance Load/Unload.	Load or unload an analytic instance.

Loading and Unloading Analytic Instances

To use the Analytic Model Viewer, you must load an analytic instance of the analytic model that you want to view or debug. You load analytic instances by using:

- The Analytic Instance Load/Unload page.
- The Analytic Model Viewer.

You unload instances by using the Analytic Instance Load/Unload page.

It is quicker to load an analytic instance by going through the Analytic Model Viewer: This approach allows you to simultaneously load and view the analytic instance. However, you can use the Analytic Instance Load/Unload page if you need to modify the timeout value or attach or detach a tree. You need to attach a tree before loading an analytic instance if you want to see the tree structure while reviewing this analytic instance within the Analytic Model Viewer.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Analytic Calculation Engine*, "Viewing and Debugging Analytic Models," Understanding Analytic Model Properties.

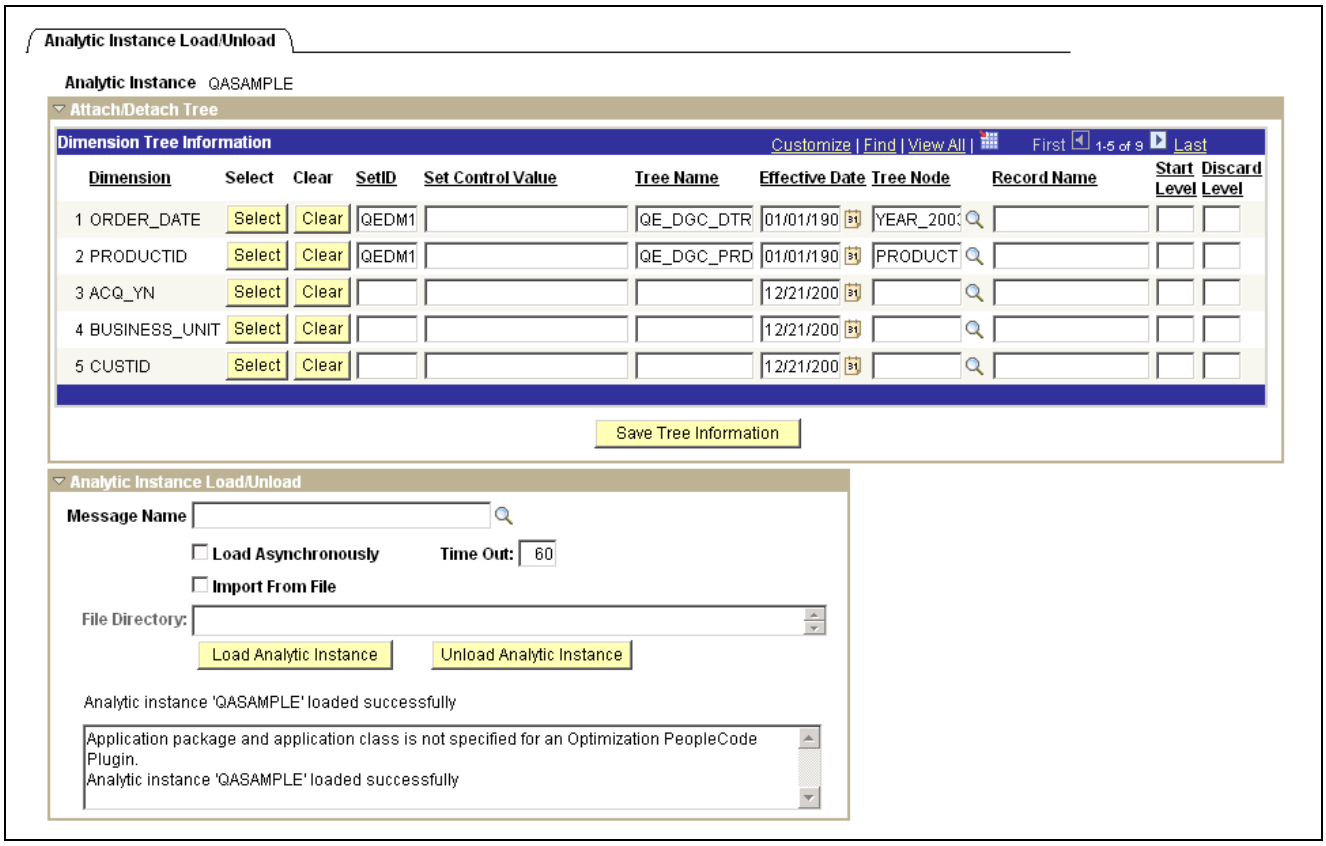
See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Analytic Calculation Engine*, "Creating Hierarchies".

Note. You can only load one analytic instance per analytic server.

You can also load and unload analytic instances by means of PeopleCode, using the AnalyticInstance class Load or Unload methods.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleCode API Reference*, "Analytic Calculation Engine Classes," AnalyticInstance Class Methods.

Access the Analytic Instance Load/Unload page by selecting the desired analytic instance on the search results page.



Analytic Instance Load/Unload page

Note. If you don't see the desired analytic instance, you need to create it as described earlier.

If you selected a PeopleSoft Optimization Framework analytic instance, you won't see the Attach/Detach Tree portion of the Analytic Instance Load/Unload page.

See Chapter 5, "Managing Analytic Servers," Creating, Deleting, and Copying Analytic Instances, page 87.

Attach/Detach Tree

Dimension Lists the dimensions in the selected analytic instance.

Select Click to select a tree to attach to the dimension. A secondary page appears from which you can select a tree.

Clear Click to disassociate a selected tree from the dimension.

SetID Displays the SetID associated with the tree, if applicable.

Set Control Value Displays the Set Control Value associated with the tree, if applicable.

Tree Name Displays the name of the selected tree.

Effective Date Displays the effective date associated with the tree.

Tree Node Specify a node from the selected tree.

Record Name Displays the name of a record containing override rules or functions.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Analytic Calculation Engine*, "Creating Hierarchies," Understanding the Relationship of PeopleSoft Trees to Analytic Models.

Start Level Enter a number to specify the type of dimension members that PeopleSoft Analytic Calculation Engine creates out of the nodes and leaves of a tree. The default value of this field is *0*. The root level is 1.

Note. If you specify a nonzero start level, you must specify the strictly enforced method to the tree in PeopleSoft Tree Manager. The strictly enforced method ensures that all members that are created out of one level are created as the same data type.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Analytic Calculation Engine*, "Creating Hierarchies," Purpose of Node Levels in Creating Hierarchies.

Discard Level Enter a number to specify the level from which PeopleSoft Analytic Calculation Engine does not attach any more of the tree to the dimension. PeopleSoft Analytic Calculation Engine does not create members out of nodes or leaves that are either at this level or lower than this level.

You must specify a start level to every tree for which you want to specify a discard level. The default value of this field is *0*. If you specify any other value, then it must be at a lower level (a higher number) than the start level.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Analytic Calculation Engine*, "Creating Hierarchies," Purpose of Node Levels in Creating Hierarchies.

Save Tree Information Click to save the dimension tree information that you've selected. The updated tree information takes effect the next time you load the analytic instance.

Analytic Instance Load/Unload

Message Name Specify an application message that should be sent if the analytic instance can't be unloaded successfully and is terminated prematurely. This can happen if the analytic server crashes while the analytic instance is loaded.

Note. The message is sent when the analytic server process restarts itself after crashing. The long edit box in this section of the page displays the content of detailed messages.

Load Asynchronously Select to indicate that the analytic instance should be run asynchronously.

Time Out Enter the number of minutes of inactivity before the analytic instance times out. The default time out is the value specified for the Analytic Instance Idle Timeout domain parameter.

See Chapter 4, "Setting Application Server Domain Parameters," Analytic Instance Idle Timeout, page 60.

Note. After an instance times out, you must reload it to continue working with it.

Import from File	Import an analytic instance from a file. You use this option to import an analytic instance that you've captured with the Analytic Instance Capture Utility. See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Analytic Calculation Engine*, "Capturing Analytic Instances".
File Directory	Specify the directory from which you want to retrieve the analytic instance that you are importing from file.
Load Analytic Instance	Click to load the selected analytic instance. PeopleSoft Analytic Calculation Engine displays a confirmation message after it successfully loads the analytic instance:
Unload Analytic Instance	Click to unload the selected analytic instance. PeopleSoft Analytic Calculation Engine displays a confirmation message after it successfully unloads the analytic instance. You must unload analytic instances once you're done working with them.

See Also

Enterprise PeopleTools 8.46 PeopleBook: PeopleCode API Reference, "Analytic Calculation Engine Classes," Load

CHAPTER 6

Working with BEA WebLogic

This chapter provides an overview of BEA WebLogic and discusses how to:

- Access the BEA WebLogic server console.
- Start BEA WebLogic.
- Stop BEA WebLogic.
- Use WebLogic server 8.1 console to monitor PeopleSoft sessions.
- Set up a reverse proxy server (RPS).
- Set up HTTP session timeout.
- Enable or disable HTTP keep alive.
- Change a WebLogic user's password.
- Implement WebLogic SSL keys and certificates.
- Restrict access to a servlet.
- Adjust the Java Virtual Machine (JVM) heap size.
- Determine the service pack level.
- Enable or disable the HTTP access log.

Understanding BEA WebLogic

This section discusses the PeopleSoft domain and the WebLogic session cookie name format

The PeopleSoft Domain

PeopleSoft Internet Architecture installation on BEA WebLogic Server 8.1 provides three primary server configuration options. Those options and their intended purpose are:

- Single server.

  This domain configuration contains one server named PIA, and the entire PeopleSoft enterprise application is deployed to it. This configuration is intended for single user or very small scale, non-production environments.

- Multi server.

  This domain configuration contains seven unique server definitions and a WebLogic cluster, and the PeopleSoft enterprise application is split across multiple servers. This configuration is intended for the production environment.

- Distributed managed server.

  This option is an extension of the “Multi server” selection and installs the necessary files to boot a managed server. This option requires a “Multi server” installation to be performed to some other location that contains the configuraiton for this managed server.

See Also

Appendix A, “BEA WebLogic 8.1 Managed Server Architecture,” page 279

WebLogic Session Cookie Name Format

When a user signs in to a PeopleSoft Pure Internet Architecture application, the portal servlet generates a cookie containing the user’s HTTP session ID, and sends it to the user’s browser to maintain the state of the session. The name of the cookie is fixed for all users accessing that portal.

On a WebLogic portal, the session cookie’s name is generated at install time based on the portal hostname and port number, which uniquely identify the portal within your PeopleSoft system. This name is stored in the portal’s weblogic.xml file.

However, the cookie name must not start with a number, and it must not contain any periods. If your users are experiencing problems signing in to PeopleSoft applications at different URLs from the same browser session, make sure that the session cookie names at those sites are valid.

To ensure valid WebLogic session cookie names:

1. Shut down your WebLogic server.
2. Open the weblogic.xml file for your web server in a text editor.

   You can find it in *PS_HOME*\webserv*domain_name*\applications\peoplesoft\PORTAL\WEB-INF.
3. Check the value of the session parameter called CookieName.

   Ensure that the content of the param-value element doesn’t start with a number or contain any periods. For example, the following session cookie name is invalid:

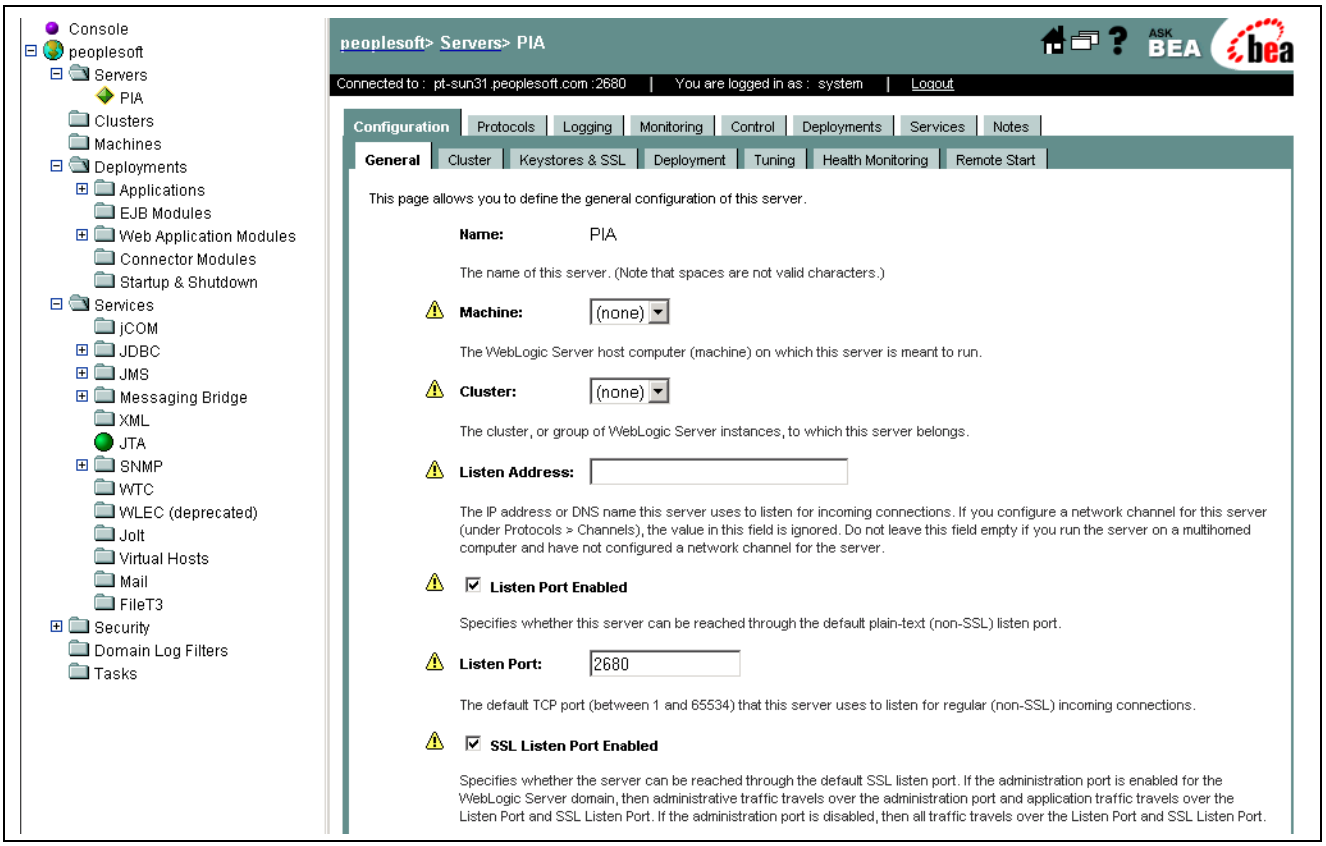
   ```
   <session-param>
     <param-name>CookieName</param-name>
     <param-value>57.28.208.21-80-WebLogicSession</param-value>
   </session-param>
   ```

   You can replace the periods with dashes (-). Following is a valid version of the session cookie name:

   ```
   <session-param>
     <param-name>CookieName</param-name>
     <param-value>c57-28-208-21-80-WebLogicSession</param-value>
   </session-param>
   ```
4. Save and close the file.
5. Restart your WebLogic server.

Accessing the BEA WebLogic Server Console

The BEA WebLogic Server console is the main utility that is used to administer and monitor the BEA WebLogic Server processes. Access the console by pointing your browser to http://*weblogic_servername*/console. Before the console opens, you will be prompted for the weblogic system ID and password that you specified during the PIA install. The default ID is *system* and the default password is *password.* After you are authenticated, the console appears. The BEA WebLogic server console provides an interface to monitor and tune aspects of a PeopleSoft application from a web server perspective.



BEA WebLogic server console

Starting BEA WebLogic

This section discusses how to:

- Start BEA WebLogic on Microsoft Windows.
- Start BEA WebLogic on UNIX.

See Also

Appendix A, “BEA WebLogic 8.1 Managed Server Architecture,” Administering a WebLogic Server Life Cycle, page 299

Starting BEA WebLogic on Microsoft Windows

To run BEA WebLogic Server on Microsoft Windows, you can use a Windows service or a foreground process.

Using the Command Prompt

Running BEA WebLogic as a foreground process is beneficial if you need to monitor WebLogic in real time. To run WebLogic as a foreground process, enter the following at the command prompt in the weblogic domain directory that the PIA install created for you (*PS_HOME*\webserv\peoplesoft\):

- Single server

```
startPIA.cmd
```

- Multi server
  - To start the weblogic domain admin server run `startWebLogicAdmin.cmd`
  - To start a managed server, such as PIA, run `startManagedWebLogic.cmd PIA`

Using the Windows Service

Two benefits of running BEA WebLogic as a Windows service are:

- BEA WebLogic can automatically start when the Windows server boots.
- You can start and stop the service from a remote Windows machine.

To install the service, enter the appropriate command from your weblogic domain directory at a command prompt as follows:

- Single server:

```
installNTservicePIA.cmd
```

- Multi server:

```
InstallNTservice.cmd weblogic_server_instance_name
```

For example:

```
installNTservice.cmd PIA
```

To start BEA WebLogic as a Windows service, use either of these methods:

- Start the service named *WebLogicdomain-servername* (for example, peoplesoft-PIA) by using the Services utility in the Windows Control Panel.
- Start the service from a command prompt by entering the following command:

```
NET START peoplesoft-PIA
```

Note. If WebLogic fails to start as a service, try starting it as a foreground process. To uninstall the service, enter the following command:

```
UninstallNTservicePIA.cmd
```

Starting BEA WebLogic on UNIX

To start PeopleSoft on UNIX execute the following script in the weblogic domain directory that the PIA install created for you (for example, *PS_HOME*/webserv/peoplesoft/):

- Single server

  ```
  startPIA.sh
  ```

- Multi server
  - To start the weblogic domain admin server run `startWebLogicAdmin.sh`
  - To start a managed server such as PIA, run `startManagedWebLogic.sh PIA`

To start the BEA WebLogic server, use the $WL_HOME/startWebLogic.sh script that is provided. This script sets some required environment variables and then starts a Java runtime environment to run BEA WebLogic within it. As delivered, this script starts the Java runtime environment, and, in effect, starts BEA WebLogic as a foreground process.

Stopping BEA WebLogic

For both Windows and UNIX, you can stop the PeopleSoft server from the BEA WebLogic Server console (http://localhost/console). To stop the PeopleSoft server:

1. Expand the peoplesoft domain.
2. Select Servers.
3. Right-click the name of the server to shut down.
4. Select Stop this server.

You can also stop the server through the command line by running:

- Single server

  ```
  stopPIA.cmd
  ```

  or

  ```
  stopPIA.sh
  ```

- Multi server

  ```
  stopWebLogic.cmd
  ```

  or

  ```
  stopWebLogic.sh
  ```

If WebLogic is running as a Windows service you can also stop it via the service in Windows Control Panel.

See Also

Appendix A, “BEA WebLogic 8.1 Managed Server Architecture,” Administering a WebLogic Server Life Cycle, page 299

Using WebLogic Server 8.1 Console to Monitor PeopleSoft Sessions

The WebLogic Server 8.1 console can display a list of established HTTP sessions for that instance of WebLogic Server. The steps below detail how to enable this level of monitoring for a WebLogic instance.

Note. The following describes how to enable session monitoring for the single server configuration of PIA. When in production, a multi server configuration would be used to perform these steps to the server instance that you intend to monitor, such as PIA1 or PIA2, or both.

1. Start the PIA server.

   Start the PIA server either via startPIA.cmd(.sh) or if installed as a Windows service, " NET START peoplesoft-PIA".

2. Log on to PeopleSoft

   Log on to your PeopleSoft application. If possible, log on from a couple different workstations using different PeopleSoft IDs. For the purpose of this test, do not log off.

3. Log on to the WebLogic Server Administrative Console.

   In a new browser, access the WebLogic Server console at http://localhost/console and specify the WebLogic administrative ID you specified during the PIA installation. The default ID and password are system/password, respectively.

4. Monitor established HTTP sessions for the PORTAL web application.

   In the graphical domain hierarchy on the left, use the following navigation to view the list of established HTTP sessions for the PORTAL web application:

   a. Expand ’peoplesoft’.

   b. Expand ’Deployments’.

   c. Expand ’ Applications’.

   d. Expand ’peoplesoft’.

   e. Select ’PORTAL’.

   f. Click the ’Monitoring’ tab.

   g. Click the ’Sessions’ tab.

      If ’Session Monitoring Enabled’ is not selected, then Session monitoring has been disabled on this WebLogic instance. To re-enable session monitoring simply re-check that box, click ’Apply’ and restart WebLogic Server.

      With HTTP session monitoring enabled, the information displayed will be client’s PeopleSoft ID, IP address and PeopleSoft site they are logged onto.

Main Attribute	Time Last Accessed	Server	Name
QEDMO@216.131.220.97/ps	Tue Oct 21 13:55:38 PDT 2003	PIA	1VdBHOc5OHx2kQTxioEBbRtGDY1KE8wH! 1066769729489
QESS@216.131.220.72/ps	Tue Oct 21 13:56:43 PDT 2003	PIA	1VdpACQ35gN8r4VOrJ1Uo8m83NL2e7AM! 1066769769778

Monitor WebLogic server console

Note. An established HTTP session remains on the web server until the client logs off of PeopleSoft or until their HTTP session times out. Simply closing the browser doesn't log a PeopleSoft user off. As a result, when user closes the browser without logging out of the PeopleSoft session, the corresponding HTTP session remains on the webserver until it times out. HTTP session timeouts are controlled via thet site's Web Profile in the PeopleSoft database.

See Also

Appendix A, "BEA WebLogic 8.1 Managed Server Architecture," Tuning Performance and Monitoring Resources, page 303

Setting Up an RPS

PeopleSoft applications support the use of reverse proxy servers (RPS) with BEA WebLogic. An RPS supplies the URL to which the browsers connect, but a backend web server handles the transaction processing.

This section discusses how to:

- Configure Microsoft Internet Information Server (IIS) as an RPS.
- Configure BEA WebLogic as an RPS.
- Configure Sun iPlanet as an RPS.
- Use the iPlanet plug-in.
- Configure Apache HTTP as an RPS.

Configuring Microsoft IIS as an RPS

This section describes how to proxy content to a single server configuration of PIA. When in production, a multi server configuration would be used to perform these steps to proxy content to your managed server instance of PIA or PIA1, etc.

Microsoft Internet Information Server (IIS) can be configured as a reverse proxy server (RPS) to one or more WebLogic Server 8.1 instances. Multiple instances can be independent instances or grouped into a cluster. When you use a reverse proxy, any URL that would be used to access your PeopleSoft application (even URLs that are stored in the database) would point to the reverse proxy, and *not* to the WebLogic Server.

These instructions are based on a logical separation of BEA WebLogic Server and Microsoft IIS, where both web servers are installed on the same machine. If your configuration has BEA WebLogic Server and Microsoft IIS on separate machines, you must perform three additional steps. Those steps are:

- From the BEA WebLogic server, copy c:\bea\weblogic81\server\bin\iisproxy.dll to c:\inetpub on your Microsoft IIS server.
- From the BEA WebLogic server, copy c:\bea\weblogic81\server\bin\iisforward.dll to c:\inetpub on your Microsoft IIS server.
- In the following procedure, change any reference from c:\bea\weblogic81\server\bin to c:\inetpub.

To set up a Microsoft IIS RPS:

1. Install the PeopleSoft Internet Architecture.

   Run the multiplatform PeopleSoft Internet Architecture install from %PS_HOME%\setup\mpinternet \setup.exe.

2. Access the Microsoft IIS configuration.

   On a Microsoft Windows server, select Start, Programs, Administrative Tools, Internet Services Manager.

   Note. Windows workstation and Windows 2000 Professional are not supported.

3. Open the Default Web Site properties

   Expand your list of available servers, right click the Default Web Site and select Properties.

4. Add an ISAPI filter.
   - Select the ISAPI Filters tab, and click Add to define a new filter.
   - Enter IISFORWARD for the filter name.
   - Enter c:\bea\weblogic81\server\\bin\iisforward.dll for the executable.
5. Define a new application extension mapping.
   - Select the Home Directory tab then click Configuration.
   - Click Add on the App Mapping tab to define a new application mapping.
   - Enter c:\bea\weblogic81\server\bin\iisproxy.dll for the executable.
   - Enter .wlforward for the extension.
   - For Verbs, enter All Verbs (or at a minimum, GET and POST).
6. Create the IIS-Plugin configuration file.

   Create c:\bea\weblogic81\server\\bin\iisproxy.ini, containing the following lines and setting the values appropriately.

```
#
#For a list of available parameters see
#http://edocs.bea.com/wls/docs81/plugins/index.html
#
WebLogicHost=<hostname or IP of weblogic server to forward requests to>
WebLogicPort=<HTTP port of weblogic server to forward requests to>
DebugConfigInfo=OFF
Debug=OFF
#
#To proxy all IIS directed requests to WebLogic set "WlForwardPath=/"
#To selectively proxy only PeopleSoft requests to WebLogic set "WlForwardPath="to
#the list of PeopleSoft sites to proxy.
```

```
#e.g. To proxy requests for only 'ps' and 'crm' set WlForwardPath to the following;
#WlForwardPath=*/ps/*,*/crm/*
WlForwardPath=/
#
#If you have specified an AuthTokenDomain during your PIA installation,
#you must set the cookieName for your reverse proxy.
#CookieName=<CookieName as specified on weblogic in PORTAL webapps's weblogic.xml>
```

7. Restart Microsoft IIS.

   Restart the two Windows services, IIS Admin Service and World Wide Web Publishing Service by using the Services utility in the Control Panel or by issuing the following three commands at a command prompt:

   ```
   NET STOP IISADMIN /Y
   NET START IISADMIN
   NET START W3SVC
   ```

8. Start the BEA WebLogic server.

   Start the PeopleSoft Internet Architecture server either by invoking startPIA.cmd (.sh) or if installed as a Windows service, "NET START peoplesoft–PIA".

   See Chapter 6, "Working with BEA WebLogic," Starting BEA WebLogic, page 99.

   See Chapter 6, "Working with BEA WebLogic," Stopping BEA WebLogic, page 101.

9. Test your configuration by accessing the Microsoft IIS server by using the URL for your site.

   For example, http://*IIS_server*:*port*/ps/signon.html.

Note. To connect to Microsoft IIS by using HTTPS, you must install digital certificates on the Microsoft IIS server.

See Also

"BEA documentation for IIS-plugin, " http://e-docs.bea.com/wls/docs81/plugins/isapi.html

"BEA documentation for IISPROXY.INI parameters, " http://e-docs.bea.com/wls/docs81/plugins/plugin_params.html

Configuring BEA WebLogic as an RPS

This section discusses how to configure a BEA WebLogic server as a reverse proxy server (RPS).

Creating the RPS

To create an RPS, select *Multi Server Domain* as the configuration to install during PIA setup. As a result, a server named "RPS" is automatically defined in addition to the main PIA server, and is configured to be a reverse proxy server to other managed servers. By default, the following settings are applied to the RPS:

Setting	Value
Name	RPS
HTTP Listen Port	8080

Setting	Value
HTTPS Listen Port	8443
Default web application	HttpProxyServlet
Address of back-end WebLogic content server	The hostname of the machine from which the PIA setup was run, with the HTTP listen port specified during the PIA setup.

The default address specified for the back-end WebLogic content server assumes that it's the same machine as the one on which you're configuring the RPS, using the HttpProxyServlet application. There's no need to change this setting unless the content server is a different machine, or you enable load balancing with multiple content servers. If it's a different machine, you must change this setting to specify the correct content server. If you enable load balancing, you'll need to specify additional content servers.

Enabling Load Balancing

In addition to the *HttpProxyServlet* application, the PIA setup also defines an *HttpClusterServlet* application in your WebLogic configuration, which by default isn't active. The primary difference between the two applications is that for a given HTTP request, HttpProxyServlet can proxy content only from a single back-end content server, whereas HttpClusterServlet can proxy content from multiple back-end content servers, all of which serve the same content. This enables the RPS to load-balance the requests across a cluster of WebLogic servers.

You can configure the RPS for load balancing by changing the default web application from HttpProxyServlet to HttpClusterServlet, which becomes active as a result.

To change the default web application:

1. Start the WebLogic server.
2. Sign in to the WebLogic administration console.
3. Navigate to Deployments, Web Application Modules, HttpProxyServlet.
4. Select the Targets tab.
5. Clear the RPS Server check box, then click Apply.
6. Navigate to Deployments, Web Application Modules, HttpClusterServlet.
7. Select the Targets tab.
8. Select the RPS Server check box, then click Apply.
9. Sign out of the WebLogic administration console.

Specifying Back-End WebLogic Content Servers

You need to specify back-end WebLogic content servers only for the currently designated default web application (HttpProxyServlet or HttpClusterServlet).

You can edit the appropriate web.xml configuration file directly, or you can use the WebLogic Builder software provided with BEA WebLogic. Refer to your BEA WebLogic documentation for details about WebLogic Builder. To edit the configuration file directly:

- For the HttpProxyServlet application —

You need to change this setting only if the back-end WebLogic content server is on a different machine than the one where you're configuring the RPS. Edit the web.xml configuration file in *PS_HOME*\webserv*weblogic_domain*\applications\HttpProxyServlet\WEB-INF.

Modify the param-value elements for the *WebLogicHost* parameter and the *WebLogicPort* parameter to specify the hostname and HTTP listen port, respectively, of the back-end content server.

- For the HttpClusterServlet application —

  Edit the web.xml configuration file in *PS_HOME*\webserv*weblogic_domain*\applications\HttpClusterServlet\WEB-INF.

  Modify the param-value element for the *WebLogicCluster* parameter to specify multiple back-end content servers separated by "|" symbols, using the following format:

  ```
  host1:http_port:https_port|host2:http_port:https_port
  ```

Starting the RPS

To start the RPS, open a command prompt, change to *PS_HOME*\webserv*weblogic_domain*, and launch the following commands:

1. startWebLogicAdmin
2. startManagedWebLogic RPS

Note. You can also run the RPS as a service on Windows.

See Also

Chapter 6, "Working with BEA WebLogic," Starting BEA WebLogic, page 99

Chapter 6, "Working with BEA WebLogic," Stopping BEA WebLogic, page 101

"BEA documentation for WebLogic Proxy, " http://e-docs.bea.com/wls/docs81/plugins/http_proxy.html

"BEA documentation for proxy parameters, " http://e-docs.bea.com/wls/docs81/plugins/plugin_params.html

Configuring Sun iPlanet as an RPS

This section describes how to proxy content to a single server configuration of PIA. When in production, a multi server configuration would be used to perform these steps to proxy content to your managed server instance of PIA or PIA1.

Sun's iPlanet web server can be installed and configured as a reverse proxy to WebLogic Server. BEA has certified different version of iPlanet web server version on different OS platforms. PeopleSoft extends that certification list to its customer.

See http://e-docs.bea.com/platform/suppconfigs/index.html

See http://e-docs.bea.com/wls/docs81/plugins/nsapi.html

To configure iPlanet as an RPS:

1. Download iPlanet Web Server, Enterprise Edition.

   Download and install a BEA certified platform/version of iPlanet Web Server from Sun.

   See http://www.sun.com/software/products/web_srvr/home_web_srvr.html

2. Install WebLogic iPlanet plug-in.

Note. If you are going to run iPlanet on the same machine as WebLogic, it is recommended to skip this copy step.

- On Microsoft Windows machines, copy *WebLogic_home*\weblogic81\server\\bin*shared_library* to *iPlanet_dir*\plugins.
- On UNIX machines, copy *WebLogic_home*/weblogic81/server/lib/*iPlanet_platform*/*shared_library* to *iPlanet_dir*/plugins.

  See http://e-docs.bea.com/platform/suppconfigs/index.html

WebLogic_home refers to the root directory of your WebLogic installation.

iPlanet_dir refers to the location where iPlanet is installed. For iPlanet 4.x on Windows, the default is c:\netscape\server4\. For iPlanet 6.x on Windows, the default is C:\iPlanet\servers\

iPlanet_platform refers to the OS platform on which BEA has certified iPlanet.

shared_library refers to the iPlanet plugin library that BEA provides with WebLogic.

3. Define the NSAPI Module

   Be sure to backup your obj.conf before you begin this step. This step covers modifying the iPlanet configuration file, obj.cont, (magnus.conf for iPlanet (6.x) so as to reference the BEA provided NSAPI module.

   Following are examples using configuration files on a Windows machine named crm.peoplesoft.com.

   - For iPlanet 4.x:

     Edit the configuration file C:\Netscape\Server4\https-crm.peoplesoft.com\config\obj.conf for your iPlanet instance.

     Add the following lines to the top of the obj.conf file, preceding any comments. This instructs iPlanet to load the native library as an NSAPI module. For *iPlanet* and *drive*, substitute the actual location, including the drive letter of the NSAPI module you copied in at previous steps:

```
Init fn="load-modules" funcs="wl-proxy,wl-init"\
 shlib=drive:/iPlanet/plugins/proxy36.dll
Init fn="wl-init"
```

     If you skipped Step 1 because iPlanet and WebLogic will be running on the same machine, update your configuration file similar to the following:

```
Init fn="load-modules" funcs="wl_proxy,wl_init"\
 shlib="drive:/WebLogic_home/weblogic81/server/bin/proxy36.dll"
Init fn="wl_init"
```

- For iPlanet 6.x:

  iPlanet 6.x -

  Edit the configuration file C:\iPlanet\server\https-crm.peoplesoft.com\config\magnus.conf for your iPlanet instance.

  Add the following lines to the bottom of the magnus.conf file. This instructs iPlanet to load the native library as an NSAPI module. For *iPlanet* and *drive*, substitute the actual location, including the drive letter of the NSAPI module you copied in at previous steps:

  ```
  Init fn="load-modules" funcs="wl-proxy,wl-init"\
   shlib=drive:/iPlanet/plugins/proxy36.dll
  Init fn="wl-init"
  ```

  If you skipped Step 1 because iPlanet and WebLogic are running on the same machine, update your configuration file similar to the following:

  ```
  Init fn="load-modules" funcs="wl_proxy,wl_init"\
   shlib="drive:/WebLogic_home/weblogic81/server/bin/proxy36.dll"
  Init fn="wl_init"
  ```

4. Define which requests to be handled by the plug-in.

   The type of requests to be handled by the iPlanet plug-in, and subsequently handed off to BEA WebLogic, must be declared as part of an object definition in the obj.conf file. A specific string in the URL, referred to as a *ppath*, can identify these requests.

   To proxy all requests of a single PeopleSoft Internet Architecture site, such as ps (which would be accessed as http://crm.peoplesoft.com/ps/signon.html), define the following object tag in the obj.conf file. Define this and any other object tags directly following the default object tag.

   ```
   <Object name="ps" ppath="*/ps/*">
   Service fn=wl-proxy WebLogicHost=server1\
    WebLogicPort=7001
   </Object>
   ```

   The default object tag is generally several lines long and can be identified by `<Object name=default>...</Object>`.

   To proxy additional sites, add subsequent object tags referencing the other site names:

   ```
   <Object name="hr" ppath="*/hr/*">
   Service fn=wl_proxy WebLogicHost=server1\
    WebLogicPort=7001
   </Object>
   ```

   To proxy all requests that are made to iPlanet, create a single object tag named “peoplesoft” and set the ppath parameter to *.

5. Apply changes to iPlanet

   With these settings saved, access the iPlanet server manager, perhaps http://localhost:8888. Supply the ID and password that you specified during the iPlanet install. The default ID/password is admin/password. When prompted, click the Apply button to update iPlanet with your changes and restart it.

6. Start WebLogic Server.

   Start the PIA server either via starPIA.cmd(.sh) or if installed as a Windows service, “NETSTART peoplesoft-PIA.

See Chapter 6, "Working with BEA WebLogic," Starting BEA WebLogic, page 99.

See Chapter 6, "Working with BEA WebLogic," Stopping BEA WebLogic, page 101.

7. Confirm the configuration.

   To confirm an installation, with both the WebLogic Server and iPlanet servers started, simply access PeopleSoft using the typical URL, http://*iPlanet*/ps/signon.html. If you are able to logon to PeopleSoft, your installation and configuratiion was successful.

   See http://edocs.bea.com/wls/docs81/plugins/index.html

Applying Changes to iPlanet

After saving settings, access the iPlanet server manager (for example, http://localhost:8888).

Enter the ID and password that you specified during the iPlanet installation. The default ID and password are admin and password. When prompted, click Apply to update iPlanet with your changes and restart it.

Starting the Server and Confirming the Installation

Start the PIA server with either `startPIA.cmd(.sh)` or, if installed as a Microsoft Windows service,`NET START peoplesoft-PIA.`.

To confirm an installation, with both the BEA WebLogic server and iPlanet servers started, access the PeopleSoft system by using the typical URL, http://*iPlanet*/ps/signon.html. If you can sign in to the PeopleSoft system, your installation and configuration was successful.

Using the iPlanet Plug-in

If you plan to proxy all requests for the PeopleSoft Internet Architecture through iPlanet, you must also update any URLs that are defined in the PeopleSoft database to reference the iPlanet server, not the BEA WebLogic server.

Those URLs are:

- For the PeopleSoft portal, any content URLs that you have defined that directly reference PeopleSoft content (psc, psp) on the BEA WebLogic server (meaning that the BEA WebLogic server is referenced in the URL) must be updated to reference the iPlanet server in the URL
- For the PeopleSoft integration gateway, any node definitions that directly reference an integration gateway on the BEA WebLogic server must be updated to reference the iPlanet server in the URLs.
- For the PeopleSoft report repository, any report node definitions that directly reference a report server on the BEA WebLogic server must be updated to reference the iPlanet server in the URLs.
- Any of your own definitions or objects that reference the URL of the BEA WebLogic server must be updated to reference the iPlanet server in the URLs.

The iPlanet obj.conf file is strict about the placement of text. To avoid problems, follow these guidelines:

- Eliminate extraneous leading and trailing white space.

  If you must enter more characters than can be fit on one line, place a backslash (\) at the end of that line and continue typing on the following line. The backslash directly appends the end of the first line to the beginning of the following line.

- If a space is necessary between the words that end the first line and begin the second line, use one space, either at the end of the first line (before the backslash), or at the beginning of the second line.
- Do not split attributes across multiple lines.

The BEA online documentation contains a complete listing of BEA WebLogic plug-in attributes and parameters.

See http://edocs.bea.com/wls/docs81/plugins/index.html

iPlanet Plug-in Considerations

If you intend to proxy all requests for PeopleSoft Internet Architecture through iPlanet, you must also update any URLs defined in your PeopleSoft database to reference the iPlanet server, not the WebLogic Server. Those URLs are:

- For PeopleSoft Portal, any content URLs that you have defined that directly referene PeopleSot content (psc, psn) on your WebLogic Server directly (meaning that the WebLogic Server is referenced in the URL) must be updated to reference your iPlanet server in the URL.
- For PeopleSoft Application Messaging Gateway, any Node definitions that directly reference an Application Messaging Gateway on your WebLogic Server must be updated to reference your iPlanet server in the URLs.
- For PeopleSoft Report Repository, any Report Node definitions that directly reference a Report Server on your WebLogic Server must be updated to reference your iPlanet server in the URLs.
- For any custom definition or objects that reference the URL of your WebLogic Server must be updated to reference your iPlanet server in the URLs.

The iPlanet obj.conf file is strict about the placement of text. To avoid problems, be certain that you adhere to the following guidelines:

- Eliminate extraneous leading and trailing white space.

  If you must enter more characters than can be fit on one line, place a backlash \ at the end of that line and continue typing on the following line. The backslash directly appends the end of the first line to the beginning of the following line. If a space is necessary between the words that end the first line and begin the second line, be certain to use one space, either at the end of the first line (before the backslash), or at the beginning of the second line.
- Attributes must not be split across multiple lines.

Configuring Apache HTTP as an RPS

This section describes how to proxy content to a single server configuration of PIA. When in production, a multi server configuration would be used to perform these steps to proxy content to your managed server instance of PIA or PIA1, etc.

Apache HTTP server can be installed and configured as a reverse proxy server to WebLogic Server. For a list of certified platforms,

See http://e-docs.bea.com/platform/suppconfigs/index.html

To configure Apache HTTP:

1. Download the Apache HTTP server.

   See http://www.apache.org/dist/httpd/.
2. Install Apache.

   See http://httpd.apache.org/docs-project/.
3. Install the Apache HTTP server plug-in.

The installation of the Apache plug-in from BEA depends on whether you are installing the plug-in as a dynamic shared object (DSO) or a statically linked module. If you have downloaded the binary distribution of Apache, you will probably install the Apache plug-in from BEA as a shared object. (If you are in doubt as to which type, install the plug-in as a DSO.) Exact instructions are available from BEA.

See http://e-docs.bea.com/wls/docs81/plugins/apache.html

4. Specify the parameters that will be used by the Apache plug-in by defining them in an IfModule tag for BEA WebLogic in the Apache httpd.conf file.

   Add this tag in the ### Section 2: 'Main' server configuration section of httpd.conf. For example, to configure the Apache to proxy all requests that it receives to a BEA WebLogic server that is running on a machine named crm.peoplesoft.com and listening on port 7001, you would define the following tag:

```
<IfModule mod_weblogic.c>
   WebLogicHost crm.peoplesoft.com
   WebLogicPort 7001
   MatchExpression /</IfModule>
```

   BEA provides sample and template configuration files.

   See http://edocs.bea.com/wls/docs81/plugins/index.html

   To proxy requests to a cluster of BEA WebLogic servers, replace the two attributes, WebLogicHost and WebLogicPort, with WebLogicCluster.

   The syntax of the WebLogicCluster is `wlserver1:port,wlserver2:port`.

   Details about clustering setup are available in a red paper.

   See The red paper on the PeopleSoft Customer Connection website: *Clustering and High Availability for PeopleSoft 8.4*

   If you specified an AuthTokenDomain during the PeopleSoft Internet Architecture installation, you must set the cookieName for the reverse proxy to that same value. To do so, add the cookieName attribute and set its value to CookieName, as specified on the BEA WebLogic server in the PORTAL web application's weblogic.xml file (for example, c:\bea\wlserver6.1\config\peoplesoft\applications\PORTAL\web-inf\weblogic.xml).

5. Start the Apache HTTP server following the Apache usage instructions.
6. Start the BEA WebLogic server with either `startPIA.cmd(.sh)` or, if installed, as a Microsoft Windows service, `NET START peoplesoft-PIA`.
7. To confirm an installation, with both the BEA WebLogic server and Apache servers started, access the PeopleSoft system by using the typical URL, http://*Apache*/ps/signon.html.

   If you can sign in to the PeopleSoft system, your installation and configuration was successful.

See Also

http://edocs.bea.com/wls/docs81/plugins/index.html

Setting Up HTTP Session Timeout

HTTP session timeout controls are accessible on the Security page of the web profiles in the PeopleSoft database. PeopleSoft Internet Architecture no longer uses session timeout control set on the web server. The session timeouts set in the Web Profiles override any HTTP session timeouts set on the webserver at runtime.

See *Enterprise PeopleTools 8.46 PeopleBook: Internet Technology*, "Configuring the Portal Environment," Configuring Portal Security.

Enabling or Disabling HTTP Keep Alive

This section describes how to change HTTP Keep-Alive settings for a single server configuration of PIA. When in production, a mult server configuration would be used to perform these steps to your managed server instance of PIA, PIA1, etc.

Keep-Alive, or more accurately termed "Persistent Connections" is a default feature of HTTP 1.1 as described in http://www.w3.org/Protocols/rfc2616/rfc2616.html. Keep-Alive allows for the client (generally a web browser) and the web server to maintain open connections between requests for specified period of time. That time period is generally less then 60 seconds. The benefit of a persistent connection is that with each subsequent request the client and the server do not need to perform the overhead of opening a new connection. Enabling keep-Alive is generally recommended, but in some situations it may introduce a problem. Sporadic "The Page cannot be displayed" can be the result of a problem with keep-Alive. In situations where keep-Alive issues are suspected, disabling the web server keep-Alive will help to determine if the problem is indeed related to connection persistence.

To enable or disable Keep-Alive:

1. Start the PIA server.

   Start the PIA server either via startPIA.cmd(.sh) or if installed as a Windows service, " NET START peoplesoft-PIA".

   See Chapter 6, "Working with BEA WebLogic," Starting BEA WebLogic, page 99.

   See Chapter 6, "Working with BEA WebLogic," Stopping BEA WebLogic, page 101.

2. Log on to the WebLogic Server Administrative Console.

   In a new browser, access the WebLogic Server console at http://localhost/console and specify the WebLogic administrative ID that you specified during the PIA installation. The default ID and password are system and password, respectively.

3. Open Server's HTTP configuration page.

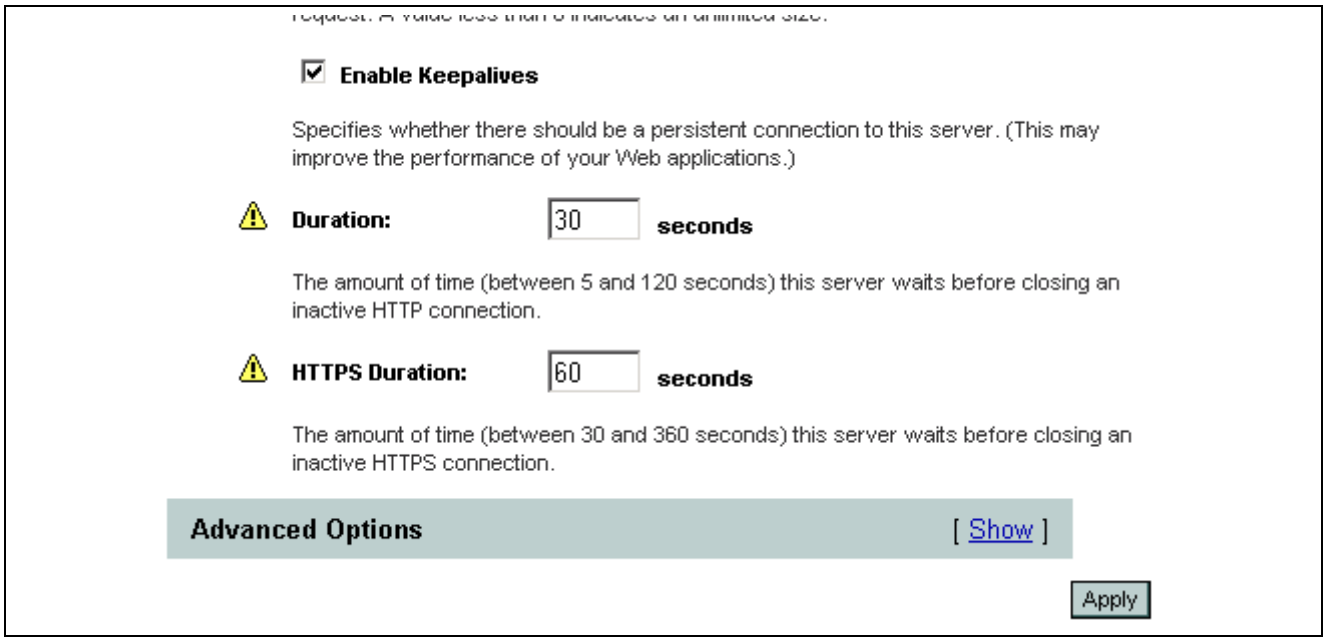
   In the navigation window on the left,use the following navigation to open the PIA server's HTTP configuration settings. (If you are using a custom server name, substitute that name where appropriate.):

   a. Expand 'peoplesoft'.
   b. Expand 'Servers'.
   c. Select 'PIA'.
   d. Click the 'Protocols' tab.

4. Change keep-alive settings.

- To disable keep-Alive: Uncheck "Enable Keepalives" and click ’Apply’ . With keep-Alive disabled, HTTP keep-Alive Duration and HTTPS keep-Alive Duration are not used.
- To enable keep-Alive: Check "Enable Keepalives" and update HTTP Keep-Alive ’Duration’ and ’HTTPS Keep-Alive Duration’ values as deemed necessary. Once done click ’Apply’. Minimum/maximum values for HTTP are 5/120 seconds respectively. For HTTPS the minimum/maximum values are 120/360 seconds.

5. Restart WebLogic Server.



Keep-Alive page

Changing a WebLogic User’s Password

The WebLogic domain built by the PIA install includes 3 WebLogic user IDs. Those IDs are: system, operator and monitor. Each of those IDs have a default password of ’password’. It is *highly*recommended to change this password on any production or critical servers.

To change the password for the system:

1. Start the PIA server.

   Start the PIA server either via *PS_HOME*\webserv*weblogic_domain*\startPIA.cmd(.sh) or if installed as a Windows service, " NET START peoplesoft-PIA".

   See Chapter 6, “Working with BEA WebLogic,” Starting BEA WebLogic, page 99.

   See Chapter 6, “Working with BEA WebLogic,” Stopping BEA WebLogic, page 101.

2. Log in to the WebLogic Server Administrative Console.

   Access the WebLogic Server console at http://*webserver*/console (for example, http://localhost/console). When prompted for a user name and password, specify the WebLogic system ID and password. If you’ve followed the default WebLogic Server install, the ID and password are ’system’ and ’password’. Otherwise, specify the password supplied during your PIA installation.

3. Change a WebLogic Server user's password.

   In the graphical domain hierarchy on the left, use the following navigation path to change a user's password.

   a. Expand 'peoplesoft'.

   b. Expand 'Security'.

   c. Expand 'Realms'

   d. Expand 'myrealm' or your custom realm if you have created one.

   e. Select 'Users'.

   f. Select the user whose password you would like to change, in this case that is 'system'.

   g. Click the Change button.

   h. Enter and re-enter a new password for this user.

   i. Click the Apply button.



WebLogic Console page

Extra Step for Running WebLogic as a Windows Service

When you run WebLogic as a Windows service, WebLogic uses the default ID or 'operator' and its password of 'password'. Changing the password for the WebLogic ID that runs the Windows service requires an additional manual step. That step is to update setEnv.cmd (for example, *PS_HOME*\webserv\peoplesoft\setEnv.cmd) and set the WLS_PW environment variable to 'operator's new password. Once that is done, reinstall the Windows service by re-running the installNTservice command file located in the same WebLogic domain directory as the setEnv.cmd that you edited.

Implementing WebLogic SSL Keys and Certificates

This section provides an overview of Secure Sockets Layer (SSL) encryption with WebLogic 8.1 and discusses how to:

- Obtain encryption keys.
- Prepare keys and certificates for the keystore.
- Import keys and certificates into the keystore.
- Configure WebLogic SSL encryption keys.

Understanding SSL Encryption with WebLogic 8.1

To use SSL encryption with WebLogic 8.1 and the current PeopleTools release, the WebLogic keystore must contain the following appropriately configured encryption keys:

- The web server's private key.
- The web server's public key, digitally signed by a trusted *certificate authority* (CA).
- The digitally signed public key of the same CA that signed the web server's key.

A public key is transferred and stored as a data element in a digital certificate or a *certificate signing request* (CSR). You can obtain public keys from a variety of sources, in several different formats.

You must ensure that the encryption keys are correctly formatted, install them in the keystore, then configure them using the WebLogic server administration console.

Note. If you've already installed and configured a set of encryption keys for use with WebLogic 5.1 or 6.1 in a previous PeopleTools release, they're maintained by those earlier versions of WebLogic as external files. You must migrate them to the WebLogic 8.1 keystore so that they work correctly with the current release.

Obtaining Encryption Keys

If you already have a set of existing encryption keys configured as external files, you don't need to obtain new ones. To find the existing keys, refer to the documentation for the PeopleTools and WebLogic releases for which those keys were installed.

The following procedure describes how to obtain new encryption keys, using as an example the 14-day free trial certificate available from Verisign.

To obtain new encryption keys:

1. At a command prompt, change to the following directory:

   PS_HOME\webserv*domain_name*

   Where *domain_name* is the name of the installed PeopleSoft Pure Internet Architecture domain for which you want to obtain encryption keys.

2. Enter the following command:

   ```
   pskeymanager -create
   ```

   Note. Pskeymanager is a script wrapper to Java's keytool, provided by PeopleSoft to manage the WebLogic keystore. For usage information, enter pskeymanager -help.

   ---

3. Follow the prompts and enter the requested information to create a new private key and a CSR for your web server.

   - Pskeymanager uses the keystore in *PS_HOME*\webserv*domain_name*\keystore\pskey, with a default password of *password*.

- Pskeymanager prompts you for an alias for the new keys, for example, *ServerABC*. This is the name you'll use to refer to the keys in the future.
- Pskeymanager prompts you for distinguished name fields. Enter the appropriate values for your organization.
- Pskeymanager prompts you for information about the CSR expiration date, key size, key algorithms, and the private key password. All of these fields have default values.

Pskeymanager creates the private key inside the keystore, and creates the CSR as a file called ServerABC_certreq.txt in the current directory. You use the CSR to obtain your signed public key certificate and a root certificate from a CA.

4. Decide which trusted CA you want to sign your web server's public key.

   You can use any CA that's compatible with Sun's Java 1.4 JKS standard, such as Verisign.

5. Open your CSR file in a text editor and copy its entire contents, including the first and last lines:

```
-----BEGIN NEW CERTIFICATE REQUEST-----
...
...
-----END NEW CERTIFICATE REQUEST-----
```

6. Access Verisign's test certificate enrollment site at https://www.verisign.com/products/srv/trial/intro.html.

   Verisign guides you through the CSR submission process, including:

   - Accepting the Verisign license agreement.
   - Entering your technical contact information, which includes the email address where Verisign can send your signed public key.
   - Pasting your CSR contents in the provided text field.
   - Verifying your CSR.
   - Confirming and submitting your order.

   Verisign also provides its own digitally signed public key in a certificate, which is known as a *trusted CA certificate*, a *root certificate*, or a *chain certificate*.

7. Download the VeriSign test CA root certificate from http://digitalid.verisign.com/cgi-bin/getcacert.

   When prompted, save getcacert.cer to *PS_HOME*\webserv*domain_name*.

Note. If you need to FTP your certificate to UNIX, you must FTP it in ASCII mode to *PS_HOME*/webserv/*domain_name*.

---

8. Check your email.

   Verisign digitally signs your web server's public key, then returns it to you in a certificate, called the *server certificate*. Following is an example of the contents of a server certificate:

```
-----BEGIN CERTIFICATE-----
DMICHDCCAcYCEAHSeRkM2guFL+6OvHr4AS0wDQYJKoZIhvcNAQEEBQAwgakxFjAP
AANVBAoTDVZlcmlTaWduLCBLbAMxRzBFBgNVBAsTPnd3dy52ZXJpc2lnbi5jb20S
VcVwb3NpdG9yeS9UZXN0Q1ETIEluY29ycC4gQnkgUmVmLiBMaWFiLiBMVEQuMUYF
LIGEc3VyYW5jZXMgKEMpVRMxOSDFertdsfh67TIwNDAwMDAwMFoXDTAwMTIxODIA
ONT1LVoweTELMAkGA1UERhMCVVMxEzARBgNVBAgTCkNhbGlmb3JuaWExEzARBgNK
VBAUCOBsZWFzYW50b24BEzARBgNVBAoUClBlb3BsZVNvZnQxFDASBgNVBAsUC1BT
```

```
Eb3sZVVvb2xzMRUwEwADVQQDFAxEQlJPV04xMTE0MDAwXDANBgkqhkiG9w0BAQET
SAALADBEAkEAucfM/GOQhdkk4Q0ZD5i1l4gp6WTYMc4IaReoCYkEAmDKAVcYzY3R
Mdbp4RC8SABd3bjjDOHcoCak9U6oSwL+HQIDAQABMA0GCSqGSIb3DQEBBAUAA0EO
Arm3uf634Md0fqgNxhAL+e9rbY0ia/X48Axloi17+kLtVI1YPOp+Jy6Slp5iNIFC
DhskdDFH45AjSDAFhjruGHJK56SDFGqwq23SFRfgtjkjyu673424yGWE5Gw4576K
DosdDFG256EDHY45yTRH67i345314GQE356mjsdhhjuwbtrh43Gq3QEVe45341tS
YDY6d47lDmQxDs9wGt1bkQ==
-----END CERTIFICATE-----
```

9. Copy the entire certificate contents, and save it as a text file called ServerABC-cert.pem in *PS_HOME*\webserv*domain_name*.

   Be sure to include the first and last lines.

   Note. If you need to FTP your certificate to UNIX, you must FTP it in ASCII mode.

Note. It's a good idea to make backup copies of the server certificate and the root certificate before proceeding.

Preparing Keys and Certificates for the Keystore

Your encryption keys must be in *privacy enhanced mail* (PEM) format, which is Base64-encoded data. Base64 encoding uses only ASCII characters. A PEM-formatted key or certificate file has an extension of either .pem or .cer. If the file is in the binary *distinguished encoding rules* (DER) format, it has a .der extension. Use the *der2pem* Java utility to convert DER-formatted keys and certificates to PEM format.

For SSL to work, your WebLogic 8.1 server must present its own public key to each client browser, along with the self-signed public key of a root CA that's also in the browser's keystore, as well as any keys necessary to establish a *chain of trust* between the two. All of these keys must be part of the same certificate file before you can import them into the WebLogic 8.1 keystore.

If you generated the private key using pskeymanager on a WebLogic 8.1 platform, it's automatically correctly formatted, password protected, and installed in the keystore with no additional steps required. However, if the private key was configured as an external file on an earlier WebLogic platform, you must properly format it and incorporate a password, before importing it into the WebLogic 8.1 keystore along with the public key certificates.

Converting DER Files to PEM Format

It's important to convert all DER-formatted key and certificate files to PEM format before you work with them further.

To convert DER-formatted key and certificate files to PEM format:

1. At a command prompt, change to the following directory:

   PS_HOME\webserv*domain_name*

   Where *domain_name* is the name of an installed PeopleSoft Pure Internet Architecture domain.

2. Enter the following command:

   ```
   setenv.cmd
   ```

   This sets the appropriate environment for java commands.

3. For each DER-formatted key or certificate file, enter the following command:

```
java utils.der2pem filename.der
```

Make sure that you include the DER file's directory path. A new PEM file by the same name is created in the same location.

If you converted a private key file to PEM format, you must modify the header and footer to be compatible with WebLogic 8.1.

To modify the private key file header and footer:

1. Open the PEM-formatted private key file in a text editor.
2. Change the following line:

   ```
   -----BEGIN CERTIFICATE-----
   ```

   To this:

   ```
   -----BEGIN RSA PRIVATE KEY-----
   ```

3. Change the following line:

   ```
   -----END CERTIFICATE-----
   ```

   To this:

   ```
   -----END RSA PRIVATE KEY-----
   ```

4. Save and close the private key file.

Establishing the Server Certificate Chain of Trust

Your server certificate must contain, in addition to the web server's public key, any keys necessary to establish a chain of trust that culminates in the self-signed root certificate of a trusted root CA. That CA's root certificate must be in the keystore of any browser that's used to access your web server. Most browsers have an extensive set of trusted root certificates in their keystores.

First append the root certificate of the CA who issued your server certificate to the server certificate file. If you determine that that root certificate is not likely to be in your users' browsers, you must also append to the certificate file a chain certificate that was issued to your CA by another CA, then a chain certificate issued to that CA, and so on,

For example, if your server certificate file is demo_cert.pem and the CA's root certificate is ca_cert.pem, you can open demo_cert.pem in a text editor, then insert the contents of ca_cert.pem after a newline at the end of the file. Make sure that each certificate follows the previous one on the next line, as follows:

```
...
...
DosdDFG256EDHY45yTRH67i345314GQE356mjsdhhjuwbtrh43Gq3QEVe45341tS
YDY6d47lDmQxDs9wGt1bkQ==
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
DMICHDCCAcYCEAHSeRkM2guFL+6OvHr4AS0wDQYJKoZIhvcNAQEEBQAwgakxFjAP
...
...
```

The result is that demo_cert.pem, for example, now contains the data from both certificates.

If you determine that ca_cert.pem won't be recognized as a trusted root by all of your users' browsers, you must obtain the root certificate of the CA who issued ca_cert.pem and append that to demo_cert.pem as well, and so on, until you append a root certificate that was issued by a trusted CA to itself.

Note. You can also use the *type* command in Windows or the *cat* command in UNIX to combine the certificate files.

Password Protecting the Private Key

Private keys inside the WebLogic 8.1 keystore are password protected. You can't import an external private key file into the keystore without a password. If it isn't currently password protected, use the WebLogic *wlkeytool* utility to incorporate a password into the private key file.

To password-protect an external PEM-formatted private key file:

1. At a command prompt, change to the following directory:

   WL81_HOME\server\bin

   Where *WL81_HOME* is the root directory of your installed WebLogic 8.1 server, for example, C:\bea\weblogic81.

2. Enter the following command:

   ```
   wlkeytool insecure_privatekey.pem secure_privatekey.pem
   ```

   Where *insecure_privatekey*.pem is the name of the original private key file, and *secure_privatekey*.pem is the name of the resulting password-protected private key file.

   Note. Make sure that you include directory paths for the private key files.

   The following message appears:

   ```
   Enter password to unprotect private key:
   ```

3. Press ENTER.

   The following message appears:

   ```
   Private key not PKCS8 encoded, trying RSA key
   Private key file opened successfully
   Enter password to protect private key :
   ```

4. Enter the password that you want to use for this key.

   The following message appears:

   ```
   Verify password to protect private key :
   ```

5. Enter the password again to confirm it.

   The utility creates the password protected private key file that you specified. You can now import the key into the WebLogic 8.1 keystore.

Importing Keys and Certificates Into the Keystore

Each WebLogic domain maintains its own keystore in *PS_HOME*\webserv*domain_name*\keystore\pskey, and all servers within a domain can share the same keystore.

Two tools are available for importing keys and certificates into the keystore:

- If you created the private key using the pskeymanager utility on a WebLogic 8.1 platform, it's already installed in the keystore. You need only use pskeymanager to import your server certificate, which should

contain your web server's signed public key, your trusted CA's root certificate, and any public keys necessary to establish a chain of trust between them.

- If the private key was previously configured as an external file on an earlier WebLogic platform, you must import it into the WebLogic 8.1 keystore along with the server certificate, using the *ImportPrivateKey* utility. The private key should be password-protected.

Using Pskeymanager to Import the Server Certificate

To import the server certificate into the WebLogic 8.1 keystore:

1. At a command prompt, change to the following directory:

   PS_HOME\webserv*domain_name*

   Where *domain_name* is the name of the installed PeopleSoft Pure Internet Architecture domain.

2. Enter the following command:

   ```
   pskeymanager -import
   ```

   Note. Pskeymanager is a script wrapper to Java's keytool, provided by PeopleSoft to manage the WebLogic keystore. For usage information, enter `pskeymanager -help`.

3. Follow the prompts and enter the requested information to create a new private key and a CSR for your web server. Keep the following in mind:

   - Pskeymanager uses the keystore in *PS_HOME*\webserv*domain_name*\keystore\pskey, with a default password of *password*.
   - Pskeymanager prompts you for an alias for the server certificate, for example, *ServerABC*. This should be the same alias that you specified for the corresponding private key when you created it.
   - Pskeymanager prompts you for the name of the server certificate file, for example, *ServerABC-cert.pem*. Include the file path if necessary.

   Pskeymanager imports the server certificate into the keystore.

Using ImportPrivateKey to Import an External Private Key File with the Server Certificate

To import a password-protected private key and the server certificate into the WebLogic 8.1 keystore:

1. At a command prompt, change to the following directory:

   PS_HOME\webserv*domain_name*

   Where *domain_name* is the name of an installed PeopleSoft Pure Internet Architecture domain.

2. Enter the following command:

   ```
   setenv.cmd
   ```

   This sets the appropriate environment for java commands.

3. Enter the following command:

   ```
   java utils.ImportPrivateKey keystore\pskey store_pass privatekey_alias
   privatekey_pass servercert_file privatekey_file
   ```

   The parameters for this command are as follows:

store_pass Specify the password for the WebLogic pskey keystore. The default password is *password*.

privatekey_alias Specify an alias for the private key. This is the name by which the key will be accessible inside the keystore.

privatekey_pass Specify the password for the private key.

servercert_file Specify the path and name of the server certificate file that includes the issuing CA's root certificate.

privatekey_file Specify the path and name of the private key file.

The encryption keys are installed in the WebLogic 8.1 keystore, and you can now configure them using the WebLogic server administration console.

Configuring WebLogic SSL Encryption Keys

This section describes how to configure the SSL encryption keys that you previously imported into the WebLogic 8.1 keystore in *PS_HOME*\webserv*domain_name*\keystore\pskey, where *domain_name* is the name of an installed PeopleSoft Pure Internet Architecture domain.

The following procedure applies to a single server configuration of PIA. In a production environment, you would perform these steps for managed server instances of PIA, PIA1, PSOL, RPS, and so on, in a multi-server domain configuration.

To configure WebLogic SSL encryption keys for the PIA server:

1. With the PIA server running, sign in to the WebLogic Server Administration Console.

   Access the WebLogic Server console at http://*webservername*/console (for example, http://localhost/console). When prompted for a user name and password, enter the WebLogic system ID and password, which you defined during the PIA install. The default user name and password are *system* and *password*, respectively.

2. Access the keystore configuration pages.
   a. In the lefthand navigation tree, navigate to peoplesoft, Servers, PIA.
   b. Select Keystores & SSL.
   c. In the Keystore Configuration section, click Change.
   d. Select *Custom Identity and Custom Trust*, and click Continue.

   The Configure Keystore Properties page appears, with the configuration fields for the pskey keystore.

3. Update the fields on the Configure Keystore Properties page as follows:

Field	Value	Comment
Custom Identity Key Store File Name	*keystore/pskey*	This should be the relative path and name of the keystore into which you imported your SSL keys.
Custom Identity Key Store Type	*JKS*	Don't change this value.
Custom Identity Key Store Pass Phrase	*password*	See the following note regarding passwords.

Field	Value	Comment
Confirm Custom Identity Key Store Pass Phrase	Same as the value of Custom Identity Key Store Pass Phrase.	
Custom Trust Key Store File Name	*keystore/pskey*	This should be the relative path and name of the keystore into which you imported your SSL keys.
Custom Trust Key Store Type	*JKS*	Don't change this value.
Custom Trust Key Store Pass Phrase	*password*	See the following note regarding passwords.
Confirm Custom Trust Key Store Pass Phrase	Same as the value of Custom Trust Key Store Pass Phrase.	

Note. The default keystore and private key password is *password*. This should never be used in a production environment. You can change a private key's password and a keystore's password using pskeymanager's change password options: -changeprivatekeypassword and -changekeystorepassword, respectively.

4. Click Continue to access the Review SSL Private Key Settings page.
5. Update the fields on the Review SSL Private Key Settings page as follows:

Field	Value	Comment
Private Key Alias	Specify a unique identifier, such as the webserver's machine name.	This is the alias that you specified for this server's private key.
Passphrase	*password*	See the following note regarding passwords.
Confirm Passphrase	Same as the value of Passphrase.	

Note. The default keystore and private key password is 'password'. This should never be used in a production environment. A private key's password and a keystore's password can be changed via pskeymanager's change password options of –changekeystoreword and –changeprivatekeypassword.

6. Save your keystore changes.

   You *must* click the 'Finish' button to save your changes. If you close your browser without clicking 'Finish', your changes will be lost.
7. Restart the WebLogic PIA server.

Restricting Access to a Servlet

This section describes how to restrict access to a web resource for a single server configuration of PIA. When in production, a multi server configuration would be used to perform these steps to your managed server instance of PIA or PIA1, etc.

BEA WebLogic Server provides an optional level of security to restrict access to resources on the web server. The steps below describe how to restrict access via a WebLogic ID and password to the PeopleSoft Portal servlet:

To restrict access to a servlet:

1. Start the PIA server either through `startPIA.cmd(.sh)` or, if installed, as a Microsoft Windows service, `NET START peoplesoft-PIA`.
2. Log in to the BEA WebLogic Server console (http://localhost/console).
3. (Optional) Define the BEA WebLogic users that you want to use.

   If you want to use one of the WebLogic provided user accounts, system, operator and monitor, you can skip this step. Otherwise create a new BEA WebLogic user account by performing the following from the navigation window on the left of the WebLogic console:

   a. Expand peoplesoft, Security, Realms, myrealm.
   b. Select Users.
   c. Select Configure a Nrew User.
   d. Enter the new user information and click Apply.
4. (Optional) Create a user group.

   If you want to create a user group, add your users to that group and in the following steps select Caller is Member of group instead of User name of caller. To create a group, perform the following from the navigation window on the left of the WebLogic console.

   a. Expand peoplesoft, Security, Realms, myrealm.
   b. Select Groups.
   c. Select Configure a new Group.
   d. On the Create Group - General tab, enter a new group name.

      To add a new user to this group, select the Users folder from the navigation on the left, select your new user, then click the Groups tab and add the user to your newly created group.
   e. Click Apply.
5. Link security roles to security policies.

   In the navigation window on the left, perform the following to enable security policy checks for web applications:

   a. Expand peoplesoft, Security, Realms.
   b. Select myrealm.
   c. On the myrealm - General tab, select *All Web applications and EJBs* from the Check Roles and Policies fordrop down list.
   d. Click Apply.

   This doesn't mean that all web applications require authorization. That's controlled with the following steps.
6. Define a security policy for the PeopleSoft Portal web application.

   To restrict access to the Portal web application, perform the following in the navigation window on the left:

   a. Expand peoplesoft, Deployments, applications, peoplesoft.
   b. Right click the PORTAL web application and select Define Security Policy.

Note. To apply a security policy to a different web application, substitute that web application name.

c. Specify the URL which requires authentication.

In this case simply specify /* and click Define Security Policy.

d. To restrict access to a specific user, select the Policy condition of *User name of caller*, click Add, and when prompted specify the user name.

Repeat this step for additional users, groups, or access times. For access times, the server's local time is used.

7. Exit the console, shut down WebLogic to apply your changes, then restart WebLogic.
8. Test the configuration.

Test your new security policy by accessing the URL you defined in step 6. If the security policy is active, you'll be prompted to sign in using a user ID that you added in step 3.

See Also

http://edocs.bea.com/wls/docs81/secwlres/index.html

Adjusting the JVM Heap Size

The Java options including the JVM heap size, VM mode, such as HotSpot Server, used by the WebLogic server are stored in your WebLogic domain's setEnv script (for example, *PS_HOME*\webserv\peoplesoft\setEnv.cmd). These options are specified in the script using the JAVA_OPTIONS_*OSplatform* environment variable. If you need to adjust any of the java options, including changing the JVM heap size, you must manually edit the script.

The Microsoft Windows setEnv.cmd script contains the following default setting:

```
JAVA_OPTIONS_WIN32="-server -Xms32m -Xmx200m -XX:MaxPermSize=128m"
```

The UNIX standard setEnv.sh script contains the following default settings for supported Linux and UNIX platforms:

```
JAVA_OPTIONS_AIX="-Xms32m -Xmx200m"
JAVA_OPTIONS_HPUX="-server -Xms32m -Xmx200m -XX:MaxPermSize=128m"
JAVA_OPTIONS_LINUX="-Xms128m -Xmx256m"
JAVA_OPTIONS_TRU64="-Xms32m -Xmx200m"
JAVA_OPTIONS_SOLARIS="-server -Xms32m -Xmx200m -XX:MaxPermSize=128m"
```

You modify the –Xms parameter to adjust minimum heap size, and modify the –Xmx parameter to adjust maximum heap size.

In a multi-server domain, the platform-specific versions of the JAVA_OPTIONS environment variable that are shown in the setEnv script apply only to managed servers. The administration server doesn't use any of these variables, but it assumes default JVM heap size values of `"-Xms32m -Xmx64m"`.

To adjust the JVM heap size for the administration server, add the environment variable JAVA_OPTIONS_ADMINSERVER following the last entry for JAVA_OPTIONS_*OSplatform*, and set it to your own minimum and maximum values, for example:

```
JAVA_OPTIONS_ADMINSERVER="-Xms64m -Xmx128m"
```

Note. If you're running BEA WebLogic as a Microsoft Windows service and you modify setEnv.cmd, you must reinstall the service by running installNTservicePIA.cmd or InstallNTservice.cmd from the WebLogic domain directory again.

See Also

Appendix A, "BEA WebLogic 8.1 Managed Server Architecture," Managing JVM Heap Size and Execute Thread Usage, page 303

Determining the Service Pack Level

A summary of installed products, their versions and service pack levels, is maintained in the *BEA_HOME*\registry.xml file. However, to confirm version information, it's more accurate to check the BEA WebLogic log. A failed service pack install may be indicated in the log, but not found at runtime.

This section discusses how to:

- Check the log
- Query BEA WebLogic

Checking the Log

In the BEA WebLogic log (*PS_HOME*\webserv\peoplesoft\logs*weblogic_server*_weblogic.log), look for an entry similar to this:

```
WebLogic Temporary Patch 1 CR110692 for PeopleSoft 10/17/2003 18:11:35
WebLogic Server 8.1 SP1  Fri Jun 20 23:06:40 PDT 2003 271009
```

Querying BEA WebLogic

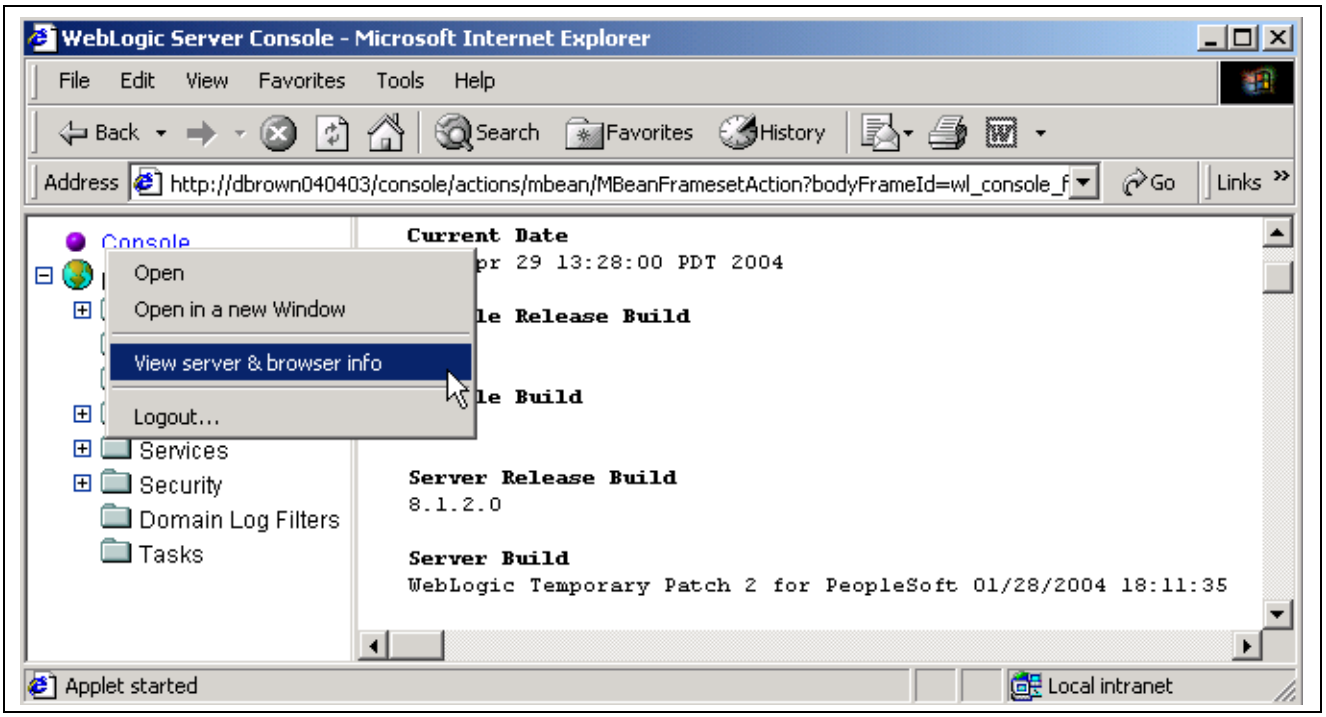
You can query BEA WebLogic at the command line or by using the BEA WebLogic Server console.

Perform a query at the command line as shown in this example (for UNIX, use setEnv.sh):

```
PS_HOME\webserv\peoplesoft\setenv.cmd⇒
 java weblogic.Admin VERSION -url t3://localhost:80

WebLogic Temporary Patch 1 CR110692 for PeopleSoft 10/17/2003 18:11:35
WebLogic Server 8.1 SP1  Fri Jun 20 23:06:40 PDT 2003 271009
```

Perform a query by using the BEA WebLogic Server console (http://localhost/console) by right-clicking Console. Select View Server & Browser info as shown in this example:



Querying the WebLogic console

Enabling or Disabling HTTP Access Log

This section describes how to change HTTP logging for a single server configuration of PIA. When in production, a multi server configuration would be used to perform these steps to your managed server instance of PIA or PIA1, etc.

To enable or disable HTTP access log:

1. Start the PIA server.

   Start the PIA server either via startPIA.cmd(.sh) or if installed as a Windows service, " NET START peoplesoft-PIA".

   See Chapter 6, “Working with BEA WebLogic,” Starting BEA WebLogic, page 99.

   See Chapter 6, “Working with BEA WebLogic,” Stopping BEA WebLogic, page 101.

2. Log on to the WebLogic Server Administrative Console.

   In a new browser access the WebLogic Server console at http://localhost/console and specify the WebLogic administrative ID that you specified during the PIA installation. The default ID and password are ’system’ and ’password’, respectively.

3. Open Server’s Logging configuration page.

   In the navigation window on the left, navigate to the following to open the PIA server’s HTTP configuration settings. (If you are using a custom server name, substitute that name where appropriate.)

   a. Expand ’peoplesoft’.

   b. Expand ’Servers’.

   c. Double click ’PIA’ .

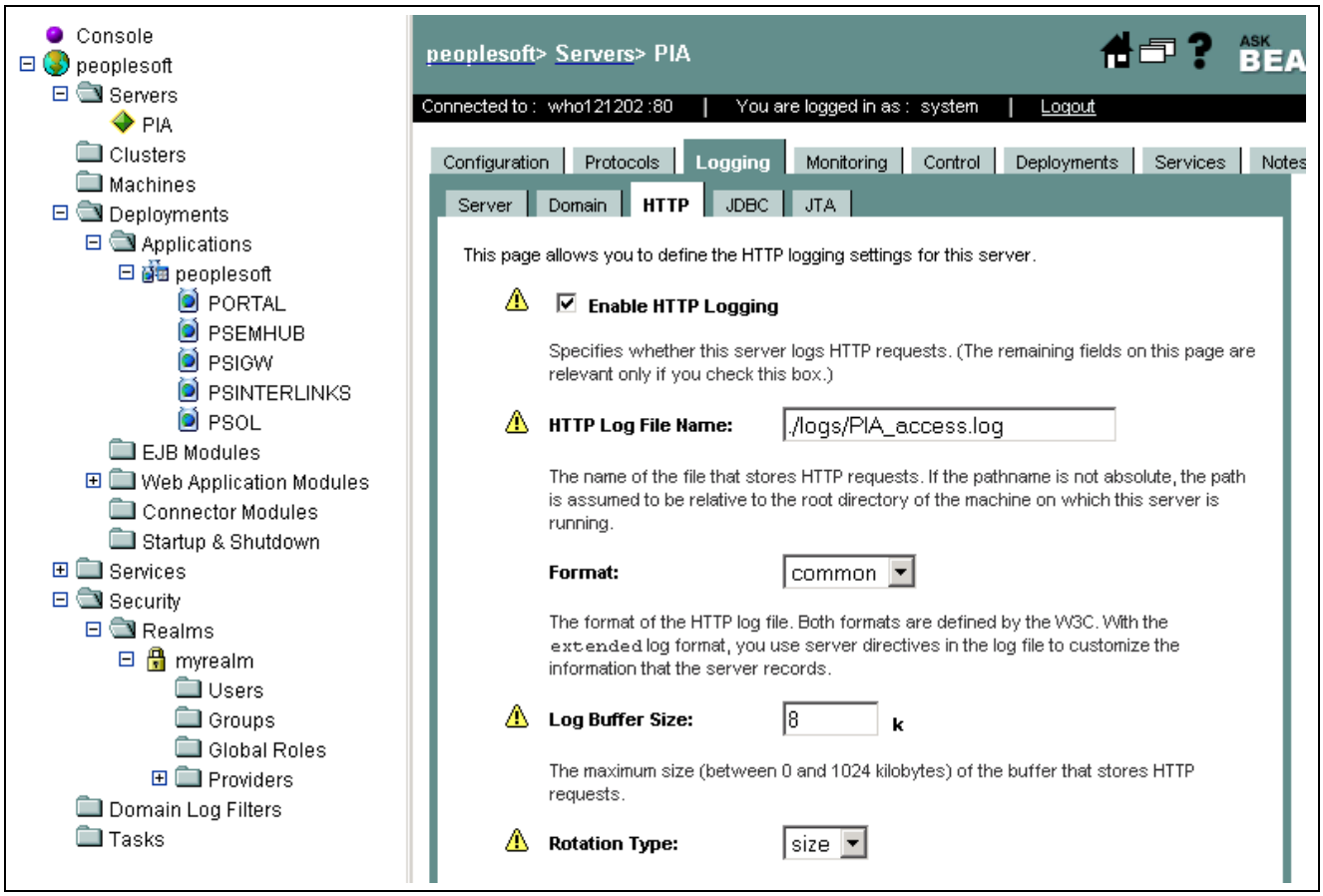
d. Select 'Logging'.

e. Select 'HTTP'.

f. Select the 'Enable HTTP Logging' check box to turn on the access.log. Change the Logfile name if desired.

g. Click the 'Apply' button on the bottom of the page.

4. Restart the WebLogic Server.



WebLogic Console page

CHAPTER 7

Working with IBM WebSphere

This chapter discusses how to:

- Install and uninstall WebSphere features.
- Set up HTTP Server (RPS) with WebSphere 5.1.1.
- Set up SSL on WebSphere 5.1.1.
- Set up SSL on IBM HTTP Server.
- Use the WebSphere 5.1.1 administration console.
- Use the log analyzer and resource analyzer.
- Uninstall PIA on WebSphere.
- Install additional PIA to existing WebSphere 5.1.1 instance.
- Deploy multiple PIA instances on WebSphere 5.1.1.
- Reserve WebSphere 5.1.1 PIA Ports.
- Access the WebSphere InfoCenter.

Note. This chapter discusses two editions of IBM's web server software: *WebSphere Application Server* (base) and *WebSphere Application Server Network Deployment*. For clarity and brevity, these products are referred to respectively as WebSphere Base and WebSphere ND, or collectively as just WebSphere. The home directory where you install WebSphere is referred to as *WAS_HOME*.

This information is included in the PeopleSoft documentation because it applies directly to the PeopleSoft system. It's not intended to replace any IBM WebSphere documentation. Always refer to the IBM documentation for detailed information about IBM WebSphere.

Installing and Uninstalling WebSphere Features

This section discusses how to:

- Install WebSphere 5.1.1 using silent install.
- Uninstall WebSphere 5.1.1.
- Uninstall WebSphere embedded messaging.

Installing WebSphere 5.1.1 Using Silent Install

This section discusses how to use the *silent install* feature of WebSphere, which completes the installation without requiring additional input from a user. Silent install relies on the provided response file (wasBase.resp for WebSphere Base, and DeployMgr.resp for WebSphere ND). The response file parameters are predefined for PeopleSoft application optimization.

Note. It's recommended that you be signed in as a member of the Administrator group to install WebSphere. On UNIX systems, you must be signed in as root to install WebSphere. After installation, WebSphere can be changed to run as a non-root user.

To install WebSphere using silent install:

1. Stop any HTTP Server (IBM Http Server, Microsoft IIS, or Sun ONE Web Server) running on the system.
2. At a DOS command prompt in Windows or an xterm command prompt in UNIX, navigate to the root of the WebSphere installation CD.
3. Launch the appropriate install command for your platform.
   - To install WebSphere Base in Windows:

     ```
     installBase.bat -silent
     ```

   - To install WebSphere ND in Windows:

     ```
     installDeployMgr.bat -silent
     ```

   - To install WebSphere Base in UNIX:

     ```
     installBase.sh -silent
     ```

   - To install WebSphere ND in UNIX:

     ```
     installDeployMgr.sh -silent
     ```

   WebSphere is silently installed without interaction.

   Note. The silent install does not install any HTTP proxy server plug-ins or the IBM HTTP Server.

   You can monitor the installation process by viewing the log.txt file in the system Temp directory. After the installation completes, this file is moved to the WebSphere logs directory.
4. Check the logs for any errors encountered during the installation.

   The WebSphere install log is *WAS_HOME*\logs\log.txt. The IBM HTTP Server log is *WAS_HOME*\logs\ihs_log.txt.
5. Restart the HTTP servers.

Uninstalling WebSphere 5.1.1

To uninstall WebSphere 5.1.1:

1. Stop any HTTP Server (IBM Http Server, Microsoft IIS, or Sun ONE Web Server) running on the system.
2. At a DOS command prompt (Windows) or an xterm command prompt (UNIX), navigate to *WAS_HOME*_uninst .
3. Launch the `uninstall` command.

   Note. On UNIX systems, if the DISPLAY variable is set, the uninstall program will work in GUI mode. You can also launch the command `uninstall -silent` to run the uninstall process silently.
4. You must manually delete the WebSphere directory structure.

5. Restart the HTTP servers.

Uninstalling WebSphere Embedded Messaging

If you don't have IBM WebSphere MQ installed as a separate product on the same machine, you can use the procedure for your platform, as follows, to manually uninstall WebSphere embedded messaging.

In Linux

If you're certain that there's no embedded messaging data to preserve, use the following procedure:

1. At a command prompt, enter :

```
rm -fr /var/wemps /opt/wemps
rm -fr /var/mqm /opt/mqm
```

2. Enter the following commands to search for WebSphere-related packages:

```
rpm -qa | grep WS
rpm -qa | grep MQ
rpm -qa | grep wemps
```

3. Enter the following command to remove each package found in the previous step:

```
rpm -e packagename
```

In AIX

At a command prompt, launch the following command as root:

```
Installp -u wemps mqjava mqm
```

Enter *y* (yes) in response to all prompts.

In Solaris

At a command prompt, launch the following command as root:

```
pkgrm wemps mqjava mqm-upd04 mqm
```

Enter *y* (yes) in response to all prompts.

In HP-UX

To remove embedded messaging:

1. Start the System Administration Manager (SAM) utility.
2. Verify that your DISPLAY and TERM environment variables are set properly.
3. Click Software management.
4. Click View installed software.
5. Note whether either of the following MQSeries entries are on the SD list:

   MQSERIES -> B.11.530.01 WebSphere MQ for HP-UX

   MQSERIES -> B.11.530.03 WebSphere MQ Update (U485562)
6. Close the SD list.

7. Click Remove local host software.
8. On the SD Remove list, remove these MQSeries instances, if they exist:

   MQSERIES -> B.11.530.01 WebSphere MQ for HP-UX

   MQSERIES -> B.11.530.03 WebSphere MQ Update (U485562) for HP-UX

   a. Highlight the MQSeries instances on the list.
   b. Select Actions, Remove.
   c. In the Remove Analysis dialog, verify that Status is *Ready*.
   d. Click OK.
   e. Check the result message displayed in the dialog.
   f. If the the message indicates that the removal failed, follow the instructions that appear.
9. Return to the SD Remove list.
10. Click to hightlight *IBM HTTP Server*, *MQSERIES*, *WEMPS*, *WSBAA*, *WSNAA*, and *gsk5bas*.
11. Select Actions, Mark for remove.
12. Select Actions, Remove.
13. In the Remove analysis dialog box, click OK.
14. To display real-time removal of selected packages, click Logs.
15. When the removal process finishes, exit SAM.

In Windows

If you have other instances of WebSphere Application Server products or WebSphere MQ that are installed on the same machine and use the embedded messaging feature, do not remove embedded messaging.

However, if the copy of WebSphere that you installed for use with PeopleSoft software is the only one that uses embedded messaging, you can remove the feature.

Use the Windows Add/Remove Programs utility to remove the following programs, if they're listed:

- IBM WebSphere EMPS
- IBM WebSphere MQ

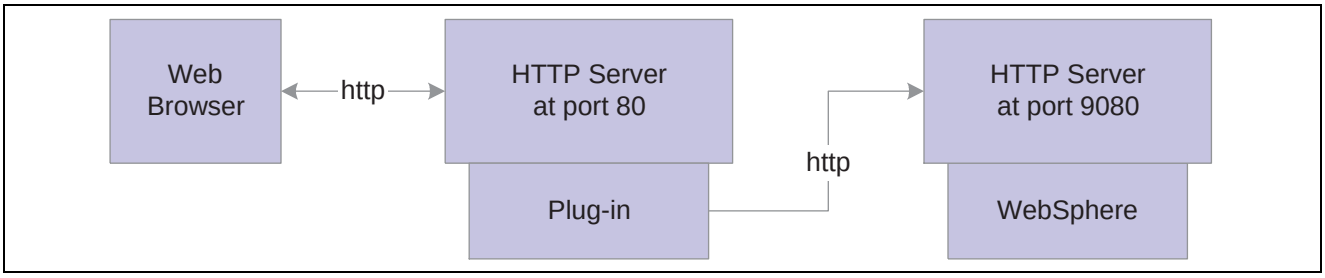
Setting Up HTTP Server (RPS) with WebSphere 5.1.1

This section provides an overview of WebSphere reverse proxy servers (RPS) and discusses how to:

- Set up a WebSphere machine that requires RPS support.
- Install IBM HTTP Server and its RPS plug-in together.
- Install only the RPS plug-in for IBM HTTP Server.
- Install only the RPS plug-in for Sun ONE Web Server.
- Install only the RPS plug-in for Microsoft Internet Information Server (IIS).
- Configure the HTTP server.

Understanding WebSphere Reverse Proxy Servers

WebSphere 5.1.1 supports several HTTP servers: IBM HTTP Server, Microsoft IIS, and Sun ONE Web Server. A client makes a request to the HTTP Server, which delegates the request to an RPS plug-in, which forwards the request to WebSphere.



WebSphere RPS Plug-in

You can install the RPS plug-in by itself on a machine where WebSphere 5.1.1 Base has been installed but the plug-in has not. You can also install a plug-in on a remote machine where the HTTP proxy server is already installed (IBM HTTP Server, Microsoft IIS, or Sun ONE Web Server).

To install the WebSphere plug-in for a given HTTP proxy server, you must first manually update a response file with the correct options for that HTTP proxy server. Template response files are provided with options optimized for PeopleSoft applications. The response files are located on the installation CD under the \base directory. After updating the appropriate response file, you can run the installer using this file as input, as described in the following sections.

Note. Before installing an RPS plug-in, shut down all HTTP proxy servers that you plan to run with WebSphere.

Setting Up a WebSphere Machine that Requires RPS Support

Use the following steps to set up the WebSphere machine that requires RPS support:

1. Start local or remote WebSphere Base.
2. Start respective WebSphere server1.
3. Open Administration Console from browser using the following URL:

   http://*WAS_machine_name*:9090/admin
4. Expand Environment, Virtual Host, default_host, Host Aliases: Add a new port with the hostname “*”. Click OK and Save.
5. Expand Environment, Update Web Server Plugin.
6. Click OK to update the web server plug-in.
7. Click OK to save your setting.
8. Log out and close the browser.

   You will see the plugin-cfg.xml file at *WAS_HOME*\config\cells\plugin-cfg.xml.

Installing IBM HTTP Server and its RPS Plug-in Together

This section assumes that you haven’t yet installed IBM HTTP Server. Launching the install for the appropriate RPS plug-in automatically installs IBM HTTP Server as well.

To install the IBM HTTP Server and its plug-in:

1. Copy the file IHS_N_Plugin.resp from the installation CD \base directory to the the system Temp directory.
2. Edit IHS_N_Plugin.resp, and update the following lines:

   ```
   -P wasBean.installLocation="C:\WebSphere51\plugin"
   -P ihsFeatureBean.installLocation="C:\IBMHttpServer"
   ```

   Change the value of wasBean.installLocation to the directory location where you want the plug-in modules to be installed. Set ihsFeatureBean.installLocation to the directory location where you want to install IBM HTTP server.
3. At a command prompt, change to the installation CD directory \base*operating_system*.

   Operating_system can be *sun*, *win*, *aix*, *hp*, or *linux*.
4. Enter the following command:

   ```
   install -options tempdir\IHS_N_Plugin.resp
   ```

   This starts the install with just the selected options, and begins the installation of the plug-in silently. You can monitor the installation by viewing the log.txt file in the system Temp directory.

Note. Check the logs for any errors encountered during the installation. WebSphere's install log is located in *WAS_HOME*\logs\log.txt. The IBM HTTP Server log is in *WAS_HOME*\logs\ihs_log.txt.

Installing Only the RPS Plug-in for IBM HTTP Server

This section assumes that IBM HTTP Server is already installed.

The following procedure describes how to set up the response file if you're installing only the IBM HTTP Server RPS plug-in. You do this if you already have IBM HTTP Server installed.

To install the IBM HTTP Server plug-in:

1. Copy the file IHSproxyPlugin.resp from the installation CD \base directory to the the system Temp directory.
2. Edit IHSproxyPlugin.resp, and update the following lines:

   ```
   -P wasBean.installLocation="C:\WebSphere51\plugin"
   -W defaultIHSConfigFileLocationBean.value="C:\IBMHttpServer\conf\httpd.conf"
   ```

   Change the value of wasBean.installLocation to the directory location where you want the plug-in modules to be installed. Set defaultIHSConfigFileLocationBean to the full path of the IBM HTTP Server configuration file (httpd.conf).
3. At a command prompt, change to the installation CD directory \base*operating_system*.

   Operating_system can be *sun*, *win*, *aix*, *hp*, or *linux*.
4. Enter the following command:

   ```
   install  options tempdir\IHSproxyPlugin.resp
   ```

   This starts the install with just the selected options, and begins the installation of the plug-in silently. You can monitor the installation by viewing the log.txt file in the system Temp directory.
5. (Windows server only) Reboot the server machine.

   A reboot is not necessary on UNIX platforms.

6. Check whether any iFixes are needed for WebSphere.

   Refer to the Customer Connection link ftp://ftp.peoplesoft.com/outgoing/PTools/websphere/511PT846. If any iFixes are present, you must install them.

Note. Check the logs for any errors encountered during the installation. WebSphere's install log is located in *WAS_HOME*\logs\log.txt. The IBM HTTP Server log is in *WAS_HOME*\logs\ihs_log.txt.

Installing Only the RPS Plug-in for Sun ONE Web Server

This section assumes that Sun ONE Web Server is already installed.

The following procedure describes how to set up the response file if you're installing only the Sun ONE Web Server RPS plug-in. You do this if you already have Sun ONE Web Server installed.

To install the Sun ONE Web Server plug-in:

1. Copy the file SunOneProxyPlugin.resp from the installation CD \base directory to the the system Temp directory.
2. Edit SunOneProxyPlugin.resp, and update the following lines:

   ```
   -P wasBean.installLocation="C:\WebSphere51\plugin"
   -W defaultIPlanetConfigFileLocationBean.value="C:\iPlanet60\conf\magnus.conf"
   ```

   Change the value of wasBean.installLocation to the directory location where you want the plug-in modules to be installed. Set defaultIPlanetConfigFileLocationBean to the full path of the Sun ONE Web Server configuration file (for example, C:\iPlanet60\conf\obj.conf).
3. At a command prompt, change to the installation CD directory \base*operating_system*.

   Operating_system can be *sun*, *win*, *aix*, *hp*, or *linux*.
4. Enter the following command:

   ```
   install  options tempdir\SunOneProxyPlugin.resp
   ```

   This starts the install with just the selected options, and begins the installation of the plug-in silently. You can monitor the installation by viewing the log.txt file in the system Temp directory.
5. (Windows server only) Reboot the server machine.

   A reboot is not necessary on UNIX platforms.
6. Check whether any iFixes are needed for WebSphere.

   Refer to the Customer Connection link ftp://ftp.peoplesoft.com/outgoing/PTools/websphere/511PT846. If any iFixes are present, you must install them.

Note. Check the logs for any errors encountered during the installation. WebSphere's install log is located in *WAS_HOME*\logs\log.txt.

Installing Only the RPS Plug-in for Microsoft IIS

This section assumes that Microsoft IIS is already installed.

The following procedure describes how to set up the response file if you're installing only the Microsoft IIS RPS plug-in. You do this if you already have Microsoft IIS installed.

To install the MicroSoft IIS plug-in:

1. Copy the file IISproxyPlugin.resp from the installation CD \base directory to the the system Temp directory.
2. Edit IISproxyPlugin.resp, and update the following lines:

   ```
   -P wasBean.installLocation="C:\WebSphere51\plugin"
   ```

   Change the value of wasBean.installLocation to the directory location where you want the plug-in modules to be installed.
3. At a command prompt, change to the installation CD directory \base\win.
4. Enter the following command:

   ```
   install  options tempdir\IISproxyPlugin.resp
   ```

   This starts the install with just the selected options, and begins the installation of the plug-in silently. You can monitor the installation by viewing the log.txt file in the system Temp directory.
5. (Windows server only) Reboot the server machine.

   A reboot is not necessary on UNIX platforms.
6. Check whether any iFixes are needed for WebSphere.

   Refer to the Customer Connection link ftp://ftp.peoplesoft.com/outgoing/PTools/websphere/511PT846. If any iFixes are present, you must install them.

Note. Check the logs for any errors encountered during the installation. WebSphere's install log is located in *WAS_HOME*\logs\log.txt.

Configuring the HTTP Server

The plugin-cfg.xml file on the HTTP Server machine (RPS) reads WebSphere's plugin-cfg.xml file to forward requests to WebSphere. Copy the existing plugin-cfg.xml file on the RPS machine to a safe backup location and replace it with the plugin-cfg.xml file from *WAS_HOME*\config\cells\plugin-cfg.xml.

You can find the location of the existing plugin-cfg.xml on the RPS machine as follows:

- On an IBM Http Server machine —

  Open *IBM_HTTP_Server_HOME*\conf\httpd.conf, and search for "plugin-cfg.xml" in it to discover the location of the plugin-cfg.xml file on the RPS machine.
- On a Sun ONE (iPlanet) Web Server machine —

  Open *Sun_ONE_HOME*\servers\https-*machine.domain*\config\magnus.conf, and search for "plugin-cfg.xml" in it to discover the location of the plugin-cfg.xml file on the RPS machine.
- On a Microsoft IIS machine —

  Open the Windows registry, expand HKEY_LOCAL_MACHINE > SOFTWARE > IBM > WebSphere Application Server > 5.0.0.0.Plugin Config. The key points to the plugin-cfg.xml file on the RPS machine.

If you experience any problem with the HTTP plug-in component—the component that sends requests from your HTTP server (for example, IBM HTTP Server, Apache, iPlanet, or IIS) to WebSphere—try doing the following to resolve the problem:

- Review the file *install_dir*\logs\http_plugin.log for clues. Look up any error or warning messages in the message table.
- Review the HTTP server's error and access logs to see if the HTTP server is experiencing a problem. Following are the log files for each server:

- IBM HTTP Server and Apache — access.log and error.log
- iPlanet — access and error
- Microsoft IIS — timedatestamp.log

Setting Up SSL on WebSphere 5.1.1

This section discusses how to:

- Generate a certificate for the web container.
- Modify the web container to support SSL
- Test your setup.

Note. This section assumes that you've set up PeopleSoft Internet Architecture (PIA) on your web server machine. The PIA setup creates a WebSphere domain directory called *PS_HOME*\webserv*cellname_nodename_servername*\peoplesoft.ear.

Generating a Certificate for the Web Container

Use the following steps to generate a self-signed certificate for the web container.

1. At a command prompt, change to the WebSphere domain directory, for example:

   PS_HOME\webserv\mymachine_mymachine_server1\peoplesoft.ear

2. Create a new private key and certificate request for your server.

   a. To create a new private key and certificate signing request, run pskeymanager.cmd -create.

   b. Follow the prompts and specify the information that you normally would when creating a certificate. Pskeymanager script is a wrapper to Java's keytool, provided by PeopleSoft to manage the predefined WebSphere keystore of *PS_HOME*\webserv*cellname_nodename_servername*\peoplesoft.ear\keystore\pskey.

3. Decide which Certificate Authority you wish to use.

   At the completion of step 2 a Certificate Signing Request (CSR) file named %ALIAS%_certreq.txt was created in *PS_HOME*\webserv*cellname_nodename_servername*\peoplesoft.ear, and its contents displayed. If you submit this data to a Certificate Authority for processing , you obtain a public key that you can load into your keystore.

   At this point, you may use any Certificate Authority that is compatible with Sun's Java 1.4 JKS standard.

   As an example, the following steps indicate how to provide the CSR that you generated in step 4 to Verisign to obtain a 14-day free trial certificate.

4. Submit your CSR to Verisign.

   Access Verisign's test cart enrollment site at https://www.verisign.com/products/srv/trial/intro.html. Agree to the license and continue to "Step 2 of 5: Submit CSR". In the large edit box provided, copy and paste the contents of your CSR generated in step 2.

5. Supply Verisign with contact information.

Fill out the table titled "Enter Technical Contact Information" with your information and verify that the radio button for the "Free 14-day Trial Server ID" is selected. Once this is done, agree to the license information and click 'Accept'. Your certificate will be emailed to the email address you specified. By selecting the free trial ID, you do not need to fill out the "Cardholder Information" table.

6. Check your email.

   Once you've received your certificate email from VeriSign, you can see your actual certificate in the following format:

   This is a sample certificate file:

```
-----BEGIN CERTIFICATE-----
DMICHDCCAcYCEAHSeRkM2guFL+6OvHr4AS0wDQYJKoZIhvcNAQEEBQAwgakxFjAP
AANVBAoTDVZlcmlTaWduLCBLbAMxRzBFBgNVBAsTPnd3dy52ZXJpc2lnbi5jb20S
VcVwb3NpdG9yeS9UZXN0Q1ETIEluY29ycC4gQnkgUmVmLiBMaWFiLiBMVEQuMUYF
LIGEc3VyYW5jZXMgKEMpVRMxOSDFertdsfh67TIwNDAwMDAwMFoXDTAwMTIxODIA
ONT1LVoweTELMAkGA1UERhMCVVMxEzARBgNVBAgTCkNhbGlmb3JuaWExEzARBgNK
VBAUCOBsZWFzYW50b24BEzARBgNVBAoUClBlb3BsZVNvZnQxFDASBgNVBAsUC1BT
Eb3sZVVvb2xzMRUwEwADVQQDFAxEQlJPV04xMTE0MDAwXDANBgkqhkiG9w0BAQET
SAALADBEAkEAucfM/GOQhdkk4Q0ZD5i1l4gp6WTYMc4IaReoCYkEAmDKAVcYzY3R
Mdbp4RC8SABd3bjjDOHcoCak9U6oSwL+HQIDAQABMA0GCSqGSIb3DQEBBAUAA0EO
Arm3uf634Md0fqgNxhAL+e9rbY0ia/X48Axloi17+kLtVI1YPOp+Jy6Slp5iNIFC
DhskdDFH45AjSDAFhjruGHJK56SDFGqwq23SFRfgtjkjyu673424yGWE5Gw4576K
DosdDFG256EDHY45yTRH67i345314GQE356mjsdhhjuwbtrh43Gq3QEVe45341tS
YDY6d47lDmQxDs9wGt1bkQ==
-----END CERTIFICATE-----
```

   Copy the certificate information, including --BEGIN CERTIFICATE-- and --END CERTIFICATE-- and save it as a file called *webservername*-cert.pem. (Don't use a word processor such as Microsoft Word that inserts formatting or control characters.) If you need to FTP your certificate to UNIX, you must FTP it in ASCII mode.

7. Download the VeriSign TestCA certificate:

   Download the VeriSign test CA certificate from http://digitalid.verisign.com/cgi-bin/getcacert. When prompted, save getcacert.cer to your WebSphere domain directory. If you need to FTP your certificate to UNIX, you must FTP it in ASCII mode to your WebSphere domain directory.

8. Import the Verisign test Certificate Authority's certificate into your keystore.

   To import the Certificate Authority's public certificate (which you received from Verisign) into your keystore, run pskeymanager.cmd -import. When prompted for an alias, specify VerisignTestCA as the name to store this CA as. This name is simply an alias for this certificate. When prompted for the certificate file to import, specify the getcacert.cer file.

9. Import your certificate into your keystore.

   To import your public certificate (which you received from Verisign in step 8) into your keystore, run pskeymanager.cmd -import. When prompted for an alias specify the same alias you did when you created your private key and cert request in step 4. When prompted for the certificate file to import, specify your certificate file, *webservername*-cert.pem.

Modifying the Web Container to Support SSL

To complete the configuration between Web server plug-in and Web Container, the WebSphere Web Container must be modified to use the previously created self-signed certificates.

The following steps document the required Web Container modifications:

1. Start the WebSphere Administration Console, then after login, select Security, SSL.
2. Click New to create a new entry in the repertoire. Provide the following values to fill out the form:
   - Alias: Web Container SSL
   - Key File Name: *PS_HOME*\webserv*cellname_nodename_servername*\peoplesoft.ear\keystore\pskey
   - Key File Password: password
   - Key File Format: JKS
   - Trust File Name: *PS_HOME*\webserv*cellname_nodename_servername*\peoplesoft.ear\keystore\pskey
   - Trust File Password: password
   - Trust File Format: JKS
   - Client Authentication: not selected
   - Security Level: HIGH
3. Click OK when you have finished.
4. Save the configuration in the WebSphere Administration Console.
5. Select Servers -> Application Servers, then select the server you want to work with, in this case: server1.
6. Select the Web Container under the server.
7. Select HTTP transports under the Web Container.
8. Select the entry for the transfer you want to secure and click the item under the Host column.

   Select the * (asterisk) in this case in the line where 9443 is the Port.
9. In the configuration panel, select the Enable SSL box and select the desired SSL entry from the repertoire on the SSL drop-down list, in our case Web Container SSL.
10. Click OK, then save the configuration for WebSphere.

Testing Your Setup

Re-start WebSphere server and invoke https://*machine-name*:9443/ps/signon.html. You should see the Sign on page.

Setting Up SSL on IBM HTTP Server

This section provides an overview of the prerequisites for setting up SSL and discusses how to:

- Create key files for SSL.
- Add the certificate authority for PeopleTools testing.
- Create a personal certificate request.

- Send the certificate request to a certification authority for enrollment.
- Add the new certificate from the Enrollment page.
- Edit the httpd.conf file.
- Test the setup.

describes how to set up the SSL on IBM Http Server 1.3.28. This proxy is supported in WebSphere 5.1.1.

Prerequisites for Setting Up SSL

Before you set up SSL, you must have installed the following:

- IBM HTTP Server installed.
- IBM HTTP Server shutdown.

Creating Key Files for SSL

To create key files for SSL:

1. Create a folder named myKeys in your C:\IBM HTTP Server\ directory.
2. Start the Key Management Utility.
   - Windows: Start, Programs, IBM HTTP Server, Start Key Management Utility
   - UNIX: At a command prompt, enter `ikeyman`.
3. To run on AIX, HP, or Linux, or to use another JDK on Solaris, set your system environment using the following guidelines:
   - Set the JAVA_HOME variable as follows:

     EXPORT JAVA_HOME=the JDK (1.4 or higher) home directory full path name. If just IBM Http Server is installed from from WebSPhere Install guide , then you can find java in IBMHTTPServer Home directory. You can set it as JAVA_HOME. If WebSphere is installed on same machine , then set JAVA_HOME to WebSphere's java located at WAS_HOME/java.
   - Update the PATH variable as follows:

     EXPORT PATH = *JDK_home_full_path*/jre/sh:*JDK_home_full_path*/sh:$PATH
   - If you want the ability to run IKEYMAN from any directory, add the path where IKEYMAN is installed to your PATH environment variable:

     EXPORT PATH=$IKEYMAN_HOME/bin:$PATH

     The IBM Key Management Utility window should pop up.
4. To create a new Key Database, navigate to Key Database File, New.
5. Enter the following values to create a new Key Database:
   - Key database type: CMS key database file
   - File Names: key.kdb
   - Location: C:\IBM HTTP Server\myKeys
6. Click OK.

   After clicking OK, you get a Password Prompt window.
7. Enter a password and click "Stash the password to a file."

8. Click OK.

   You should see the awindow if you have generated the key database correctly and the password has been saved to a file.

9. Click OK.

Adding the Certificate Authority for PeopleTools Testing

This section describes how to add the Certificate Authority as a Signer Certificate using IBM Key Manager.

1. Download the PeopleTools Certificate Authority from respective vendor.
2. From the Key Database Content field, select Signer Certificates.
3. Click Add to add root certificate as a Signer Certificate.
4. Enter the following values to Add a CA certificate:
   - Data type: Binary DER data
   - Certificate file name: PWONG031000_PeopleTool.cer (example)
   - Location: C:\IBM HTTP Server\MyKeys\
5. Click OK.
6. Enter the name that you want to call this signer certificate.

   You should see the name that you entered as a Signer Certificate.

Creating a Personal Certificate Request

Use the following steps to create a personal certificate request:

1. In the Key database content field, select Personal Certificate Requests.
2. Click New.
3. Populate the Key Label and Organization as follows:
   - Key Label: Machine Name
   - Organization: MyCompany
4. Click OK.

   You should see a message that your request has been created.

5. Click OK.

Sending the Certificate Request to a Certification Authority for Enrollment

Use the following steps to send the certificate request to a certification authority for enrollment:

1. Using Notepad, open the request certreq.arm.

```
-----BEGIN NEW CERTIFICATE REQUEST-----
MIIBgjCB7AIBADBEMQswCQYDVQQGEwJVUzETMBEGA1UEChMKUGVvcGxlU29mdDEgMB4GA1UEAxMX
cHQtaWJtMDkucGVvcGxlc29mdC5jb20wgZ4wDQYJKoZIhvcNAQEBBQADgYwAMIGIAoGAeX/WO0A6
vmK/LUI+scTwpmbC87tHA7OyS8ULOF0ktt0BHmrOZxQQHUIMtjc3gL32RCN90cZsr4GntmUVAreC
```

```
DqqZTK69qX6IwY/KByTWdRcHeTPW/OSeAhrIf7kaP+DM/lXGOYBXJPBUQS0TP977TXW0c2TYdOIq
qpxyRMMV8KsCAwEAAaAAMA0GCSqGSIb3DQEBBAUAA4GBAHNHeMHAugG6Cm873oFTWGnkWBUkFlzJ
fNfEDIc8xZwWrLtWqiK45dV9C4k/0zUFAR0rFl/tROkAL2zstvVw5oV4/hM4XfnhfpJZyMIkk990
YiyM94hgwd8cAFaCctxYFHx8qwh1AtoywRQROgLrgSZHKlLpF+YBf5zCE1WFpUX2
-----END NEW CERTIFICATE REQUEST----
```

From the notepad file above, you can get the certificate from the respective vendor.

2. Click Submit and download the new certificate to your myKeys directory (C:\IBM HTTP Server\myKeys).

Adding the New Certificate from the Enrollment Page

Use the following steps to add the new certificate from the enrollment page:

1. From the IBM Key Manager, select Personal Certificates in the Key Database Content field.
2. Click Receive and enter the following values:
   - Data Type: Base64–encoded ASCII data
   - Certificate file name: Name of new cert from vendor
   - Location: C:\IBM HTTP Server\MyKeys
3. Click OK.

   You should see your personal certificate in the Personal Certificate box.

Editing the httpd.conf File

Edit the file /IBM Http Server/conf/httpd.conf in different operating systems.

Windows

The following is a sample of the file in Windows:

```
LoadModule ibm_ssl_module modules/IBMModuleSSL128.dll
Listen 443
<VirtualHost (your machine name):443>
SSLEnable
SSLClientAuth none
</VirtualHost>
SSLDisable
Keyfile "C:\IBM HTTP Server\myKeys\key.kdb"
SSLV2Timeout 100
SSLV3Timeout 1000
```

UNIX

The following is a sample of the file in UNIX:

```
LoadModule ibm_ssl_module libexec/mod_ibm_ssl_128.so
AddModule mod_ibm_ssl.c
Listen 443
<VirtualHost (your machine name):443>
SSLEnable
```

```
SSLClientAuth none
</VirtualHost>
SSLDisable
Keyfile /opt/IBMHTTPD/ssl/key.kdb
SSLV2Timeout 100
SSLV3Timeout 1000
```

Testing the Setup

Invoke the URL: https://localhost and you should be able to see the IBM Http Server page. To check the certificate, select File, Properties, Certificates to find out the validity of the certificate.

Using the WebSphere 5.1.1 Administration Console

This section discusses how to:

- Start and stop the administration console.
- Administer WebSphere 5.1.1.
- Run an application server as a non-root user.
- Set up IBM Http Server as a non-root user.
- Disable directory browsing in IBM HTTP server.

Starting and Stopping the Administration Console

You must first start the WebSphere server to access the administrative console. After you've finished working in the console, save your work and log out.

Use the following steps to start and stop the administrative console:

1. Start the administrative console.
    a. Verify that the application server for the administrative console is running : WAS_HOME/bin, startServer server1 , starts Server.
    b. In the same Web browser, type http://your_server_name:9090/admin, where your_server_name is the short or fully qualified host name for the machine containing the administrative server. When the administrative console is on the local machine, your_server_name can be localhost. On Windows platforms, use the actual host name if localhost is not recognized.
    c. Wait for the console to load into the browser.

    If you cannot start the administrative console because the console port conflicts with an application already running on the machine, change the port number in the install_root/config/cells/cell_name/nodes/node_name /servers/server_name/server.xml and install_root/config/cells/cell_name/virtualhosts.xml files. Change all occurrences of port 9090 (or whatever port was selected during installation of WebSphere Application Server) to the desired port for the console. Alternatively, shut down the other application that uses the conflicting port before starting the WebSphere Application Server product.
2. Log into the console when a Login page appears after the console starts.
    a. Enter your user name (or any user ID).

Changes made to server configurations are saved to the user ID. Server configurations also are saved to the user ID if there is a session timeout.

A user ID lasts for the duration of the session for which it was used to log in. If you enter an ID that is already in use (and in session), you are prompted to do one of the following: force the existing user ID out of session, wait for the existing user ID to log out or time out of the session, or specify a different user ID

b. If the console is secured, you must also enter a password for the user name. The console is secured if the following has been done for the console: specified security user IDs and passwords, or enabled global security.

c. Click OK.

3. (Optional). Stop the administrative console. Click Save on the console taskbar to save work that you have done and then click Logout to exit the console.

Note. If you close the browser before saving your work, when you next log in under the same user ID, you can recover any unsaved changes.

Administering WebSphere 5.1.1

The following are administrative tasks associated with WebSphere 5.1.1:

Start and Stop Respective Server

Use the following commands to start and stop the server:

- WAS_HOME/bin> startServer.bat(sh) *server_name*
- WAS_HOME/bin> stopServer.bat(sh) *server_name*

Troubleshooting

Use the following steps to troubleshoot:

1. Start WebSphere (WAS_HOME)/bin$startServer server1.
2. Open Admin console at URL http://*machine-name*:9090/admin,login.
3. Troubleshooting, Logs and Trace server_name, service_type.

Change WebSphere Http Ports

Use the following steps to change the http ports:

1. Start WebSphere (WAS_HOME)/bin$startServer server1.
2. Open Admin console at URL http://*machine-name*:9090/admin,login.
3. Servers>Application SErvers>server1>Web container>Http Transport : Change default 9080/9443 or admin port i.e 9090 ports.Clik OK and Save.
4. Environment>Virtual Hosts>default_host>Host Aliases : Change default 9080/9443 ports.Clik OK and Save
5. Environment>Update Web Server plugin. Click OK to Update Web server plugin.Clik OK and Save.
6. Logout Admin Console.

Set Verbose GC

Use the following steps to set verbose GC:

1. Open Admin console at http://*machine-name*:9090/admin

2. Open Admin console at http://*machine-name*:9090/admin.
3. Expand Servers > Application Servers > server1 > Process Definition > Java Virtual Virtual Machine.
4. Set check box Verbose garbage collection and bounce the respective server.

Change JVM Heap

Use the following steps to change the JVM heap:

1. Open Admin console at http://*machine-name*:9090/admin.
2. Login as any user
3. Expand Servers > Application Servers > server1 > Process Definition > Java Virtual Virtual Machine.
4. Enter Initial Heap and Max heap.
5. Save the configuration and Logout
6. Re-start WebSphere.

Add JVM System Properties

Use the following steps to add JVM system properties:

1. Open Admin console at http://*machine-name*:9090/admin
2. Login as any user
3. Expand Servers > Application Servers > server1 > Process Definition > Java Virtual Virtual Machine > Custom Properties.
4. Click New and add Name : HttpSessionIdReuse & Value as true.
5. Click New and add Name : com.ibm.websphere.cookies.no.header & Value as true
6. Save the configuration and Logout
7. Re-start WebSphere.

Regenerate Plug-in

Us the following steps to regenerate the plug-in:

1. Open Admin console at http://*machine-name*:9090/admin.
2. Log in as any user
3. Environment>Update Web Server plugin. Click OK to update the web server plug-in. Click OK and save.

Add Ports to Default_Host Virtual Host

Use the following steps to add ports to the default_host virtual host:

1. Open Admin console at http://*machine-name*:9090/admin.
2. Login as any user.
3. Environment>Virtual Hosts>default_host>Host Aliases : Add new ports with hostname as *.Clik OK and Save.

Create New Server

Use the following steps to create a new server:

1. Open Admin console at http://*machine-name*:9090/admin
2. Login as any user
3. Servers > Application servers > Click New Button> Enter Name of the Server(server2) > Click Finish to create > Save configuration > Logout.
4. Start/Stop Server server2 as WAS_HOME/bin$startServer server2 , stopServer server2

Set KeepAlive in WebSphere

Use the following steps to set keep alive in WebSphere:

1. Start WebSphere *WAS_HOME*/bin$startServer server1
2. Open Admin console at URL http://*machine-name*:9090/admin,login.
3. Servers>Application Servers>server1>Web container>Http Transport > Click http_port i.e 9080 > Custum Properties
4. You can set the following KeepAlive properties , values for the following are in integer.
   - ConnectionIOTimeout
   - ConnectionKeepAliveTimeout
   - MaxKeepAliveConnections
   - MaxKeepAliveRequests
   - KeepAliveEnabled

Configuring access/error logging for Internal Web Server HTTP transport

Use the following steps to configure acesss/error logging for internal web server HTTP transport:

1. Start WebSphere (WAS_HOME)/bin$startServer server1
2. Open Admin console at URL http://*machine-name*:9090/admin,login.
3. Servers>Application SErvers>server1>Web container>Http Transport > Click http_port i.e 9080 > Custum Properties
   - Property name: AccessLogDisable
   - Values: True/False
   - Default: Access log is disabled by default(False)

     The default access log file is WAS_HOME/logs/*server_instance*/http_access.log
   - Property name: ErrorLogDisable
   - Value: True/False
   - Default: Error log is disabled by default(False)

Running Application Server as a Non-root User

By default, WebSphere Application Server on UNIX platforms uses the root user ID to run Application Servers.

To configure an Application Server to run as non-root, complete the following steps:

1. Log on as root.
2. Create the user ID was1 with a primary system user group of wasgroup.

3. Reboot the machine.
4. Specify user and group ID values for the Run As User and Run As Group settings for a server:
    a. Go to the Process Execution page for the server you want to run as non-root.
    b. Click Servers > Application Servers > server1 > Process Definition > Process Execution and change these values:Property Value
    c. Run As User *wsadmin*.
    d. Run As Group *wsgrp* .
    e. UMASK 002.
5. Use following command to stop the server:

```
stopserver server1
```

6. As root, use operating system tools to change file permissions.

    The following examples assume that the WebSphere Application Server installation root directory is /opt/WebSphere/AppServer:

```
chgrp wasgroup /opt/WebSphere
  chgrp wasgroup /opt/WebSphere/AppServer
  chgrp -R wasgroup /opt/WebSphere/AppServer/config
  chgrp -R wasgroup /opt/WebSphere/AppServer/logs
  chgrp -R wasgroup /opt/WebSphere/AppServer/wstemp
  chgrp -R wasgroup /opt/WebSphere/AppServer/installedApps
  chgrp -R wasgroup /opt/WebSphere/AppServer/temp
  chgrp -R wasgroup /opt/WebSphere/AppServer/tranlog
  chgrp -R wasgroup /opt/WebSphere/AppServer/cloudscape50
  chgrp -R wasgroup /opt/WebSphere/AppServer/cloudscape51
  chgrp -R wasgroup /opt/WebSphere/AppServer/bin/DefaultDB
  chmod g+w /opt/WebSphere
  chmod g+w /opt/WebSphere/AppServer
  chmod -R g+w  /opt/WebSphere/AppServer/config
  chmod -R g+w /opt/WebSphere/AppServer/logs
  chmod -R g+w /opt/WebSphere/AppServer/wstemp
  chmod -R g+w /opt/WebSphere/AppServer/installedApps
  chmod -R g+w /opt/WebSphere/AppServer/temp
  chmod -R g+w /opt/WebSphere/AppServer/tranlog
  chmod -R g+w /opt/WebSphere/AppServer/cloudscape50
  chmod -R g+w /opt/WebSphere/AppServer/cloudscape51
  chmod -R g+w /opt/WebSphere/AppServer/bin/DefaultDB
```

7. Run the startServer command as wsadmin:

```
startserver server1
```

You can now start an Application Server as a non-root user.

Setting Up IBM Http Server to Run as a Non-root

Use the following steps to set up IBM Http Server to run as a non-root:

1. Stop IHS.

   You can stop IHS as root.

2. Change ownership.

```
# cd /products
# chown -R wsadmin:wsgrp  /opt/HTTPServer
```

3. Change permissions.

   Change files that have the permissions for owner set to '-wx' by issuing the following command:

```
# cd /opt/HTTPServer
# chmod -R u=rwx *
```

4. Edit the /opt/HTTPServer/conf/httpd.conf file by entering the following:
   - ServerName [system name].peoplesoft.com

     For example: ServerName ss-ibm13.peoplesoft.com
   - Port #

     For example: Port 80
   - User [WebSphere User]

     For example: User wsadmin
   - Group [WebSphere Group]

     For example: Group wsgrp
5. Save the httpd.conf file.
6. Start IHS (/opt/HTTPServer/bin$ ./apachectl start).
7. Test the Http Server by http://localhost

Disabling Directory Browsing in IBM HTTP Server

To disable browsing of a directory:

1. If you are using IBM HTTP Server 1.3.19.x as RPS with IBM WebSphere AEs 4.x, open the (IBM_HTTP_SERVER_HOME)/conf/httpd.conf file.
2. Search for the following:

```
<Directory "C:/IBM HTTP Server/cgi-bin">
AllowOverride None
Options Indexes
</Directory>
```

3. Change Options Indexes to Options None.

   After the change, you should have the following:

```
<Directory "C:/IBM HTTP Server/cgi-bin">
AllowOverride None
Options None
</Directory>
```

This setting prevents display of the directory structure on the browser when the IBM HTTP Server fails to find the respective file within the document root.

Using the Log Analyzer and Resource Analyzer

This section discusses how to:

- Use the Log Analyzer.
- Use the Resource Analyzer.

Using the Log Analyzer

Using the Log Analyzer, you can view the sevice or the WAS_HOME/logs/activity.log.

Use the following commands to launch the Log Analyser:

UNIX

```
install_root/bin/waslogbr
```

Windows

```
install_root\bin\waslogbr.bat
```

Using the Resource Analyzer

Tivoli Performance Viewer (resource analyzer) is a graphical user interface (GUI) performance monitor for the WebSphere Application Server.

Start Performance Monitoring

To start the performance monitoring frorm the command line, do the following:

Go to the product_installation_directory/bin directory and run the tperfviewer script. You can specify the host and port in Windows environments as:

```
tperfviewer.bat host_name port_number connector_type
```

On the AIX and other UNIX platforms, use the following:

```
tperfviewer.bat localhost 8879 SOAP
```

Enable Data Collection in the Administrative Console

To enable data collection in the administrative console, you need to select the Performance Monitoring Infrastructure (PMI) modules and levels that you want to monitor.

Use the following steps to enable data collection in the adminsitrative console:

1. Open the administrative console.
2. Click Servers, Application Servers in the console navigation tree.
3. Click Server.

4. Click the Runtime tab.
5. Click Performance Monitoring Service and select the PMI modules and levels to set the initial specification level field.
6. Click Apply or OK.
7. Click Save.

Uninstalling PIA on WebSphere

Use the following steps to uninstall PIA on WebSphere:

1. Open Admin console at http://*machine-name*:9090/admin
2. Login as any user
3. Expand Applications>Enterprise Applications.
4. Select the check boxes for the respective PIA applications to uninstall, click Stop
5. Select the check boxes for the respective PIA applications, click Uninstall
6. Save configuration.
7. Stop WebSphere server.
8. Delete directory PS_HOME/WebServ

Note. If you just delete PS_HOME to uninstall PIA without uninstalling it from WebSphere Admin Console, WebSphere registry will get corrupted . In such cases , subsequent PIA install will fail.

Installing Additional PIA to Existing WebSphere 5.1.1 Instance

Use the following steps to install additional PIA to existing WebSphere 5.1.1 instance:

1. Start the server1(WAS_HOME/bin, startServer.bat(sh) server1)that runs the administration console.
2. Open WebSphere Administration Console at URL http://machine_name:9090/admin (9090 is default admin port)
3. Create new PIA Server.
    a. Log in as any user.
    b. Navigate to Servers, Application servers. Click the New Button. Enter Name of the Server (PIAServer). Click Finish to create. Save configuration then log out.
    c. Stop server1 (WAS_HOME/bin> stopServer.bat(sh) server)
    d. Start PIAServer Start (WAS_HOME/bin$startServer PIAServer)
4. Run PIA install and enter PIAServer to deploy PIA to it.
5. Stop PIAServer (WAS_HOME/bin> stopServer.bat(sh) PIAServer).
6. Start PIAServer to log on to PIA. (Refer WAS_HOME/logs/PIAServer/SystemOut.log to view http/https listening ports on which PIA listens).
7. Log in.

Deploying Multiple PIA Instances on WebSphere 5.1.1

This section provides an overview of multiple PIA instances and virtual hosting, and discusses how to:

- Use multiple WebSphere base instances.
- Use a single WebSphere base instance.
- Use WebSphere ND.

Understanding Multiple PIA Instances and Virtual Hosting

Web applications running in a server are each rooted at a unique base URL, called a *context root*. PIA is deployed with a context root of '/'. This cannot be changed. Deploying multiple PIA instances on WebSphere can cause conflicts because their context roots are the same. Having two or more applications with the same URL makes some of the applications inaccessible.

WebSphere enables the use of virtual hosts to create unique URLs. This is accomplished with either a unique hostname or a unique port. The WebSphere virtual host should include the hostname and ports (for example 80 and 443) for the HTTP server, which enables the HTTP request to be forwarded to WebSphere through the plugin.

This section outlines examples of five different scenarios using WebSphere with multiple instances of PIA.

Note. Only the last two scenarios, which use WebSphere ND, are supported and recommended for use in a production environment.

Virtual Hosting

Virtual hosts enable the administrator to isolate and independently manage multiple sets of resources on the same physical machine.

Suppose an Internet Service Provider (ISP) has two customers whose Internet sites it would like to host on the same machine. The ISP would like to keep the two sites isolated from one another, despite their sharing a machine. The ISP could associate the resources of the first company with VirtualHost1 and the resources of the second company with VirtualHost2.

Now suppose both company's sites offer the same servlet. Each site has its own instances of the servlet, which are unaware of the other site's instances. If the company whose site is organized on VirtualHost2 is past due in paying its account with the ISP, the ISP can refuse all servlet requests that are routed to VirtualHost2. Even though the same servlet is available on VirtualHost1, the requests directed at VirtualHost2 will not be routed there.

The servlets on one virtual host do not share their context with the servlets on the other virtual host. Requests for the servlet on VirtualHost1 can continue as usual, even though VirtualHost2 is refusing to fill requests for the same servlet.

You can find more information about virtual hosting on the websites of IBM, Microsoft, and Sun Microsystems.

See Also

http://www-306.ibm.com/software/webservers/httpservers/doc/v1328/htdocs/en_US/manual/mod/core.html#namevirtualhost

http://www.microsoft.com/resources/documentation/WindowsServ/2003/standard/proddocs/en-us/Default.asp?url=/resources/documentation/WindowsServ/2003/standard/proddocs/en-us/wsa_hostmultwebsite.asp

http://docs.sun.com/source/816-5682-10/esvirtua.htm

Using Multiple WebSphere Base Instances

Examples 1 and 2 demonstrate how to deploy multiple PIA instances to multiple WebSphere base servers, one example without and the other with an HTTP server.

Note. These scenarios aren't supported in a production environment.

Example 1 — Without an HTTP Server

To configure this scenario:

1. Install WebSphere Base 5.1.1 using the installation documentation provided by PeopleSoft.
2. Start server1 from a command prompt.
   - On Windows:

     WAS_HOME\bin\startServer.bat server1
   - On Unix:

     WAS_HOME/bin/startServer.sh server1
3. Access the WebSphere Administration Console in a browser:

   http://localhost:*port*/admin
4. Expand Servers, Application Servers.
5. Click New.
6. Enter a value in the Server Name textfield.

   For example, *server2*.
7. Click Next, Finish, Save.
8. Expand Servers, Application Servers to verify that server2 was created.
9. Start server2 from a command prompt.
   - On Windows:

     WAS_HOME\bin\startServer.bat server2
   - On Unix:

     WAS_HOME/bin/startServer.sh server2
10. Install the first PIA instance on server1.

    Make sure you provide the following configuration information:
    - Specify a unique *PS_HOME* location.

- Select *server1* as the server name.
- Specify a unique application name (for example, peoplesoft1).
- Specify server1's internal HTTP transports for the HTTP port and HTTPS port.

11. Install the second PIA instance on server2.

    Make sure you provide the following configuration information:

    - Specify a unique *PS_HOME* location.
    - Select *server2* as the server name.
    - Specify a unique application name (for example, peoplesoft2).
    - Specify server2's internal HTTP transports for the HTTP port and HTTPS port.

12. Stop and restart both WebSphere instances (server1 and server2).
13. Access the PIA instances on the internal http tranports.

    - For the first PIA instance:

      http://*server_name*:9080/ps/signon.html
    - For the second PIA instance:

      http://*server_name*:9081/ps/signon.html

Example 2 — Adding an HTTP Server

The HTTP Server uses the plugin-cfg.xml file to route requests to WebSphere.

To configure this scenario:

1. Edit the default_host Virtual Host.
    a. Remove any entries for *.*.
    b. For port 80 entries, specify a hostname (for example, crm.peoplesoft.com).
    c. Add a port 443 entry if using SSL.
2. Create a new Virtual Host.
    a. Expand Environment, Virtual Hosts.
    b. Click New.
    c. Enter a name for the Virtual Host (for example, hr_host).
    d. Click OK.
3. Select the virtual host you just created.
4. Define a host alias for each server port.
    a. Click Host Aliases.
    b. Click New.
    c. Enter a hostname (for example, hr.peoplesoft.com) and a port (for example, 80).
    d. Repeat steps b and c to include all HTTP server ports (for example, 80 and 443).
    e. Click OK.
5. Expand Applications, Enterprise Applications.

    For all PIA instances that don't use the default virtual host, make the following change:

a. Select a PIA instance that doesn't use the default_host virtual host (for example, peoplesoft2).

b. Clear the Use Metadata From Binaries check box.

c. Click OK.

Save your changes.

6. Specify virtual host mapping.

a. Expand Applications, Enterprise Applications, and select the second PIA instance.

b. Click Map Virtual Hosts to Web Modules.

c. Select all of the web modules.

d. For each web module, select the new virtual host from the drop down menu.

e. Click OK and save your changes.

7. Regenerate the plugin.

a. Expand Environment, Update Web Server Plugin.

b. Click OK to regenerate the plugin.

8. Restart the WebSphere servers.

9. Edit the HTTP server's configuration to enable name-based virtual hosting.

Following is the procedure for the IBM HTTP Server. Refer to the Microsoft and Sun Microsystems websites for information about completing this configuration for IIS and SunOne, respectively.

See Chapter 7, "Working with IBM WebSphere," Understanding Multiple PIA Instances and Virtual Hosting, page 151.

a. Open the httpd.conf file in a text editor.

b. Find the section with the <VirtualHost> tag.

c. Each server name that will be used to access a PIA instance needs to be added to httpd.conf. Add the following lines with the correct ServerName values (crm.peoplesoft.com and hr.peoplesoft.com are examples):

```
NameVirtualHost *
<VirtualHost *>
ServerName crm.peoplesoft.com
</VirtualHost>
<VirtualHost *>
ServerName hr.peoplesoft.com
</VirtualHost>
```

d. Save your changes, and restart IHS to implement them.

e. Add aliases for these servernames to DNS servers for the site.

Using a Single WebSphere Base Instance

Example 3 demonstrates how to deploy multiple PIA instances to a single WebSphere base server.

Note. This scenario isn't supported in a production environment.

Example 3 — Basic Configuration

To configure this scenario:

1. Install WebSphere Base 5.1.1 using the installation documentation provided by PeopleSoft.
2. Start server1 from a command prompt.
   - On Windows:

     WAS_HOME\bin\startServer.bat server1
   - On Unix:

     WAS_HOME/bin/startServer.sh server1
3. Install the first PIA instance.

   Make sure you provide the following configuration information:
   - Specify a unique *PS_HOME* location.
   - Select *server1* as the server name.
   - Specify a unique application name (for example, peoplesoft1).
   - Specify server1's internal HTTP transports for the HTTP port and HTTPS port.
4. Install the second PIA instance.

   Make sure you provide the following configuration information:
   - Specify a unique *PS_HOME* location.
   - Select *server1* as the server name.
   - Specify a unique application name (for example, peoplesoft2).
   - Specify server1's internal HTTP transports for the HTTP port and HTTPS port.
5. Access the WebSphere Administration Console in a browser:

   http://localhost:*port*/admin
6. Edit the default_host Virtual Host.
   a. Expand Environment, Virtual Hosts.
   b. Click Default Host.
   c. Click Host Aliases.
   d. Remove any entries for *.*.
   e. For the internal transports (for example, 9080), specify a hostname (for example, crm.peoplesoft.com).
   f. Save your changes.
7. Create a new Virtual Host.
   a. Expand Environment, Virtual Hosts.
   b. Click New.
   c. Enter a name for the Virtual Host (for example, hr_host).
   d. Click OK.
   e. Select the virtual host you just created.
   f. Click Host Aliases.
   g. Click New.
   h. Enter a hostname (for example, hr.peoplesoft.com) and the internal transport (for example, 9080).
   i. Click OK, then save your changes.

8. Expand Applications, Enterprise Applications.

   For all PIA instances that don't use the default virtual host, make the following change:

   a. Select a PIA instance that doesn't use the default_host virtual host (for example, peoplesoft2).
   b. Clear the Use Metadata From Binaries check box.
   c. Click OK.

   Save your changes.

9. Specify virtual host mapping.

   a. Expand Applications, Enterprise Applications, and select the second PIA instance.
   b. Click Map Virtual Hosts to Web Modules.
   c. Select all of the web modules.
   d. For each web module, select the new virtual host from the drop down menu.
   e. Click OK and save your changes.

10. Stop and restart the WebSphere server1 instance.
11. Access the PIA instances on the internal http tranports.

    • For the first PIA instance:

      http://*server1_name*:9080/ps/signon.html

      Where *server1_name* is the hostname specified in the default_host (for example, crm.peoplesoft.com).

    • For the second PIA instance:

      http://*server2_name*:9080/ps/signon.html

      Where *server2_name* is the hostname specified in the hr_host (for example, hr.peoplesoft.com).

Using WebSphere ND

Examples 4 and 5 demonstrate how to deploy multiple PIA instances to multiple WebSphere ND servers, one eaxmple without and the other with an HTTP server.

Note. These scenarios are recommended for a production environment.

Example 4 — Without an HTTP Server

To configure this scenario:

1. Using PeopleSoft's Clustering and High Availability Red Paper, create two PIA ear files.

   These ear files should have unique *PS_HOME* and application name values. The ear files can be deployed to WebSphere clusters (as shown in the Red Paper) or to individual WebSphere application servers, as shown on the WebSphere Information Center website:

   http://publib.boulder.ibm.com/infocenter/ws51help/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/trun_app_instwiz.html

   Once the PIA instances are deployed, you must change the duplicate context roots using virtual hosts, as described in the following steps.

2. Access the WebSphere Administration Console in a browser:

   http://localhost:*port*/admin

3. Edit the default_host Virtual Host.
    a. Expand Environment, Virtual Hosts.
    b. Click Default Host.
    c. Click Host Aliases.
    d. Remove any entries for *.*.
    e. For the internal transports (for example, 9080), specify a hostname (for example, crm.peoplesoft.com).

       Create an entry with a hostname (for example, crm.peoplesoft.com) and the server's HTTP transport (for example, 9080).
    f. Save your changes.
4. Create a new Virtual Host for the second PIA instance.
    a. Expand Environment, Virtual Hosts.
    b. Click New.
    c. Enter a name for the Virtual Host (for example, hr_host).
    d. Click OK.
    e. Select the virtual host you just created.
    f. Click Host Aliases.
    g. Click New.
    h. Enter a hostname (for example, hr.peoplesoft.com) and the server's internal transport (for example, 9081).
    i. Click OK, then save your changes.
5. Specify virtual host mapping.
    a. Expand Applications, Enterprise Applications, and select the second PIA instance.
    b. Click Map Virtual Hosts to Web Modules.
    c. Select all of the web modules.
    d. For each web module, select the new virtual host from the drop down menu.
    e. Click OK and save your changes.
6. Restart the application servers and clusters to implement your changes.
7. Access the PIA instances on the internal http tranports.
    - For the first PIA instance:

      http://*server_name*:9080/ps/signon.html

      Where *server_name* is the hostname specified in the default_host (for example, crm.peoplesoft.com).
    - For the second PIA instance:

      http://*server_name*:9081/ps/signon.html

      Where *server_name* is the hostname specified in the hr_host (for example, hr.peoplesoft.com).

Example 5 — Adding an HTTP Server

To configure this scenario:

1. Define a host alias for each server port.
    a. Expand Environments, Virtual Hosts.

b. Select the default_host.

c. Click Host Aliases.

d. Click New.

e. Enter a hostname (for example, crm.peoplesoft.com) and a port (for example, 80).

f. Repeat steps d and e to include all HTTP server ports (for example, 80 and 443).

g. Click OK.

2. Repeat step 1 for the second virtual host (hr_host).
3. Regenerate the plugin.
    a. Expand Environment, Update Web Server Plugin.
    b. Click OK to regenerate the plugin.
4. Restart the application servers.
5. Edit the HTTP server's configuration to enable name-based virtual hosting.

    Following is the procedure for the IBM HTTP Server. Refer to the Microsoft and Sun Microsystems websites for information about completing this configuration for IIS and SunOne, respectively.

    See Chapter 7, "Working with IBM WebSphere," Understanding Multiple PIA Instances and Virtual Hosting, page 151.

    a. Open the httpd.conf file in a text editor.

    b. Find the section with the <VirtualHost> tag.

    c. Each server name that will be used to access a PIA instance needs to be added to httpd.conf. Add the following lines with the correct ServerName values (crm.peoplesoft.com and hr.peoplesoft.com are examples):

```
NameVirtualHost *
<VirtualHost *>
ServerName crm.peoplesoft.com
</VirtualHost>
<VirtualHost *>
ServerName hr.peoplesoft.com
</VirtualHost>
```

    d. Save your changes, and restart IHS to implement them.

    e. Add aliases for these server names to DNS servers for the site.

Reserving WebSphere 5.1.1 PIA Ports

PIA uses WebSphere server's http/https ports. When PIA is installed to respective server in WebSphere , its http and https ports assigned to applications like PIA are reserved to PIA. They should not be deleted. It is okay to modify ports. If you delete PIA http/https ports in WebSphere server , then you can't login into PIA page.

Note. Do not delete WebSphere PIA http/https ports from WebSphere 5.1.1 server.

Use the following steps to find out PIA ports usage from the Administration Console:

1. Start WebSphere (WAS_HOME)/bin$startServer server1

2. Open Admin console at URL http://*machine-name*:9090/admin,login. (9090 is default admin port)
3. Servers>Application Servers>*server_name*>Web container>Http Transport

Note. Do not delete http/https ports from the Admin Console.

In *WAS_HOME*\config\cells*cell_name*\nodes*node_name*\servers*server_name*\server.xml, HTTPTransport_1 is reserved for PIA http port and HTTPTransport_2 is reserved for PIA https port.

```
<threadPool xmi:id="ThreadPool_2" minimumSize="10" maximumSize="50"⇒
 inactivityTimeout="3500" isGrowable="false"/>
<transports xmi:type="applicationserver.webcontainer:HTTPTransport"⇒
 xmi:id="HTTPTransport_1" sslEnabled="false">
     <address port="9080" host="" xmi:id="EndPoint_1"/>
</transports>
<transports xmi:type="applicationserver.webcontainer:HTTPTransport"⇒
 xmi:id="HTTPTransport_2" sslEnabled="true"⇒
 sslConfig="RSHANKA2040303Node/DefaultSSLSettings">
     <address port="9443" host="" xmi:id="EndPoint_2"/>
</transports>
```

If HTTPTransport_1 is changed to lets say HTTPTransport_11 or any other number , then change it back to HTTPTransport_1 similarly if HTTPTransport_2 is changed , to lets say HTTPTransport_23 or any other number, then change it back to HTTPTransport_2

WebSphere Ports

Refer to the WebSphere installation guide for a list of WebSphere ports usage.

Accessing the WebSphere InfoCenter

The WebSphere InfoCenter provides detailed WebSphere 5.1.1 documentation on the web.

For the Version 5.1.1 InfoCenter in all supported languages:

See http://publib.boulder.ibm.com/infocenter/ws51help/index.jsp

For the entire InfoCenter set for all WebSphere Application Server products:

See http://www.ibm.com/software/webservers/appserv/infocenter.html

For the entire documentation set for all WebSphere Application Server products, including Adobe Acrobat PDF versions of the information:

See http://www.ibm.com/software/webservers/appserv/library.html

All of the Version 5.1.1 product documentation is in the V5.1.1 InfoCenter, including versions of the installed help files.

CHAPTER 8

Building and Maintaining Search Indexes

This chapter provides an overview of PeopleSoft search indexes and discusses how to:

- Work with indexes.
- Build record-based indexes.
- Build file system (spider) indexes.
- Build HTTP spider indexes.
- Administer search indexes.
- Modify the VdkVgwKey key.

Understanding PeopleSoft Search Indexes

This section provides an overview of search indexes and discusses:

- Types of indexes.
- Components of the search architecture.
- Index building.
- Search index limitations.
- User search strategies.

Overview of Search Indexes

A search index is a collection of files that is used during a search to quickly find documents of interest. The process of creating the search index is also called building the search index. The set of files that make up the index is a *collection.* This collection contains a list of words in the indexed documents, an internal documents table containing document field information, and logical pointers to the actual document files.

Fields contain metadata about a document. For example, Author and Title might be fields in an index. *VdkVgwKey* is a special field that identifies each document and is unique to all of the documents in the collection.

The document table is a relational table with one row for each document and columns of fields. Every index can be modified by defining a set of fields for it.

In PeopleSoft search implementations, every search index has a home location where all of the files pertaining to that index are located. This directory is the home directory of the index and is typically located at *PS_HOME*/data/search/*INDEXNAME*. You can change this location through application server and process scheduler configuration files. Under this directory is another directory named for the database to which the application server or the process scheduler is connected. The actual collection files reside in this database directory.

Every search index can be modified by changing the configuration files that are associated with the index. These configuration files are known as *style* files and reside in the style directory under the database directory. A typical configuration of style files define fields for a particular index.

Types of Indexes

PeopleSoft software supports three types of search indexes:

- Record-based indexes.
- HTTP spider indexes.
- File system indexes.

Record-Based Indexes

Record-based indexes are used to create indexes of data in PeopleSoft tables. For example, if the PeopleSoft application has a catalog record that has two fields (Description and PartID), you can create a record-based index to index the contents of the Description and PartID fields. Once the index is created, you can use the PeopleCode search application programming interface (API) to search this index.

HTTP Spider Indexes

HTTP spider indexes index a web repository by accessing the documents from a web server. You typically specify the starting uniform resource locator (URL). Then the indexer walks through all documents by following the document links and indexes the documents in that repository. You can control to what depth the indexer should traverse.

File System Indexes

File system indexes are similar to HTTP spider indexes, except that the repository that is indexed is a file system. You typically specify the path to the folder or directory. Then the indexer indexes all documents within that folder. HTTP spider indexes and file system indexes are sometimes collectively referred to as *spider* indexes. The indexer recognizes a wide variety of document formats, such as Word or Excel documents. Any document that is an unknown format will be skipped by the indexer.

Components of the Search Architecture

PeopleSoft search architecture uses two main technologies: that provided by the PeopleSoft Portal and that provided by Verity. They are connected by the PeopleSoft search API.

PeopleSoft Portal Technologies

The PeopleSoft Portal search technology contains the following components:

- Search input field.

  Captures a query string that is entered by users in the portal header.
- Search API.

Passes the query string that is captured in the search input field to the Verity search engine.

- Portal Registry API.

  Applies security to filter the search results.

- Portal registry.

  Contains a repository of content references that can be searched.

- Search results page.

  Formats and displays search results for the user.

- Search options.

  Enables users to personalize search behavior and results.

Note. By default, the PeopleSoft search performs case-insensitive searches.

Verity Technologies

The basic items of the Verity architecture that are incorporated in the PeopleSoft Portal search architecture are:

- Verity collection.

  This is the set of files forming a search index. When a user performs a search, the search is conducted against the Verity collection. You can create and maintain your own collections with the Search Design and Search Administration PeopleTools.

- BIF file.

  This is an intermediate file that is created in the process of building a Verity collection. The BIF file is a text file that is used to specify the documents to be submitted to a collection. It contains a unique key, the document size (in bytes), field names and values, and the document location in the file system.

- XML file.

  This is another intermediate file that is created in the process of building a Verity collection. The XML file is a text file named *indexname*.xml that contains all of the information from the documents that are searchable but not returned in the results list. This information is stored in zones. Zones are specific regions of a document to which searches can be limited.

- Style files.

  These files describe a set of configuration options that are used to create the indexes that are associated with a collection.

- mkvdk.

  This Verity command-line tool is used to:

  - Index a collection.
  - Insert new documents into a collection.
  - Perform simple maintenance tasks, like purging and deleting a collection.
  - Control indexing behavior and performance.

PeopleSoft Search Utilities

To create and administer search indexes for use with PeopleSoft software, use the PeopleTools utilities under PeopleTools, Search Engine. The utilities enable you to administer indexes and to create file system, spider, and record-based indexes.

Index Building

For both HTTP spider and file system indexes, options are available to include or exclude certain documents based on file types and Multipurpose Internet Mail Extensions (MIME) types. The index building procedure is different for record-based indexes and the spider indexes. Typically, the index building procedure is carried out from an Application Engine job that is scheduled by using the process scheduler.

The steps for building record-based indexes are:

1. The data from the application tables is read and two files called *indexname*.xml and *indexname*.bif are created.

   indexname.xml contains one XML record for each document that needs to be indexed. The XML record contains all of the data that needs to be indexed. *indexname*.bif contains field information, the VdkVgwKey document, and offsets to denote the start and end of each document in the XML file.
2. The XML and the bulk insert file (BIF) files are typically generated through PeopleCode and reside in the home location of the index. The Verity utility, mkvdk, is called, passing in the BIF file as the argument to build the index.

The steps for building spider indexes are:

1. The Verity utility, vspider, is called.

   The vspider utility takes a number of arguments, but the most important ones are the starting URL or directory to spider and the number of links to follow.
2. The vspider utility walks through all of the documents in the repository and builds the index.

Search Index Limitations

Following are the PeopleSoft search index limitations:

- Verity does not run on IBM z/OS.
- Verity collections must reside on the PeopleSoft application server or be accessible from it through a shared drive.

  Satisfying this requirement can take several forms, depending on the application server's operating system. On Microsoft Windows, this could be a network drive. On UNIX, this could be an NFS-mounted drive.
- Verity collections are most efficient if you index large groups of data, rather than indexing one or two documents at a time.

  Small updates degrade the index and require that you run the Verity cleanup utility.
- Style files are located in the style subdirectory of the index.

  To make style changes, apply them to the files in this directory.
- You can have only one language per collection.

Additionally, a number of Verity search index features are limited to certain maximum values, as follows:

Feature	Limitation
Wildcards	Wildcard auto-expansion is limited to 16,000 matches.
Number of collections	The maximum number of physical collections that can be searched at one time is 128.

Feature	Limitation
Documents per collection	The maximum number of documents allowed per collection is 16 million, subject to disk space availabilty.
Fields per collection	The maximum number of fields allowed per collection is 250.
Field length	The maximum length of any field is 32 kilobytes. **Note.** The actual number of characters that translates to depends on the character set being used.
Field value length in bulk files	The maximum length of a field value in a bulk file is 32 kilobytes. **Note.** The actual number of characters that translates to depends on the character set being used.
Zones per document	The number of zones allowed per doocument is unlimited.
Characters in path	The maximum path size allowed is 256 characters.
Maximum documents with sort specification	The maximum number of documents that are returned when a sort specification is applied is 16,000.
Sort fields per search	The maximum number of fields that can be included in a sort specification is 16.

Refer to the Verity documentation for details about these features.

User Search Strategies

A user submits a search request by entering a search string into the search input form field in the portal header. The "<form action=...>" element in the portal header is generated at runtime to link to a PeopleSoft Internet Architecture page, and a Java script submits the form. The query string is passed to the Search API as a parameter named PortalSearchQuery to find matching results. Those results are filtered for security through PeopleCode by the Portal Registry API. The search results page echoes the original query string and displays a list of content references that match the request. If the user clicks the Go button but does not enter a search query, the search results page displays without any results.

The search results page performs the following steps:

- Changes the case of the entered text to all uppercase characters.

  By default, the Verity search engine searches for all mixed-case variations when a query string is entered in all lowercase or in all uppercase. However, search queries that are entered in mixed-case automatically become case sensitive. (For example, a query on *Apple* behaves as if the user had specified *Apple,* which would find only the precise string *Apple,* while a query on *apple* finds *APPLE, Apple,* and *apple.*) But the portal makes one important change: It changes the case of the query sting to all uppercase, prohibiting users from truly executing case-sensitive searches. This avoids situations where mixed-case searches would otherwise return no results. On the search results page, however, the original case is echoed back to the user.

- Formats the query string to pass to the Search API.

  This includes filtering out expired and hidden content reference, and content references that are not valid yet.

- Calls the Search API.

  This returns the query results.

- Calls the Portal Registry API.

  This is done to apply security filtering to the results. Security is applied in PeopleCode by checking the Authorized property.

- Formats and displays search results.

  This completes the user's search request.

Working with Indexes

This section provides overviews of common controls and supported MIME types, and discusses how to:

- Open existing collections.
- Create new collections.

Understanding Common Controls

The following controls appear on the pages that are used for designing record-based, file system, or HTTP spider indexes.

Index	Shows the name of the index that you opened or the name that you gave the index on the Add New Value page.
Build Index	Invokes the collection build program. Before clicking this button, select all of the appropriate options for the collection.
Test Index	After building an index, click to test that the build program assembled the index properly. The Test Index page contains a single text field with a query button. Enter text to search for in the collection and click the [?] button to submit the query. The results return a list of the keys that are stored by Verity in the collection.
Show Logs	View the log files that are produced by the collection build program during execution. This is used mainly for troubleshooting.
Append to Verity Command Line	This control is for PeopleSoft internal use only.

Understanding Supported MIME Types

The following list contains the supported document MIME types. Any document that is not one of these types is ignored during the indexing process.

- application/msword
- application/wordperfect5.1
- application/x-ms-excel
- application/x-ms-powerpoint
- application/x-ms-works
- application/postscript

- application/rtf
- application/x-lotus-amipro
- application/x-lotus-123
- application/x-ms-wordpc
- application/x-corel-wordperfect
- application/x-wordprocessor
- application/x-spreadsheet
- application/x-presentation
- application/x-graphics
- application/x-keyview
- application/x-ms-write
- application/pdf
- application/x-executable
- message/rfc822
- message/news
- text/html
- text/sgml
- text/xml
- text/ascii
- text/enriched
- text/richtext
- text/tab-separated-values
- text/plain
- text/x-empty
- image/gif
- application/x-verity

Opening Existing Collections

To open an existing collection:

1. Select PeopleTools, Search Engine.
2. From the available menus, select the type of collection that you want to open, as in record-based indexes, file system indexes, or HTTP spider indexes.
3. On the Find an Existing Value tab, use the Search for drop-down list box to select the appropriate criteria (begins with or contains).
4. In the edit box to the right, enter the character string that reflects the appropriate begins with or contains criteria.
5. Click Search.

Creating New Collections

To create a new collection:

1. Select PeopleTools, Search Engine.
2. From the available menus, select the type of collection that you want to create, as in record-based indexes, file system indexes, or HTTP spider indexes.
3. Select the Add a New Value page.
4. Enter a name for the collection.
5. Click Add.
6. Specify the appropriate attributes for the collection as described in the following sections.
7. Save your work.

   Note. You cannot create indexes of the same name even if they are of different types; for example, record, HTTP, or file.

8. Build the index.

Building Record-Based Indexes

The record-based index extracts data from database tables and inserts the data into BIF and XML files, which are then indexed by Verity. The individual creating the index chooses the records (tables) to be indexed.

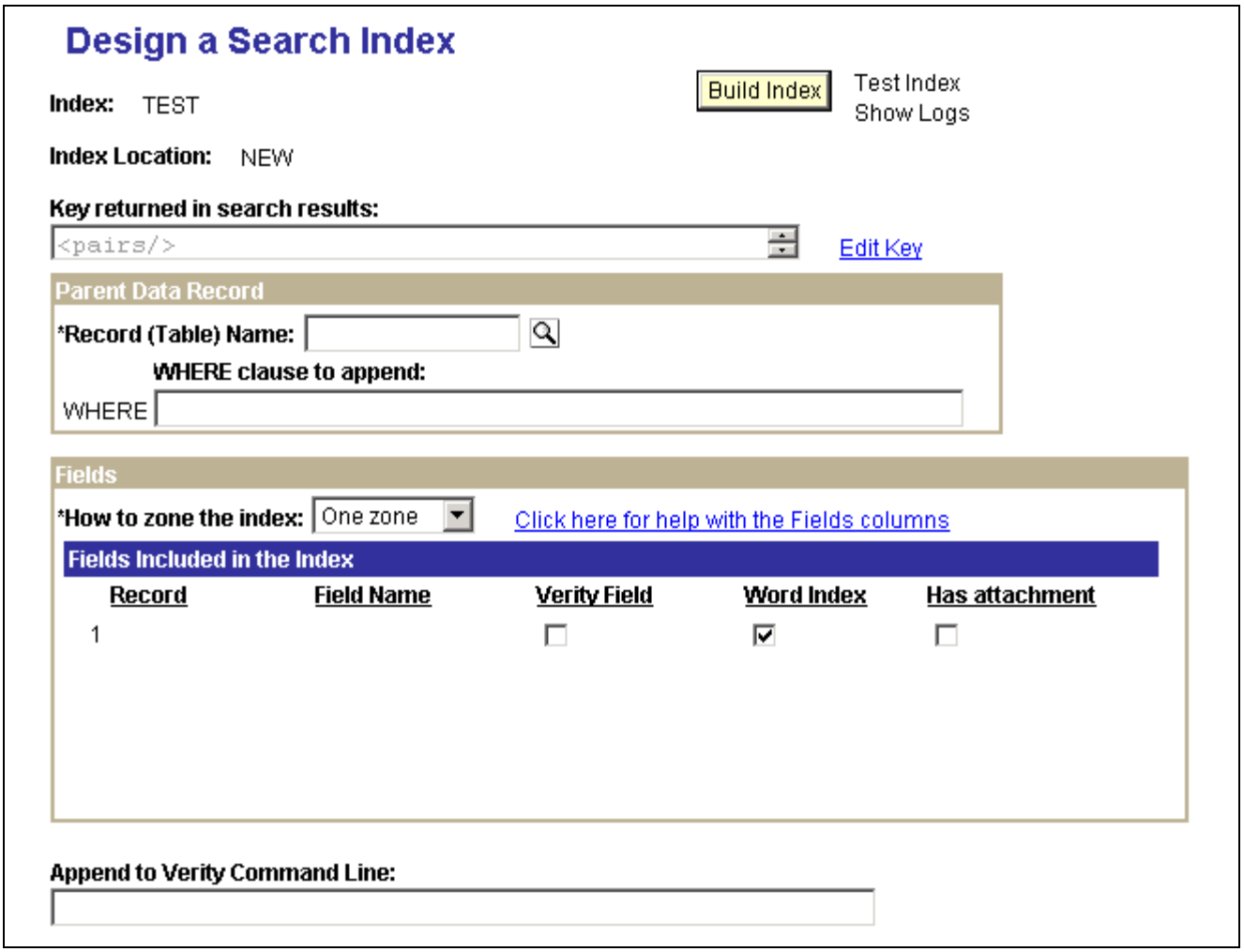
Note. The record-based index supports only data that is stored in PeopleSoft databases.

This section discusses how to:

- Modify record-based index properties.
- Add subrecords to search indexes.

Modifying Record-Based Index Properties

Select PeopleTools, Search Engine, Record-Based Indexes to access the Design a Search Index page.



Design a Search Index page

Parent Data Record

Record (Table Name) Enter tables, views, or a PeopleSoft view that contains data. To combine the data from multiple PeopleSoft tables, to create a view on those tables and specify the name of that view here.

WHERE clause to append Fine-tune the data that you receive by entering a Structured Query Language (SQL) WHERE clause.

Key returned in search results Use to synthesize the VdkVgwKey, which supports an XML-like syntax enabling you to modify the tag that is returned by Verity.

You have the following options:

- *<pairs/>:* Inserts a string of NAME=VALUE;. One such pair is returned for each key of the record.
- *<row/>:* Inserts the record keys in a SQL-like syntax.
- *<field fieldname='MYFIELD'/>:* Inserts the value of MYFIELD if it exists in the record.
- *<sql stmt='SQL STATEMENT'/>:* Inserts the value that is returned by the SQL statement. The system accepts only the first row that is returned, and

PeopleSoft software does not support SQL statements returning more than one column.

Edit Key Click to access the page where you can change the results that are returned by the Key returned in search results functionality.

Fields

How to Zone the Index

One Zone: Select to put all of the data into one zone. With this option, the collection builds more quickly but the application can't restrict searches to the portions of the index that come from a particular field.

Field Zones: Select to create one zone for each PeopleSoft field on the record. Applications can specify that they want to access that particular zone in their searches.

Field Name After you specify a record name, the fields in that record appear in this grid. Select the following options for each field in the record: Verity Field, Word Index, or Has Attachment (each option is explained in the following sections).

Verity Field Select if the PeopleSoft field should be indexed as a Verity field. In general, PeopleSoft fields that contain a lot of descriptive text, such as description fields, should be indexed as word indexes (See the following definition) and PeopleSoft fields that contain metadata about what is being indexed (such as ProductID) should be indexed as Verity fields.

Word Index Select if this PeopleSoft field should be indexed as a word index. See the preceding Verity Field definition for guidelines on defining a PeopleSoft field as a Verity field versus defining it as a word index.

Has attachment Enables you to index attachments that are referenced in the field as uniform resource identifiers (URIs). Refer to the PeopleCode Developer's Guide for a description of file attachments. If this field contains the URL to an attachment, select this check box. The indexer downloads the attachment and indexes it as part of the document. This item is enabled only if the corresponding PeopleSoft field contains character data, because numeric fields cannot contain URLs.

To use this field, you need a record that is designed with this feature in mind. In the record, each row has a text field that contains a URI or an empty string.

The text must be a valid File Transfer Protocol (FTP) URI (including the login and password string) of the following form:

- ftp://user:pass@host/path/to/filename.doc.
- A valid record URI of the form record://RECORDNAME/path/to/file.doc.
- A string of the form <urlid name="A_URLID"/>/path/to/file.doc.

The third form references an entry in the URL table (Utilities, Administration, URLs). If the URL ID that is named in the name attribute is valid, the entire URI is rewritten with the part in brackets replaced by the actual URI.

For example, if A_URLID is equal to ftp://anonymous:user@resumes.peoplesoft.com, the entire string in the previous example becomes ftp://anonymous:user@resumes.peoplesoft.com/path/to/file.doc and is treated like any other FTP URI.

Rows of data with empty strings in the URI field are ignored with no error.

If the string is one of these three valid URI forms and a document can be retrieved at that URI, the document is indexed with the same key as the rest of the row of data and is searchable.

To add subrecords to the index, select the Subrecords tab, and insert the child records that you want to include in the index.

Adding Subrecords to Search Indexes

Select PeopleTools, Search Engine, Record-Based Indexes, Subrecords.

To index more than one record as a single document, the records must be hierarchically related. For example, the record that is specified on the previous page must be a parent of all the others. Formally, this means that the keys of each subrecord named must be a superset of the keys of the parent record. The parent record is the one that you specify in the Record (Table Name) field on the Primary Record page.

To add subrecords to an index:

1. Create and save the index definition.
2. Select PeopleTools, Search Engine, Record-Based Indexes, Subrecords.
3. Click the Add a new row button to insert the names of the records that are children of the parent record that is defined on the Primary Record page.

   On the Primary Record page, the fields of the child record are added to the Fields grid. When you build the index, data from the child records whose keys match the row in the parent record is included as part of the parent record. When an end user searches for data that is found in the child record, the system returns a reference (VdkVgwKey) for the parent record.

Building File System (Spider) Indexes

You can index file systems that are local to the application server. This refers to any file system on the physical server on which your application server domain runs, and it also refers to any drives that are accessible from the application server machine. File systems might include file servers, report repositories, and so on.

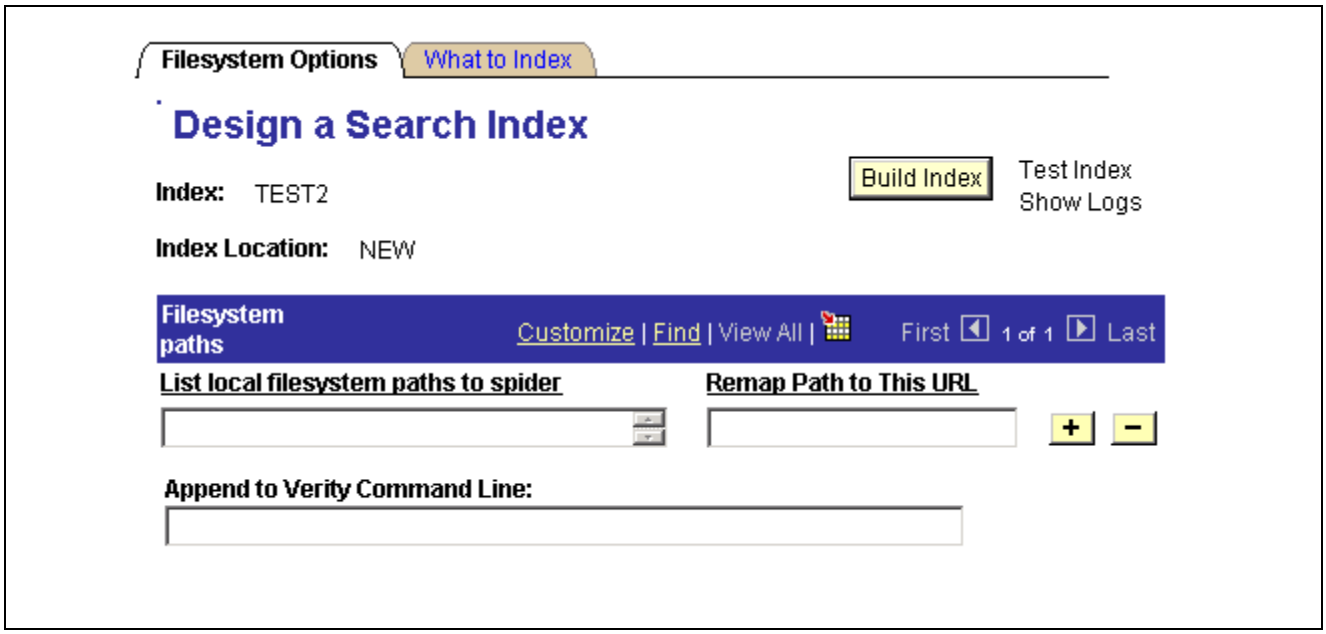
The index is compiled by using vspider. The program descends into the directory structure recursively and indexes the file types that you've selected to be indexed. It indexes only files that Verity supports for collections.

This section discusses how to:

- Set file system options
- Define what to index

Setting File System Options

Select PeopleTools, Search Engine, Filesystem Indexes to access the Filesystem Options page.



Filesystem Options page

List local filesystem paths to spider Specify the network file system path that contains the documents to index. Ensure that the local application server has the proper access to the file systems that you include in the list.

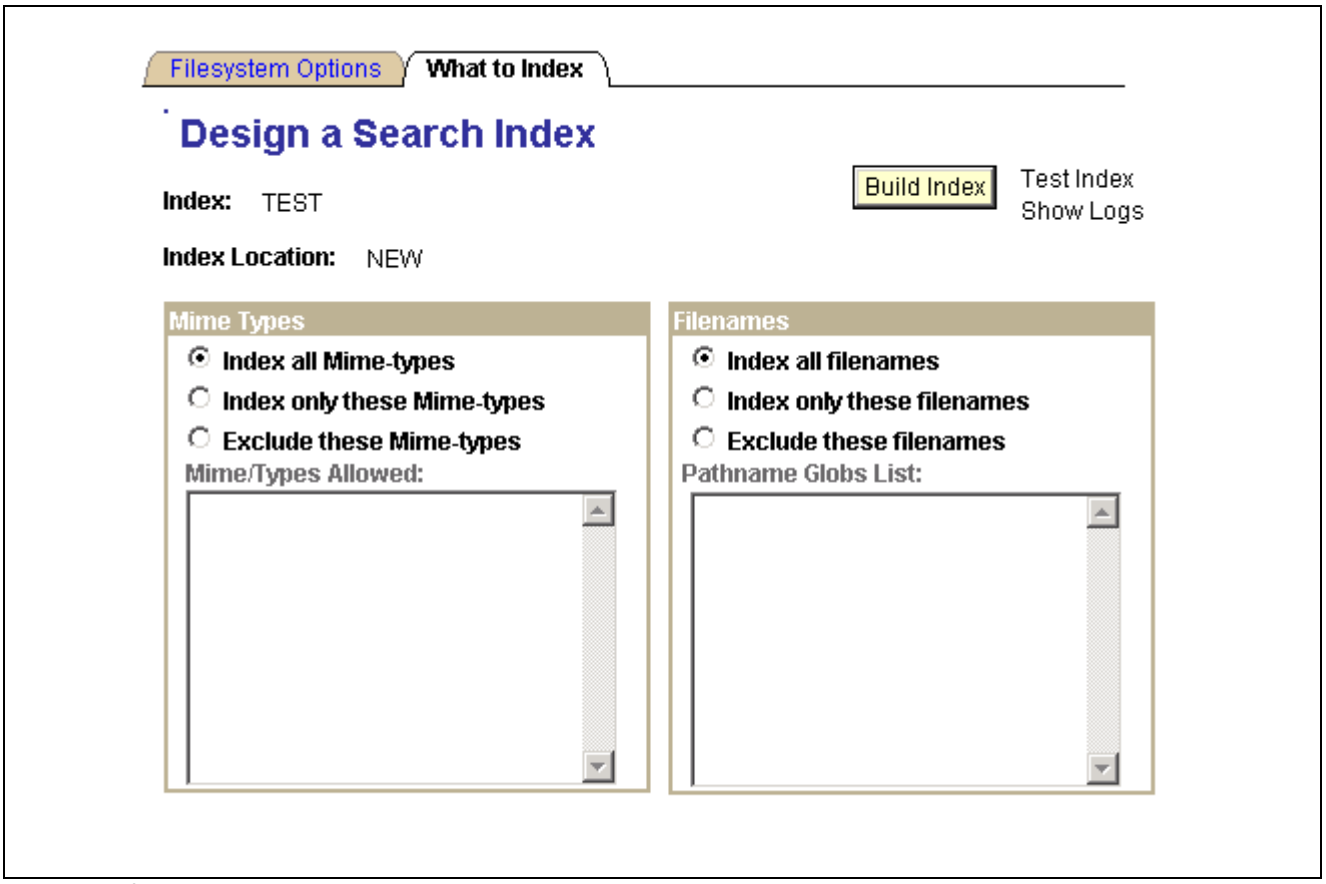
For Microsoft Windows, this means the drive mappings must be set up from the applications server. For UNIX, this means the correct network file system (NFS) mappings must be set on the application server.

To add a system path to the list, click the plus button. To remove a file system, click the minus button.

Remap Path to This URL Do not use.

Defining What to Index

Select PeopleTools, Search Engine, Filesystem Indexes, What to Index to access the What to Index page.



What to Index page

MIME Types

Index all Mime-types Select to index all MIME types on a website.

Index only these Mime-types Select to index only a certain MIME type, and specify the file type in the MIME/Types Allowed list box. Separate multiple MIME types with a space.

Exclude these Mime-types Select to exclude a set of MIME types, and specify the MIME types to exclude. Separate multiple MIME types with a space.

MIME/Types Allowed Add a list of MIME types, separated by spaces, if you selected Index only these Mime-types or Exclude these Mime-types.

Filenames

Index all filenames Select to index all file types.

Index only these filenames Select to index only a certain file type, and specify the file type in the Pathname Globs List list box.

Exclude these filenames Select to exclude a set of file types, such as temporary files, but to index all others. Also specify the file types to exclude.

Pathname Globs List Add the files that you want to incorporate into your index. Separate the entries with spaces. You can use wildcard characters (*) to denote a string and "?" to denote a single character. For example, the string '*.doc 19??.excel' means

select all files that end with the ".doc" suffix and Microsoft Excel files that start with 19, followed by 2 characters.

Building HTTP Spider Indexes

HTTP spider indexes are similar to the indexes that the spider functionality compiles for the file system index. When using the spider index on a website, vspider starts at the home page of the site and then follows each link on that page to the next level of the site. For each page at the next level, vspider follows each link on each page. After following a link, vspider indexes all of the data on the target page.

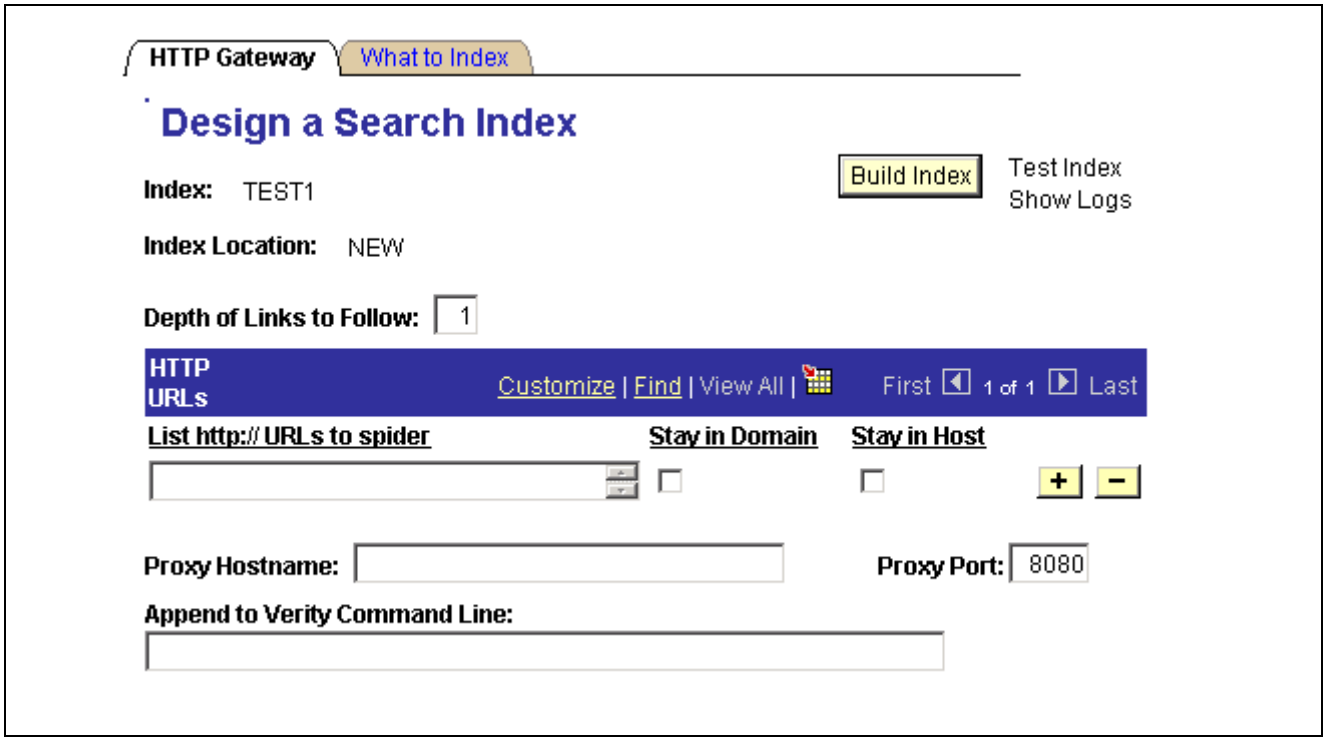
You can specify as many websites as you want, and you can configure the depth, or number of layers of links, that vspider follows into a website and index.

This section discusses how to:

- Define HTTP gateway settings.
- Define what to index.

Defining HTTP Gateway Settings

Select PeopleTools, Search Engine, HTTP Spider Indexes to access the HTTP Gateway page.



HTTP Gateway page

Depth of Links to Follow Set the level of detail that you want to index within a certain site. If you enter *1,* vspider starts at the homepage and follows each link on that page and indexes all of the data on the target pages. Then it stops. If you enter *2,* vspider follows the links on the previous pages and indexes one more level into the website.

	As you increase the number, the number of links that vspider follows increases geometrically. Do not set this value too high, because it can impact performance negatively. You should not need to set this value higher than 10.
List http://URLs to spider	Click the plus button to add multiple URLs to spider. Click the minus button to remove a URL from the list. If you forget to include the *http://* (scheme) portion of the URL, the system automatically includes it. URLs should contain only the alphanumeric characters as specified in RFC 1738. Any special character must be encoded. For example, encode a space character as *%20,* and encode a < as *%3c.* Additional examples are available. See http://www.w3.org/Addressing/rfc1738.txt
Stay in Domain	Select to limit spidering to a single domain. For example, suppose that you are spidering www.peoplesoft.com and you select this option. If a link points to a site outside the PeopleSoft domain (as in yahoo.com), the collection ignores the link.
Stay in Host	Select to further limit spidering within a single server. If you select this option, the collection contains references to content only on the current web server or host. Links to content on other web servers within the domain are ignored. For example, if you are spidering www.peoplesoft.com and you select this option, you can index documents on www.peoplesoft.com, but not on www1.peoplesoft.com.
Proxy Hostnameand**Proxy Port**	Enter a host and port for vspider to use. Enter the same settings that you would use in your web browser if you need a proxy to access the internet.

Defining What to Index

Select PeopleTools, Search Engine, HTTP Spider Indexes, What to Index. The fields on this page are documented in a previous section.

See Chapter 8, "Building and Maintaining Search Indexes," Defining What to Index, page 172.

Administering Search Indexes

After you design and build your search indexes, the Search Administration interface enables you to schedule when and how frequently the indexes must be rebuilt. An important aspect of maintaining the collections involves scheduling PeopleSoft Process Scheduler jobs that, on a regular basis, rebuild the collection completely or incrementally update the index. Search index administration also includes deleting old indexes and building indexes to support additional languages.

This section discusses how to:

- Specify the index location.
- Administer the search index.
- Edit properties.
- Schedule administration.
- Share indexes between application servers and PeopleSoft Process Scheduler.

Specifying the Index Location

By default, the files for an index are located in *PS_HOME*/data/search/*indexname*/*db_name*/*language_code*. However, you can change this location by specifying the search index location property in the application server and process scheduler configuration files.

Set the search index location at the application server level in the application server configuration file, PSAPPSRV.CFG. This enables you to specify alternate search index locations for an application server, if necessary. You also need to set this property in the process scheduler configuration file, PSPRCS.CFG, to point to the same location as specified in the application server configuration file.

Note. You must manually edit the file to include the locations. You do not add search index locations by using PSADMIN.

To add a search index location on the application server:

1. Open the PSAPPSRV.CFG file for the appropriate application server domain.
2. Locate the Search Indexes configuration section.

   For example:

```
[Search Indexes]
;============================================================================
; Search index settings
;============================================================================
: Search indexes can be given alternate locations if there is an entry here.
; Entries look like: IndexName=fs location (ie EMPLOYEE=c:\temp)
```

3. Add an entry for each search index location that you want to specify for an application server by using the following syntax:

```
index_name=location
```

   For example, to specify the location for search INDEX_A and INDEX_B, your entries would look similar to the following:

```
[Search Indexes]
;============================================================================
; Search index settings
;============================================================================
: Search indexes can be given alternate locations if there is an entry here.
; Entries look like: IndexName=fs location (ie EMPLOYEE=c:\temp)
INDEX_A=c:\temp
INDEX_B=n:\search
```

   Note. Make sure that your entries are not commented out with a semicolon (;) appearing before them.

   ---

4. Save the PSAPPSRV.CFG file.

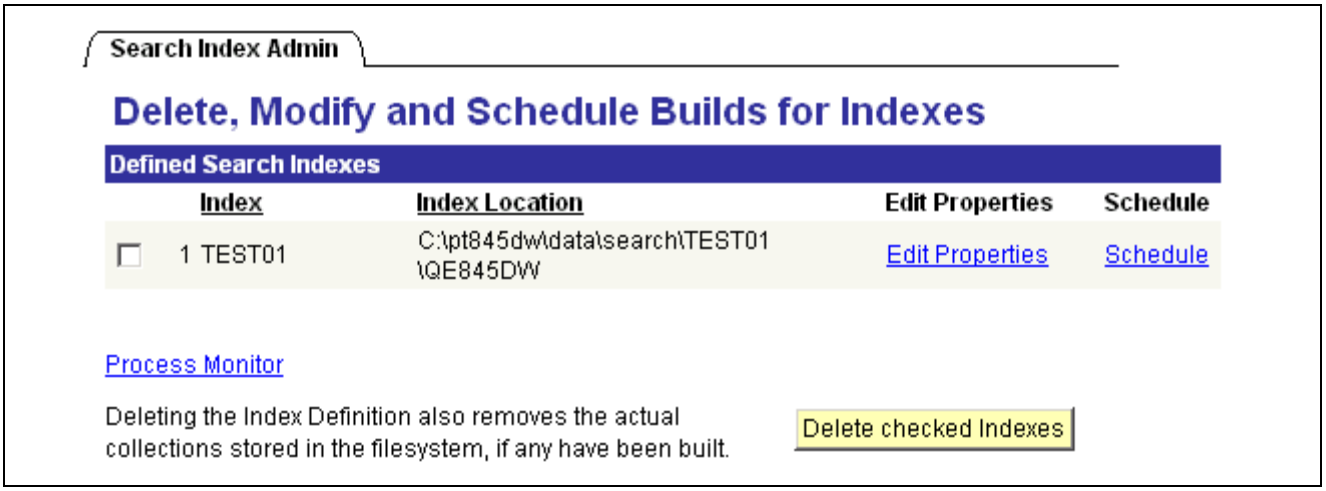
   Note. The previous procedure assumes that you've already used the Search Index Designer to define, build, and store the search indexes that you specify in the PSAPPSRV.CFG file.

   ---

5. Repeat the process with PSPRCS.CFG for PeopleSoft Process Scheduler.

Administering the Search Index

Select PeopleTools, Search Engine, Administration to access the Search Index Admin page.



Search Index Admin page

Index Displays the name of the index so that you can identify specific indexes. To select an index, select the check box to the left of the index name.

Index Location Displays the current location of the index.

Edit Properties Click to access the interface for changing the index location and to build indexes to support additional languages.

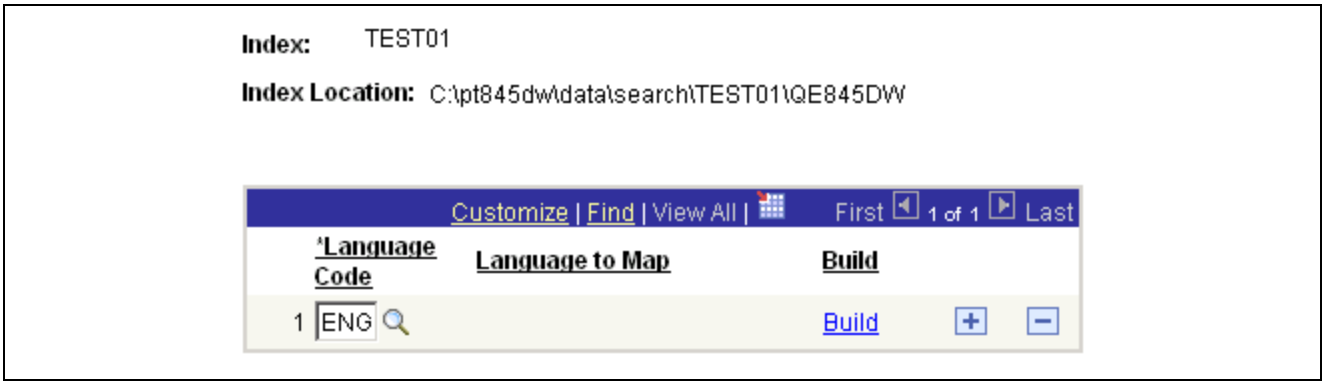
Schedule Click to access the interface for scheduling the program that maintains your collection.

Delete checked Indexes If you have selected indexes to be deleted, click this button to remove them from the system. The deletion process deletes the index definition and the collections that are stored in the file system.

Note. If you attempt to delete a scheduled index, you may see SQL errors on IBM DB2 UDB or Sybase database platforms.

Editing Properties

Select PeopleTools, Search Engine, Administration, Edit Properties.



Modifying index properties

Index Location	Displays the current location of the index.
Language Code	Select the language for which you want to build an index.
Language to Map	Currently disabled.
Build	After you add the additional indexes, click to create the indexes.

Note. Style files are located in the style subdirectory of the index. To make style changes, apply them to the files in this directory.

Scheduling Administration

Select PeopleTools, Search Engine, Administration, Schedule.



Scheduling builds

Add a new Recurrence Definition	In PeopleSoft Process Scheduler, you define run recurrence definitions that enable you to schedule jobs to run at regular intervals, such as monthly, weekly, daily, and so on. The more current you keep the collections, the more accurate your search results will be.
Type of Build	*Rebuild:* Select to drop the existing collection and rebuild a new collection. This applies to all types of collections. *Increment:* Use only for the spider indexes. For record-based indexes, only the *Rebuild* option is available.
Run Recurrence Name	Select the appropriate run recurrence definition for the collection maintenance requirements.
Server Name	Specify the PeopleSoft Process Scheduler server on which you want the build program to run. The PeopleSoft Process Scheduler system must be installed and configured before you can schedule the collection build program to run as a job.

Sharing Indexes Between Application Servers and PeopleSoft Process Scheduler

The index files reside on a file system at the home location and must be accessible to all application servers and process schedulers that will manipulate the index. An application server uses the index for searching while the process scheduler invokes an Application Engine program that builds the indexes. Therefore, if you are running a process scheduler on a different machine than the application server, ensure that the index files are accessible to both. You can do this three ways:

- Make a Microsoft Windows shared drive or NFS file system available for the index.

  Specify the index location in both the application server and the process scheduler to point to the shared directory.

- Run an instance of the process scheduler on the application server host and schedule only the building of indexes on this process scheduler.

  Because the process scheduler and the application servers are running on the same host, they create and read files from the same location.

- Use an external program such as FTP or Secure Copy (SCP) to copy all of the files and directories in the index home location from the process scheduler host (after the index has been built) to the application server host so that they are available for searching.

Modifying the VdkVgwKey Key

To make the VdkVgwKey more readable and easier to parse, use the following XML-like syntax:

```
<field fieldname='MYFIELD'/>
<row/>
<pairs/>
<sql stmt="SELECT 'Y' FROM PS_INSTALLATION"/>
```

- Fieldname and the SQL statement support single and double quotes, as well as no quotes at all (in which case only the first word is considered part of the option).

  Using double quotes for the SQL statement is recommended.

- The SQL statement must return only one column.

  Multiple rows are ignored. Trying to return more than one column results in a collection-build-time error.

- Currently, the only tag style that is supported is <tag/> with the slash (/) at the end.

- The VdkVgwKey can include any amount of literal text interspersed with the tags.

  This text is copied into the VdkVgwKey that goes into the BIF file, unmodified.

- Field names are automatically set in uppercase.

CHAPTER 9

Using PeopleSoft Configuration Manager

This chapter provides an overview of PeopleSoft Configuration Manager and discusses how to:

- Start PeopleSoft Configuration Manager.
- Specify startup settings.
- Specify display settings.
- Specify Crystal report and Business Interlink settings.
- Specify trace settings.
- Specify workflow settings.
- Specify remote call/AE settings.
- Configure developer workstations.
- Import and export environment settings.
- Configure user profiles.
- Specify command line options.
- Set up the PeopleTools development environment.

Note. PeopleSoft supports a number of versions of UNIX and Linux in addition to Microsoft Windows. Throughout this chapter, we make reference to operating system configuration requirements. Where necessary, this chapter refers to specific operating systems by name (eg Solaris, HP/UX, Linux, and so forth). However, for simplicity the word UNIX refers to all UNIX-like operating systems, including Linux.

Understanding PeopleSoft Configuration Manager

PeopleSoft Configuration Manager simplifies workstation administration by enabling you to adjust PeopleSoft registry settings from one central location. It contains a variety of controls that let you set up workstations for connecting to the database, using workflow, and so on. You can set up one workstation to reflect the environment at your site, and then export the configuration file, which can be shared among all the workstations at your site. You can also define separate profiles for connecting to different PeopleSoft databases.

PeopleSoft configuration parameters are grouped on the Configuration Manager pages according to the function, feature, or tool that they control. The label on the tab identifies the kind of parameters you can expect to find there.

Note. The changes you make within PeopleSoft Configuration Manager do not take effect until the next time you start PeopleSoft.

See Also

Chapter 9, "Using PeopleSoft Configuration Manager," Setting Up the PeopleTools Development Environment, page 209

Common Elements in PeopleSoft Configuration Manager

OK	Saves your settings and exits PeopleSoft Configuration Manager.
Cancel	Closes PeopleSoft Configuration Manager without saving any changes that you have made.
Apply	Saves your changes without exiting.

Starting PeopleSoft Configuration Manager

You can start PeopleSoft Configuration Manager by one of two methods:

- Double-click the Configuration Manager shortcut in your PeopleSoft program group.
- At a command prompt, enter:

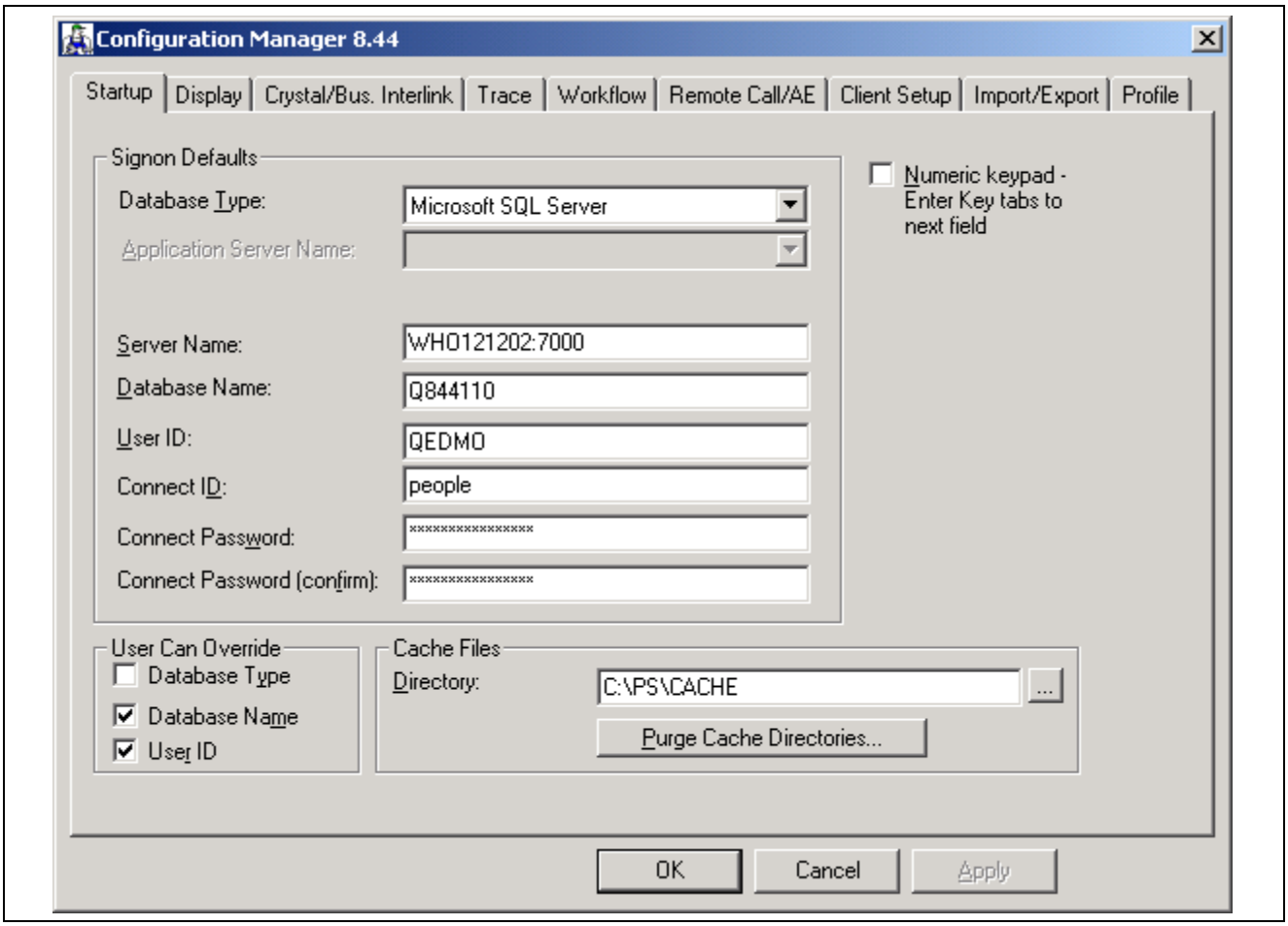
  PS_HOME\bin\client\winx86\pscfg.exe

Important! Certain PeopleSoft utilities require setting an environment variable, PS_SERVER_CFG, to run properly. However, PeopleSoft Configuration Manager isn't compatible with PS_SERVER_CFG being set.

Before you start Configuration Manager, you must ensure that PS_SERVER_CFG is not set. You can make this convenient by using a DOS batch file to unset PS_SERVER_CFG, launch Configuration Manager, then after Configuration Manager exits, reset PS_SERVER_CFG to its previous value.

Specifying Startup Settings

Select the Startup tab.

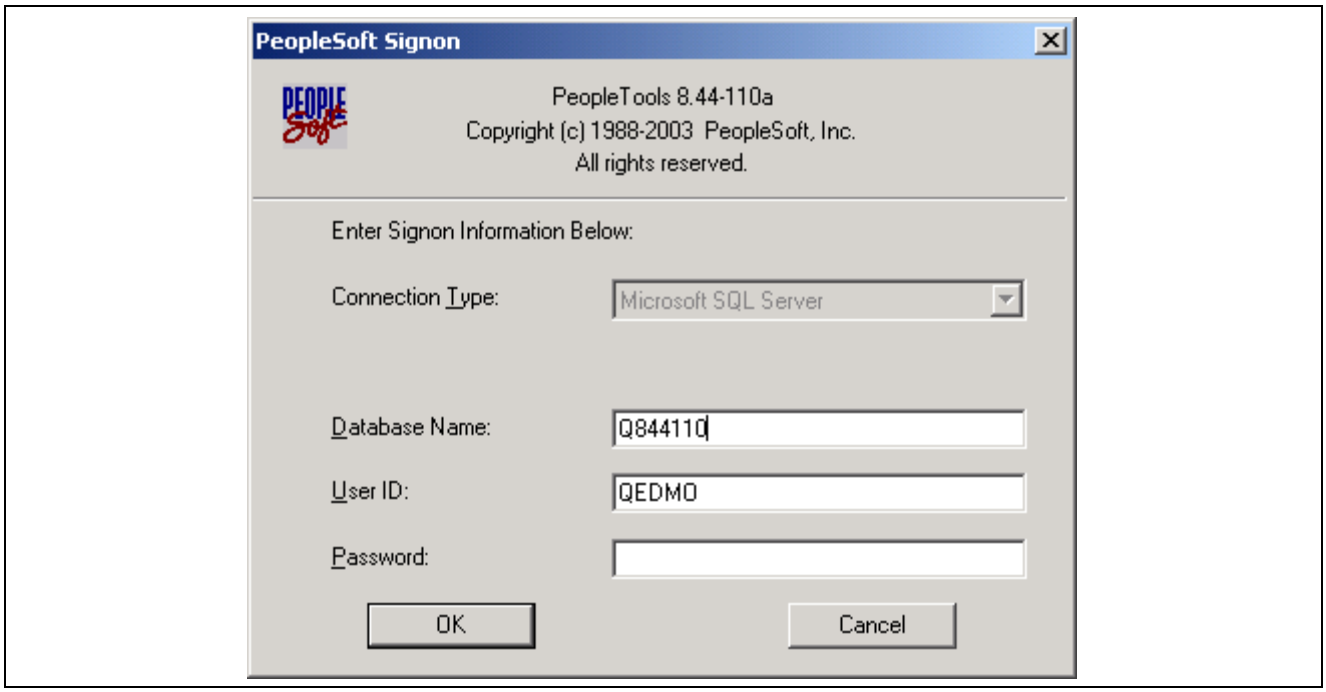


Startup tab

Use the Startup tab to customize the default values that appear on the signon screen. You do not need to change values unless you do not want to accept the defaults.

Signon Defaults

You use the Startup tab to customize signon screen default values (if you have appropriate permissions to do so). The following signon screen shows the values illustrated in the previous Startup tab example.



PeopleSoft Signon dialog box

Database Type Select the database type to appear as a default on the PeopleSoft Signon dialog box. Select *Application Server* to sign in to an application server instead of a database. To enable users to change their database type selection in the signon dialog box, you must select the Database Type option in the User Can Override group.

Note. When you select *Application Server* from the Database Type drop-down list, the Server Name and Database Namefields are disabled. The system obtains these values from the application server.

Application Server Name If you selected *Application Server* from the Database Type drop-down list, specify the application server's name in this field. You must have already configured your application server and registered it on the Profile tab.

Server Name Enter the name of the default database server. This parameter is only enabled for Informix, Sybase, and Microsoft SQL Server, and refers to the instance to which the user connects.

For Informix, enter the server name in lowercase.

Database Name Enter a default database name. You can choose any valid PeopleSoft database name. As with the database type, you must select the appropriate option in the User Can Override group to enable users to override the default database name selection when they sign in.

User ID Specify the default user ID to sign in to PeopleSoft.

You can use the user ID in conjunction with a PSUSER module containing a user-defined sign-in process. The PSUSER code, if present, can evaluate and modify the user ID value before you attempt to sign in to the selected database.

Connect ID and **Connect Password** — PeopleSoft uses the connect ID for the initial connection to the database. Use the Connect Password field to define a default connect ID password.

Note. The connect ID edit box must contain a value, or the user can't sign in to the system in a two-tier environment.

See *Enterprise PeopleTools 8.46 PeopleBook: Security Administration*, "Understanding PeopleSoft Security".

See *PeopleTools 8.46 Installation Guide for Your Database Platform.*

Numeric Keypad - Enter Key Tabs to Next Field

In Microsoft Windows applications, pressing the ENTER key in a dialog box selects the default action button. For example, in the PeopleSoft Signon dialog box, pressing ENTER is the same as clicking the OK button. Selecting the Numeric keypad check box overrides this default behavior for the ENTER key on the numeric keypad; instead of selecting the action button, pressing the ENTER key moves the cursor to the next field in the dialog box.

Note. This check box affects the ENTER key on the numeric keypad, but not the ENTER key on the main keyboard.

User Can Override

Some PeopleSoft sites use multiple database types and names. Using the check boxes in the User Can Override group box, you can enable users to enter a database type, database name, or user ID other than the default provided at sign-in. In most cases, you use these controls to prevent users from attempting to sign in onto any database other than the default.

Database Type — When selected, users can choose a database other than the default. Selecting this check box selects the Database Name and User ID options automatically. You cannot clear Database Name or User ID without first clearing Database Type. When configuring a workstation to connect in both two-tier and three-tier, you must select this box. The user needs to specify a two-tier or three-tier connection from the PeopleSoft Signon dialog box.

Database Name — When selected, the User ID check box is automatically selected, although you can clear it. To clear Database Name, you must clear the Database Type check box.

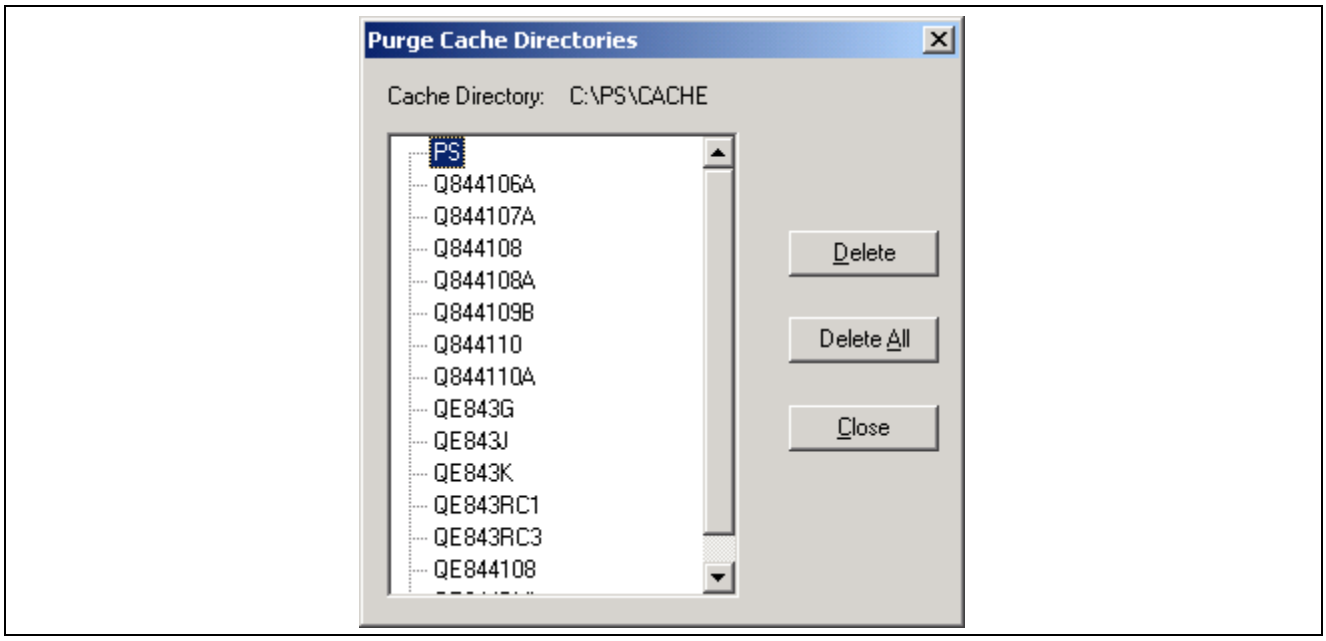
User ID — Select to enable users to users override only the user ID submitted at when they sign in. You cannot clear User ID if Database Type is selected.

Cache Files

Enter the parent directory that holds your cache file directories. For example, enter *C:\PS\CACHE.*

Note. Cache files store database object information locally and are automatically downloaded the first time you open a PeopleSoft database object. They are also downloaded automatically if the master copy of the object on the database server is changed. One cache file directory stores the cache files for each PeopleSoft database that you use.

Clicking Purge Cache Directories brings up a dialog box with your existing cache file directories, as shown in the following example:

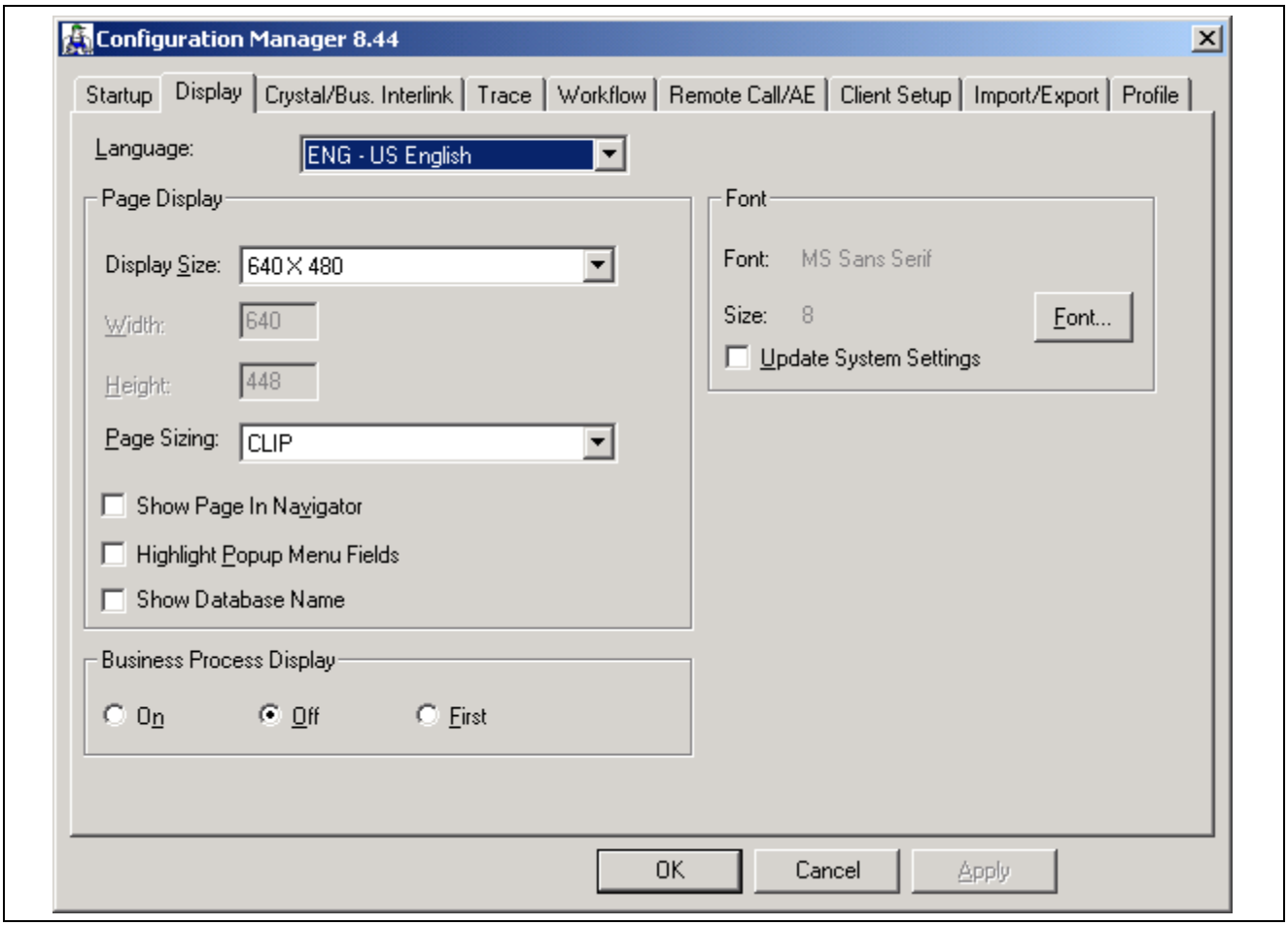


Purge Cache Directories dialog box

You can select a single directory and click Delete, or you can click Delete All to remove all directories. If a cache file directory is missing (after you delete it), the system automatically rebuilds it the next time that cache files are downloaded. After you delete the appropriate cache directory, click Close to return to the Startup tab.

Specifying Display Settings

Select the Display tab.



Display tab

Use the Display tab to configure the appearance of the PeopleSoft graphical user interface. For instance, you can adjust page width and height to fit in with the other elements on your desktop.

Language

In the Language drop-down list box, specify which language you want to display on your PeopleSoft pages. The default setting is US English.

Note. You select from the languages that PeopleSoft delivers. Although you can implement applications to appear in other languages, you cannot switch to custom languages using PeopleSoft Configuration Manager. Switch to these languages by manually changing the registry setting.

Page Display

You can adjust page display size or the page height and width.

Display Size, Width, and **Height**

Specify display size in pixels. This setting affects the default size of the PeopleSoft window as displayed in the corresponding Width and Height fields. Select from:

- *640 X 480:* The default window size is 640 pixels by 448 pixels.
- *800 X 600:* The default window size is 800 pixels by 576 pixels.
- *1024 X 768:* The default window size is 1024 pixels by 744 pixels.

- *Custom:* You can manually set the default window size by specifying width and height values.

Note. Changing these parameters does not affect open windows. If either value is either blank or zero, the values are reset to 640 by 480 pixel resolution.

Page Sizing Use this field to specify how pages that were designed for a different-size window should be displayed. Select from:

- *CLIP:*Page controls are always displayed in their normal size. If a page is too large for the window, the page information is clipped along the right and bottom edges of the window. Use scroll bars to view the remainder of the page.
- *SCALE:* Pages are scaled to fit the window as necessary. For example, if your display size is set to 640 by 480 pixels, and you open a page designed to display in an 800 by 600 pixel window, the page controls are scaled down so that all page information appears. Conversely, if you open a page designed for 640 by 480 pixel resolution in a larger window, the page controls are scaled to fill the window completely.

Show Page in Navigator Select to see the navigator tree view and the page view at the same time.

Highlight Popup Menu Fields Select to highlight fields with associated pop-up menus. The box is clear by default. In most cases, it's a good idea to indicate which fields contain pop-up menus. Pop-up menus are indicated by a black rectangle surrounding the perimeter of a page control.

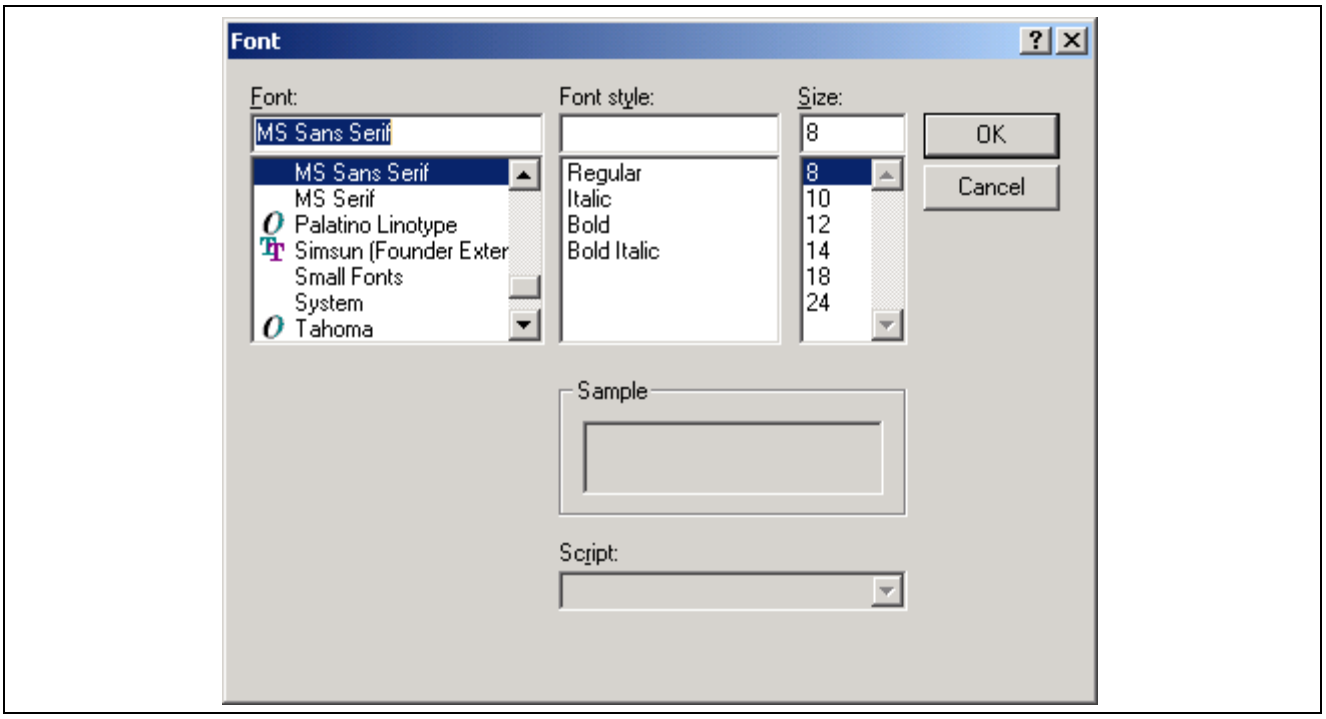
Show Database Name Select to display the name of current database in the status bar at the bottom of a PeopleSoft page, in addition to the current page name and the activity. For example, the status bar might read PTDMO, Job Data 1, Add. This feature is useful if you are running multiple instances of PeopleTools.

Note. The database name may be abbreviated to fit on the screen.

Font

Use the Font options to configure the way that text appears on the screen in PeopleSoft applications.

Click the Font button to bring up a standard font selection pop-up menu, as shown in the following example:



Font dialog box

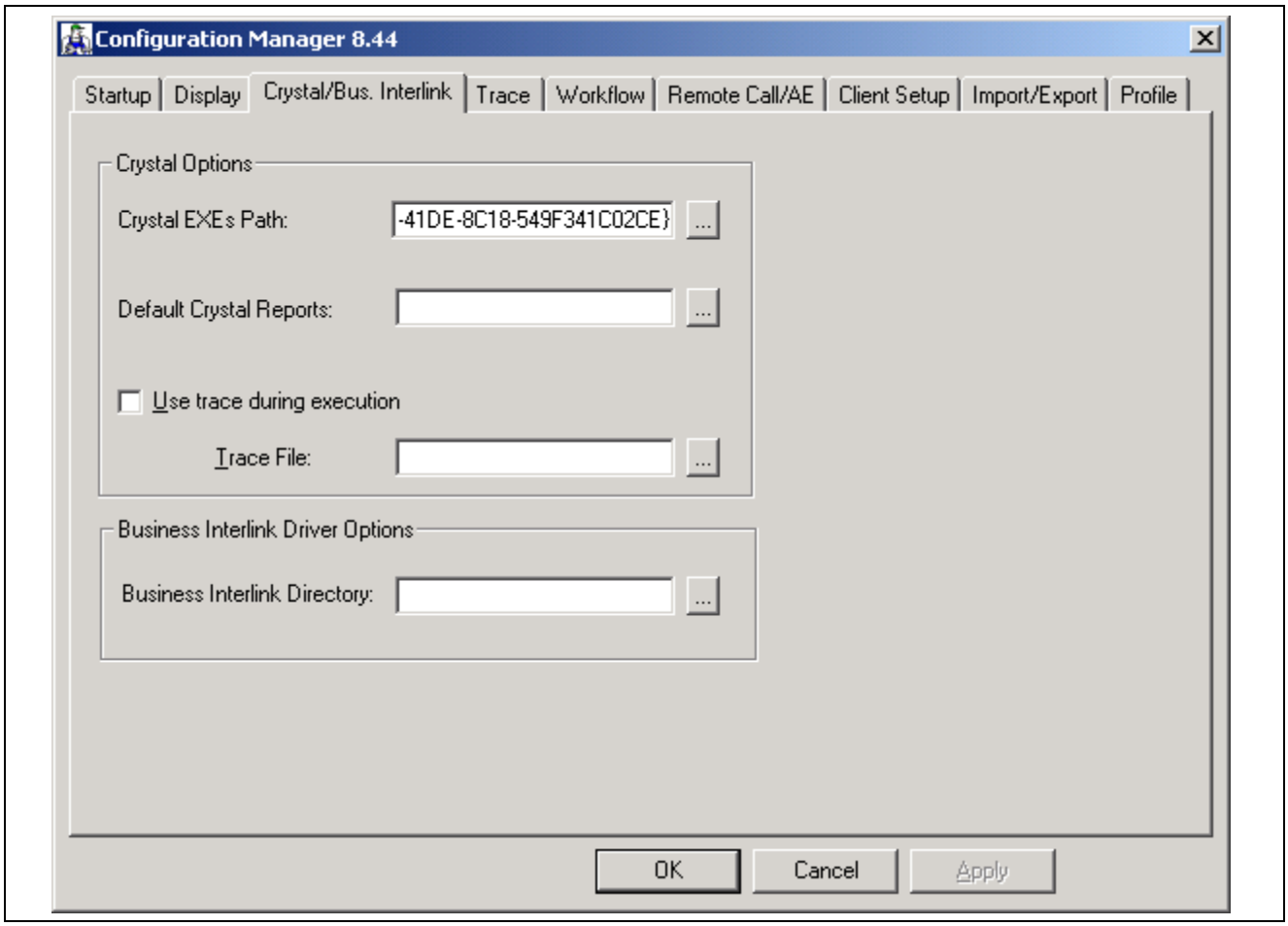
Business Process Display

Select from:

- *On:* The navigator appears with each menu group that you open.
- *Off:* You must open the navigator manually.
- *First:* The navigator appears on the first instance of PeopleSoft only. Subsequent instances do not display the navigator.

Specifying Crystal Report and Business Interlink Settings

Select the Crystal/Bus.Interlink (Crystal/Business Interlink) tab.



Crystal/Bus. Interlink tab

Use the Crystal/Bus.Interlink tab to specify settings for Crystal Reports in the PeopleSoft environment and for PeopleSoft Business Interlinks.

Crystal Options

If you have Crystal Reports installed on a workstation, the Crystal executables path is populated automatically. If Crystal Reports is installed on a network drive, use this field to reflect the location of the Crystal Reports executables. For example, you might enter *n:\hr840\bin\client\winx86\crystal.*

Use the Default Crystal Reports field to specify the default location of reports. If this setting does not apply to your site's Crystal Reports implementation, leave this field blank.

When you select Use Trace during execution, Crystal Reports writes the trace statements to a log file that you specify in the Trace File field.

Business Interlink Driver Options

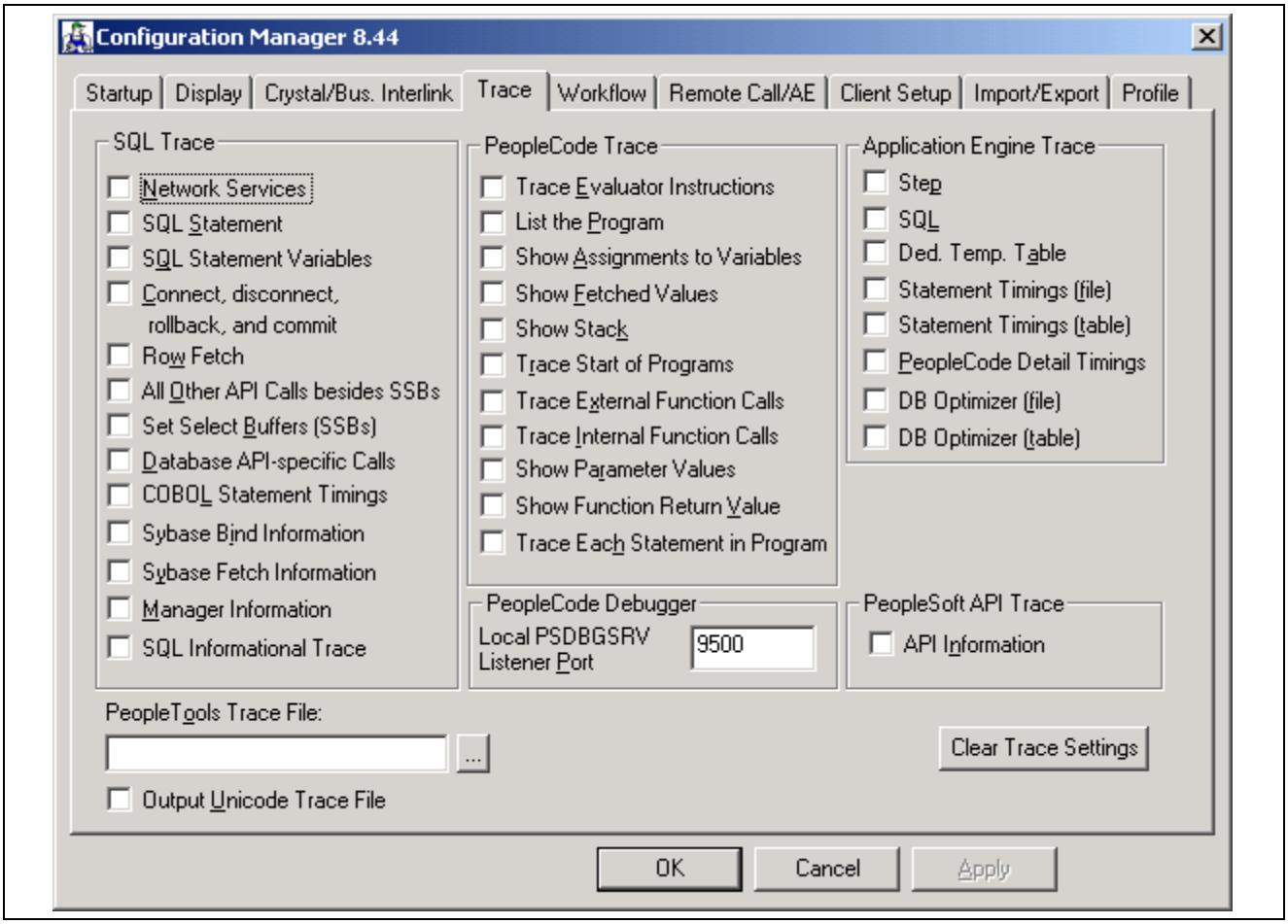
In the Business Interlink Directory box, enter the complete path to the directory that contains the drivers that PeopleSoft Business Interlinks uses to communicate with external systems.

See Also

Enterprise PeopleTools 8.46 PeopleBook: Crystal Reports for PeopleSoft

Specifying Trace Settings

Select the Trace tab.



Trace tab

Use the Trace tab to select tracing options for various parts of the PeopleTools system, such as SQL statements, PeopleCode, and PeopleSoft Application Engine. If you work on tuning your PeopleSoft system and improving online performance, familiarize yourself with this tab. When you update the Trace tab, the new settings take effect the next time you launch PeopleTools.

Note. The Trace tab in PeopleSoft Configuration Manager traces only Microsoft Windows client (two-tier) interactions. Use these settings only when you require tracing on the client.

You can override some of the trace options on this tab from the Trace SQL and Trace PeopleCode pages in PIA.

See Chapter 11, “Configuring Trace and Debug Settings,” Configuring SQL Trace, page 261 and Chapter 11, “Configuring Trace and Debug Settings,” Configuring PeopleCode Trace, page 260.

SQL Informational Trace

Select this check box to trace information messages from the Runstats command in DB2 UDB for z/OS executed as a result of an %UpdateStats meta-SQL command.

PeopleTools Trace File

The default filename for the PeopleTools trace file is DBG1.TMP. The system writes the file to the following directories:

- In Microsoft Windows: %TEMP% directory.
- In UNIX: $PS_HOME/log/dbname.

Important! The PeopleTools trace file stores elapsed times for PeopleCode and SQL events to a precision of one microsecond (six decimal places). However, due to limitations of the operating system, Windows precision is actually in milliseconds (three decimal places), so the last three digits in a Windows trace will always be zero. Elapsed times in UNIX are accurate to one microsecond.

To specify a different PeopleTools trace file:

1. Click the button on the right side of the PeopleTools Trace File edit box.

   A standard Open dialog box appears.
2. Navigate to and select the new trace file.
3. Click Open.

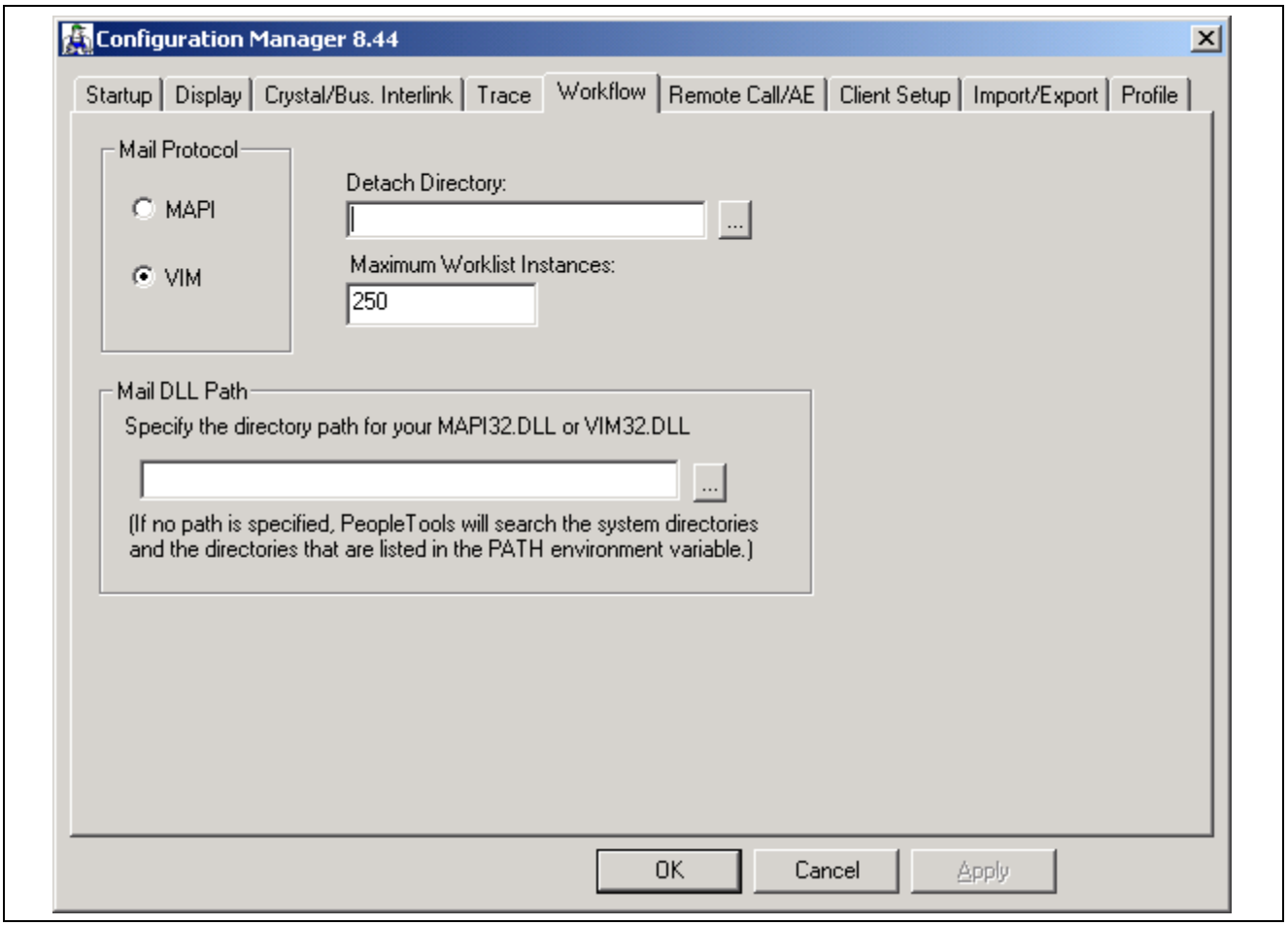
   The PeopleTools Trace File field displays the path and filename.

See Also

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Application Engine, "Tracing Application Engine Programs," Setting Options in PeopleSoft Configuration Manager

Specifying Workflow Settings

Select the Workflow tab.



Workflow tab

Use the Workflow tab to specify the options and locations related to the PeopleSoft Workflow implementation at your site.

MAPI (mail application programming interface) and **VIM** (vendor independent messaging interface)

To incorporate email into your workflow scheme, specify the mail protocol your site uses. PeopleSoft supports the following protocols:

- Select MAPI to configure your client to perform PeopleSoft Workflow email generation over the Microsoft Mail messaging system.
- Select VIM to configure your client to perform PeopleSoft Workflow email generation over a VIM-compliant messaging system, such as cc:Mail.

Mail DLL Path (mail dynamic link library path)

Enter the location of the mail DLL, such as *C:\Windows\System*.

Detach Directory

Specify the directory into which PSNOTES.EXE detaches file attachments on the forms it receives. This is also where PSNOTES.EXE places any files that it does not deliver to the message agent.

Maximum Worklist Instances

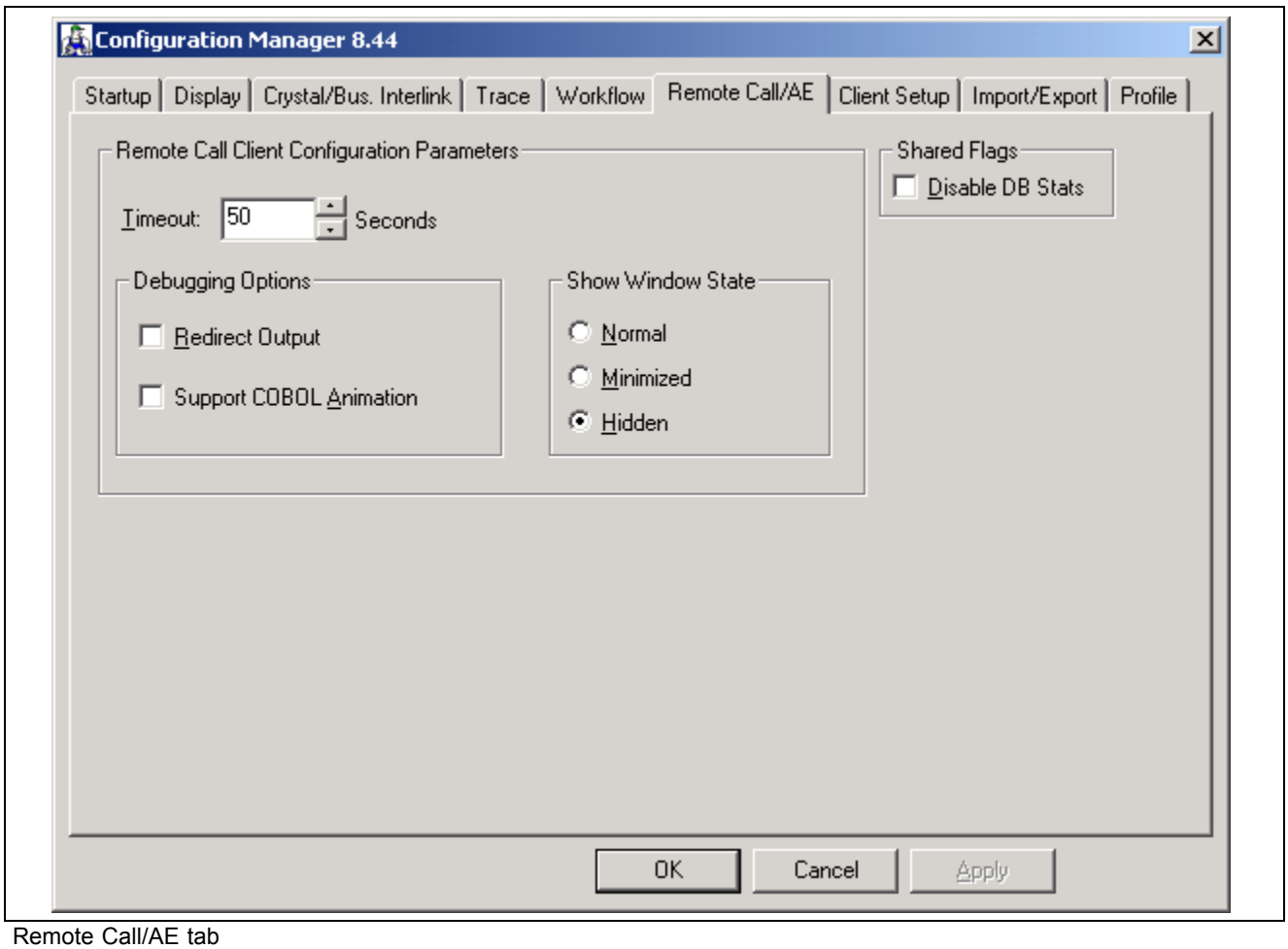
Enter a number to limit the number of worklist instances or entries that appear when viewing worklists. The default value is *250*. If you do not want any rows returned, leave the field blank.

See Also

Enterprise PeopleTools 8.46 PeopleBook: Workflow Technology, "Defining Worklist Records"

Specifying Remote Call/AE Settings

Select the Remote Cal/AE tab.



Remote Call/AE tab

Some PeopleSoft applications use the BEA TuxedoRemote Call feature, which invokes data-intensive transactions on a remote server. This helps to alleviate heavy processing on the client.

Timeout Enter the amount of time after which Remote Call terminates the child COBOL process. The default is *50* seconds.

Redirect Output Select to specify whether the standard out or standard error of the child COBOL process is directed to a file. This check box is clear by default.

Support COBOL Animation Select to save the COBOL input file so that you can reuse it with COBOL animator. This check box is clear by default.

Normal, Minimized, and **Hidden**

Specify how the window state of the child COBOL process appears on the desktop.

- Select Normal to have the window state appear like a DOS window on the desktop.
- Select Minimized to have the window state appear as an icon on the task bar.
- Select Hidden to have the window state run unseen in the background.

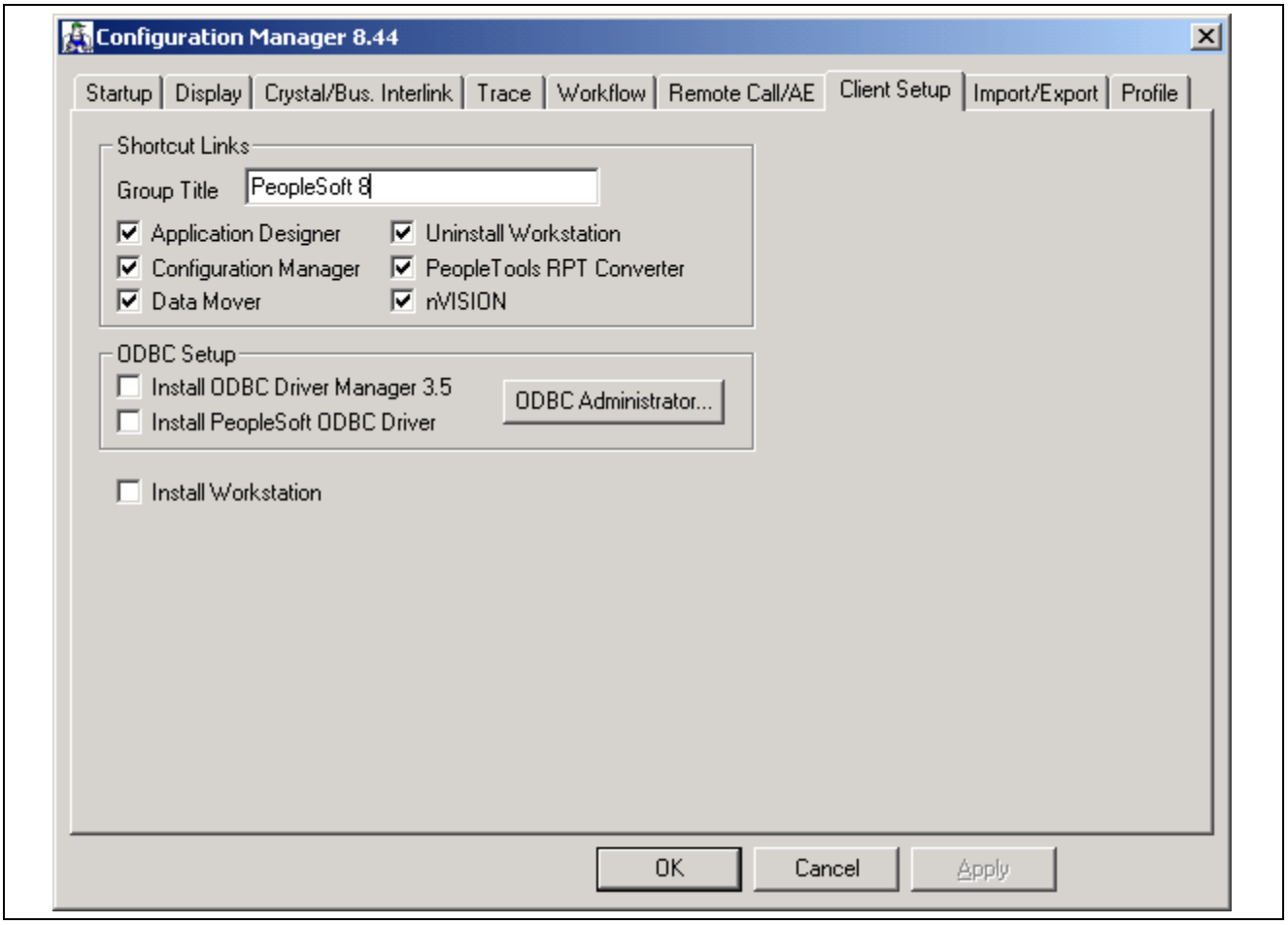
Disable DB Stats (disable database statistics)

Select to turn off the %UpdateStats meta SQL construct. This setting applies to Application Engine programs.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Application Engine*, "Using Meta-SQL and PeopleCode".

Configuring Developer Workstations

Select the Client Setup tab.



Client Setup tab

As part of the PeopleSoft installation process, you need to configure developer workstations (also called the PeopleTools development environment) to run successfully with your PeopleSoft system. Use this tab to configure developer workstations and invoke the Client Setup process. For example, you can select which shortcuts appear on a developer workstation desktop. (Although this tab is specifically for developer settings, all of the PeopleSoft Configuration Manager settings may affect developers, especially the Startup tab and the Process Scheduler tab for the default profile.)

Shortcut Links

Here are the various shortcut links:

Application Designer Adds a shortcut for the main PeopleTools development environment.

Configuration Manager Adds a shortcut for PeopleSoft Configuration Manager, which enables you to edit registry settings relevant to PeopleSoft.

Data Mover Adds a shortcut to launch PeopleSoft Data Mover.

Uninstall Workstation Adds a shortcut for Uninstall Workstation, which uninstalls the most recent client setup.

PeopleTools RPT Converter Adds a shortcut to a standalone program that converts RPT files from the format PeopleSoft used in previous releases to the PeopleTools 8 format. You only need to run this program if you are upgrading from previous versions of PeopleTools.

nVISION Adds a menu item for PS/nVision to the PeopleSoft 8 menu group in the Microsoft Windows Start menu.

Note. Back up RPT files before you run the converter program, which significantly alters them.

ODBC Setup

You need to specify one or both of the Open Database Connectivity (ODBC) setup options to run PeopleSoft Open Query.

Select Install ODBC Driver Manager 3.5 to install the Microsoft ODBC drivers that you need to run in conjunction with the PeopleSoft ODBC driver to enable PeopleSoft Open Query. If you already have the Microsoft ODBC drivers installed on your client, this is optional.

Select Install PeopleSoft ODBC Driver to enable PeopleSoft Open Query.

Note. The Client Setup process installs the ODBC Driver Manager version 3.510.3711. Any preceding versions of the ODBC driver are overwritten, and any versions higher than 3.510.3711 are not overwritten.

Install Workstation

Select the Install Workstationcheck box to run the Client Setup process. Only select the check box after specifying all the appropriate selections on all PeopleSoft Configuration Manager tabs. If you do not select this box, the Client Setup process will not run.

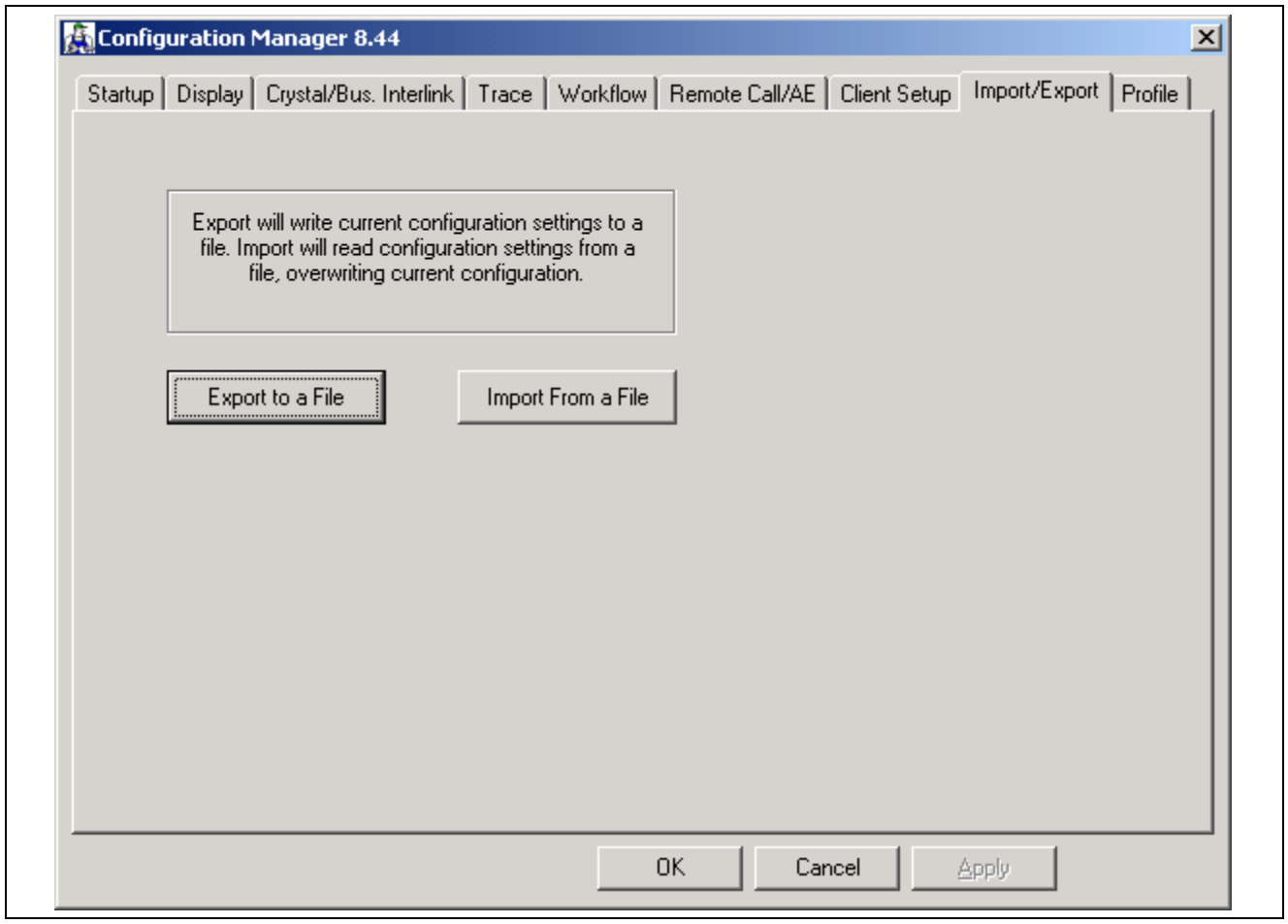
After you select this check box, click either OK or Apply.

See Also

Chapter 9, "Using PeopleSoft Configuration Manager," Setting Up the PeopleTools Development Environment, page 209

Importing and Exporting Environment Settings

Select the Import/Export tab.



Import/Export tab

Use this tab to export, or save to file, the specified environment settings, and to import previously exported settings. This feature is useful when you plan to configure multiple workstations with similar settings.

Export to a File Click to write current configuration settings to a file. A Save dialog box appears. Note the file name that you give the configuration file.

Note. Click Apply before you export a file. This ensures that the exported configuration file reflects the current settings.

Import from a File Click to import previously saved configurations on another workstation. Importing a configuration file overrides all the current environment settings on the machine that you import to.

When you click this button, an Open dialog box appears. Navigate to the directory containing the appropriate configuration file, select the file, and click Open.

Warning! In addition to overwriting environment settings, this function also overwrites all existing settings made in PeopleSoft Application Designer.

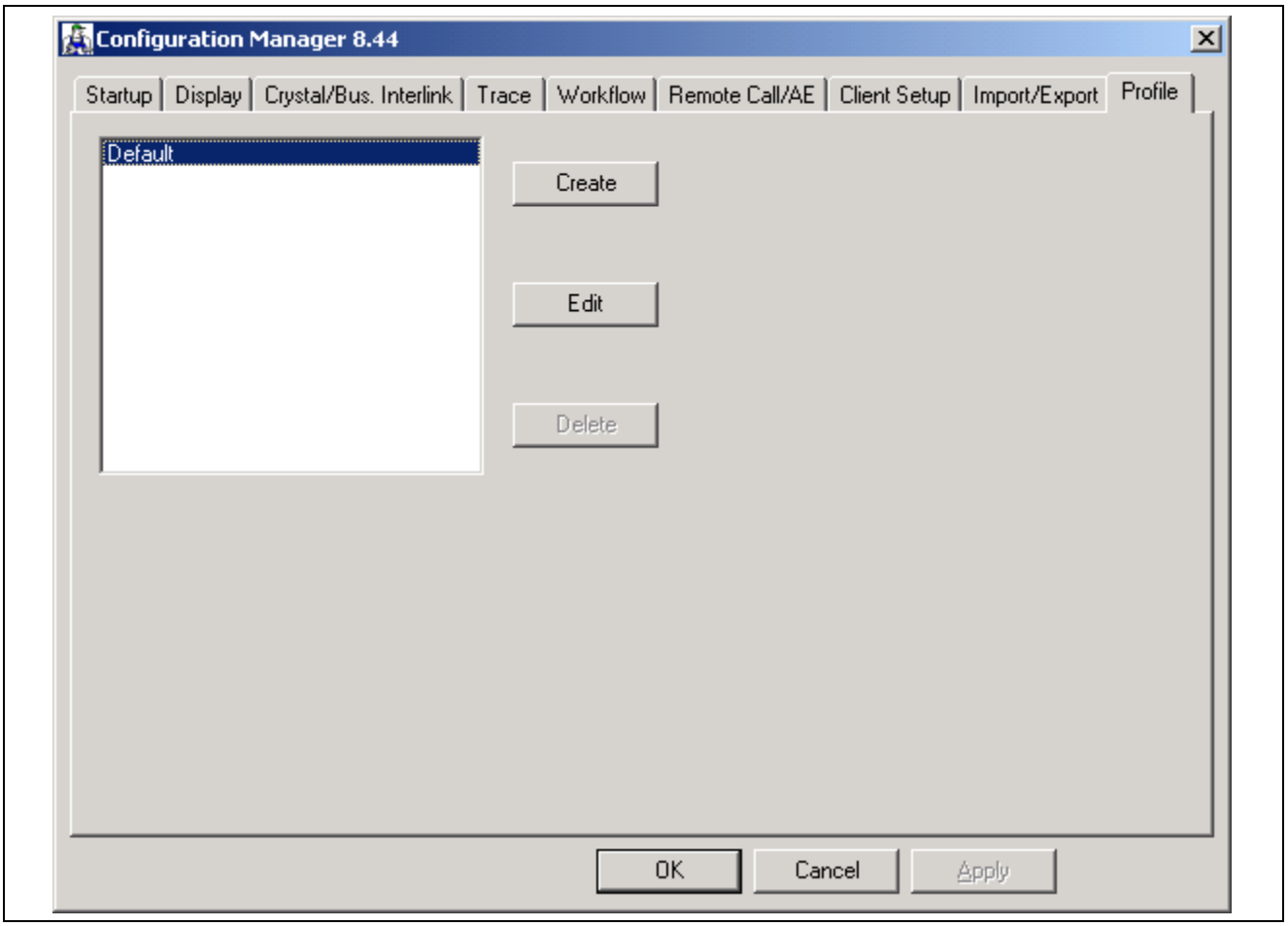
Configuring User Profiles

This section discusses how to:

- Define a profile.
- Specify databases and application servers.
- Configure process scheduler.
- Configure nVision.
- Specify common settings.

Defining a Profile

Select the Profile tab.



Profile tab

Use this tab to define one or more user profiles, each of which specifies connection parameters and file location information for a PeopleSoft installation.

Many PeopleSoft installations include multiple databases. For example, there may be one database for tracking financial information, such as expense reports, and another database for human resources processes, such as benefits enrollment. Each of these databases has its own set of supporting files, SQR reports, COBOL processes, and so on. PeopleTools locates these files by referring to the Microsoft Windows registry. By defining multiple profiles, you can tell PeopleTools to use different directory paths depending on the database.

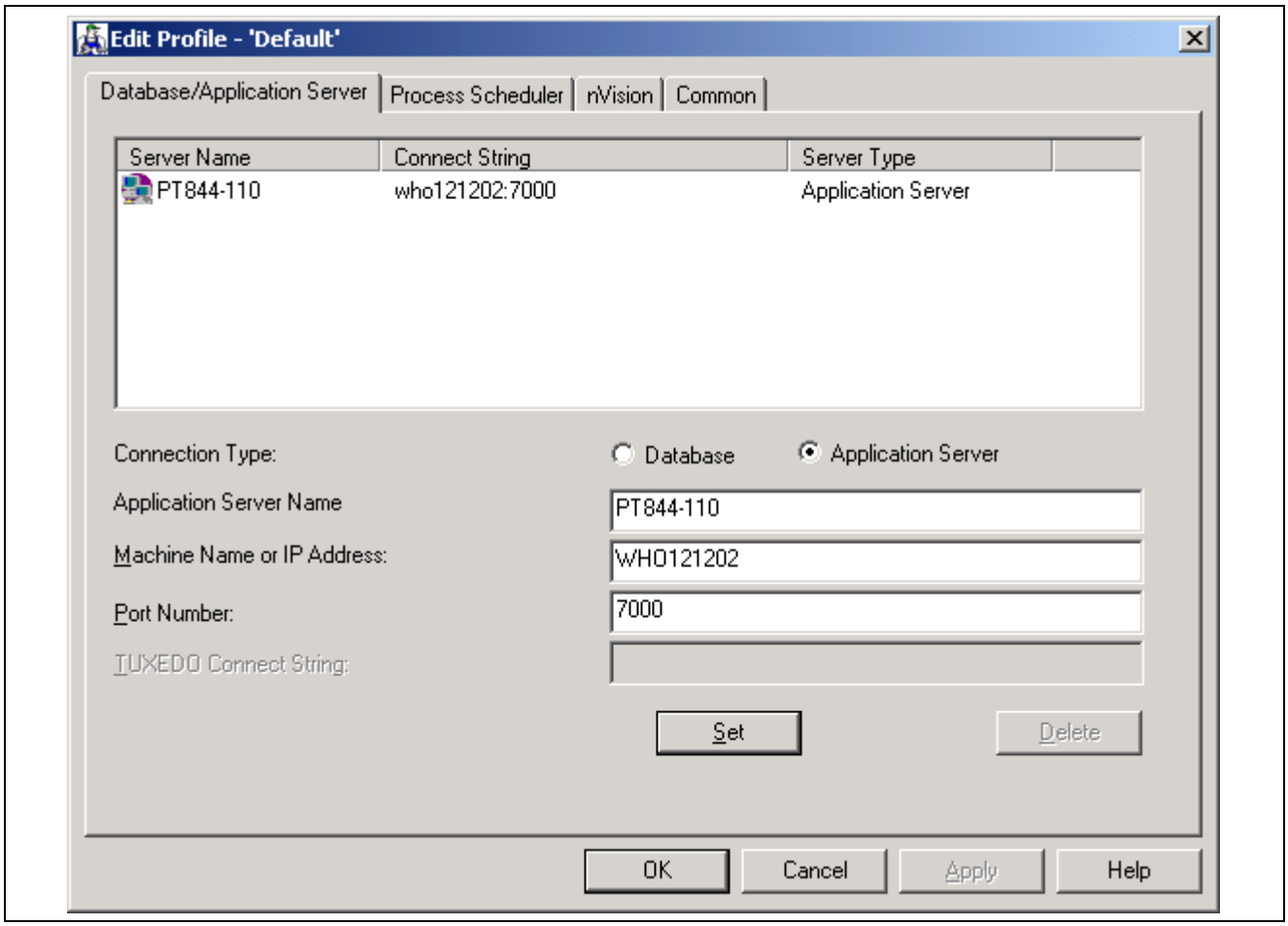
When you first open PeopleSoft Configuration Manager, the Profile tab displays a single profile named Default. To set the parameters for this profile, make sure that it's selected, and click the Edit button. The Edit Profile dialog box appears.

Each workstation must have a default profile, which is used when the user signs in to a database or application server that isn't listed in any profile. If the workstation requires only one set of profile settings, you can use the default profile. You can also set up multiple PeopleSoft Configuration Manager profiles. The profiles are set for Microsoft Windows workstations and are shared by all workstation users.

Note. You can use profiles to easily switch between applications.

Specifying Databases and Application Servers

From the Edit Profile dialog box, select the Database/Application Servers tab.



Edit Profile - Database/Application Server tab

Use this tab to specify the configured databases and application servers associated with this profile. When a user enters one of these databases or application servers in the PeopleSoft Signon dialog box, PeopleTools uses the registry settings associated with this profile.

Note. You can assign multiple databases and application servers to a single profile. However, each database and application server must be assigned to only one profile. If you try to add a database to a second profile, PeopleSoft Configuration Manager asks you if you want to remove it from the previous profile and add it to the current one.

Note. Before you enter a database or application server on this tab, you should have already installed and configured it as documented in the PeopleSoft installation documentation for your database platform.

Application Server Name

Enter a name for an application server that you have configured. This name will appear in the drop-down list box on the PeopleSoft Signon dialog box. Choose a name that's intuitive for your site.

Note. Application server names cannot exceed 24 characters.

Machine Name or IP Address

Enter the IP address or the resolvable server name of the application server you specified in the Application Server Name field. You specified the IP address in the [Workstation Listener] section of your PSAPPSRV.CFG file when you installed your PeopleSoft application server. For example, you could enter *207.135.65.20* or *sp-hp32*.

Port Number

Enter the port number for the application server that you specified in the Application Server Name field. You specified the port number when you installed and configured the application server using PSADMIN. A port number is an arbitrary number between 0 and 9999 that is determined by site specifications.

TUXEDO Connect String

Use this field to support dynamic load balancing. You can specify a free-form connect string that allows a client to connect to another application server in case another is either down or being used to full capacity.

Note. The BEA Tuxedo connect string cannot exceed 1000 characters.

When configuring load balancing, you might choose from the following approaches:

- Round robin load balancing.

  With this approach, you specify multiple application servers, and each client picks a server randomly. This approach assumes that application server will receive an equal number of connections. To specify round robin load balancing, use the following syntax for the connect string:

  (//*IP address 1*:*port 1*|//*IP address 2*:*port 2*|//*Ip address n*:*port n*)

  You can specify the IP address using either dotted notation or by using the server's DNS name. Either way, the slashes (//) preceding the IP address are required.

  If the selected application server is unavailable, your connection attempt fails, and the system does not try to connect you to the other application servers defined within the parentheses.

  Spaces are not allowed in any part of the connection string. The system automatically removes embedded spaces before storing the value in the registry.

- Round robin with failover.

  With this approach, you define a failover connection string. Use the following syntax:

  (//*IP address 1*:*port 1*|//*IP address 2*:*port 2*),(//*IP address 3*: *port 3*)

  If the application server selected from the first group of parentheses (IP addresses 1 and 2) is unavailable, the system automatically attempts to connect to an application server defined in the second group (IP address 3). If that application server fails, the system attempts to connect to the next group to the right, sequentially.

  If multiple application servers are defined within any group, the system round-robins between them. If the selected application server fails, the system attempts to connect to the next application server to the right, if any. The following list shows three examples of connect strings that use this approach:

  - (//sp-ibm01:8000|//sp-ibm02:8000),(//sp-nt01:8000)
  - (//208.136.78.88:8000|//208.136.78.88:8050|//208.136.78.88:8080)
  - (//sp-sun01:8000),(//sp-sun02:8000),(//sp-sun03:8000)

Set and Delete Buttons

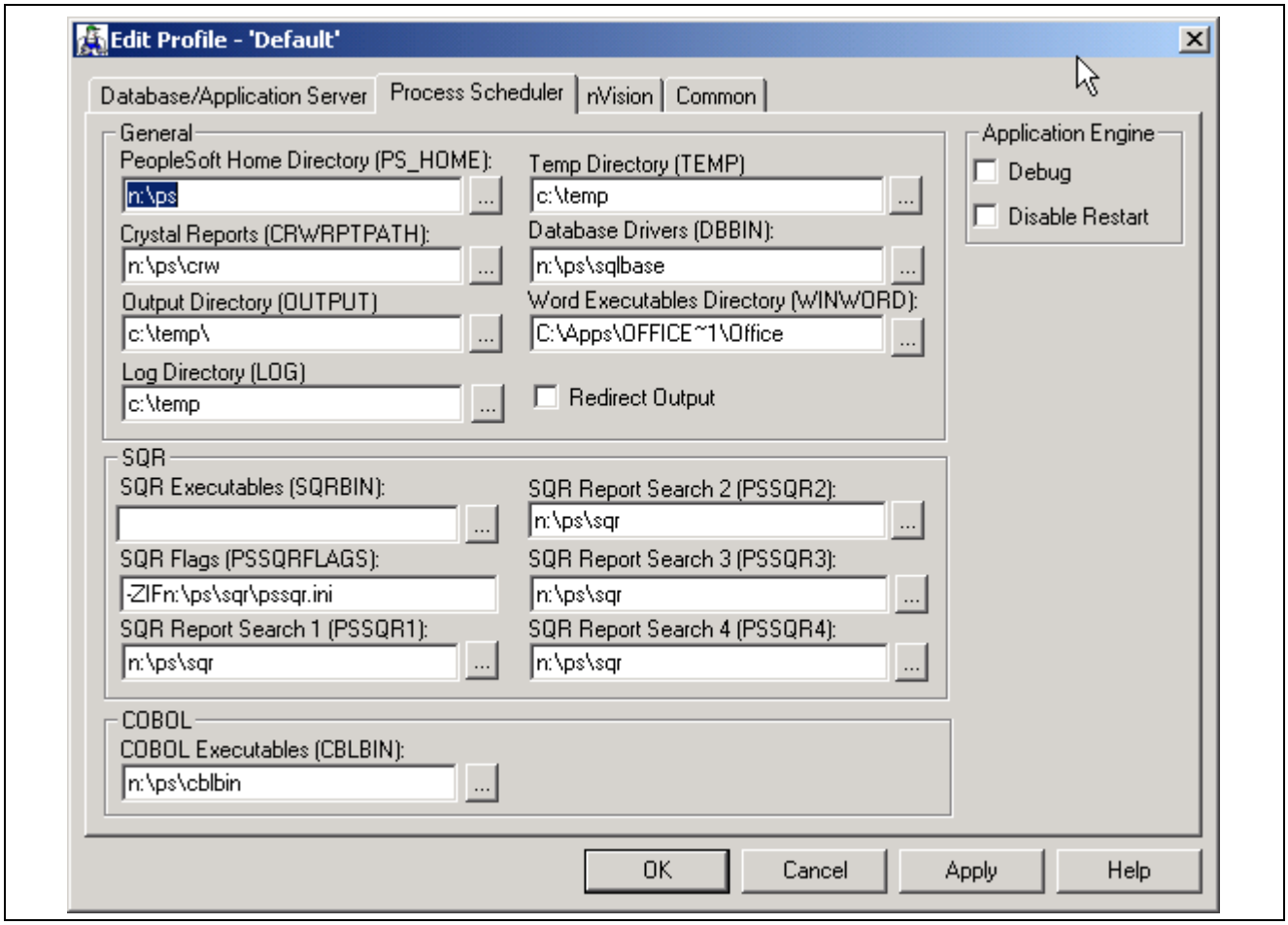
When you click Set, your application server information is displayed in the grid at the top of the dialog box. You can enter a new application server name and set up a different server if you like.

Note. The settings in the grid are not saved until you click Apply or OK. If you click Cancel without first clicking Apply or OK, you lose all the information in the grid.

To remove an application server configuration, select its application server name in the grid and click Delete.

Configuring Process Scheduler

Access the Process Scheduler tab.



Edit Profile - Process Scheduler tab

Use this tab to specify the directories that are associated with PeopleSoft Process Scheduler jobs, such as SQR and COBOL directories.

General

PeopleSoft Home Directory Enter your high-level PeopleSoft directory, such as *N:\HR840*.

Crystal Reports Enter the file path to \CRWRTPATH, where Crystal Reports sends your reports.

Output Directory (Optional) Enter the directory used with the Output Destination field when scheduling a PeopleSoft Process Scheduler request.

Log Directory Enter the directory for SQR, COBOL, and PeopleSoft Process Scheduler log files.

Temp Directory Enter he path to your temporary directory, for example, *C:\TEMP*. This directory stores log files and other output files.

Database Drivers Enter the path to the directory where your database drivers reside.

Word Executables Directory Enter the directory containing Microsoft Word executables; for example, *N:\Apps\Office2000\Office*.

Redirect Output Select to redirect onscreen COBOL Display statements to a log file. (If this check box is clear, you see the onscreen messages only.) Sending the messages to a log file is useful for debugging purposes. The log file is created in the %TEMP%\PS_HOME\DBNAME directory. In addition to the output generated by COBOL Display statements, the log file contains errors generated by the COBOL runtime system.

Note. To use the PeopleSoft Application Engine debug feature, clear Redirect Output.

Application Engine

Debug Select to enable the PeopleSoft Application Engine command-line debugger.

Warning! Select the Debug check box only when you are testing and troubleshooting client-side processes. If you select Debug and submit a process request to the server, the process hangs, waiting for a user command.

Disable Restart Select to disable the PeopleSoft Application Engine restart feature, which lets you restart an abnormally terminated Application Engine program. When selected, PeopleSoft Application Engine programs start from the beginning. This option is useful during debugging. Do not select it in a production environment.

SQR

SQR Executables Enter the path to the directory where SQR executables reside.

SQR Flags Enter the SQR parameters that PeopleSoft Process Scheduler should pass on the command line to the SQR executables. The following SQR flags are required for launching SQR reports:

- -i: Specifies the path to SQC files.
- -m: Specifies the path to the ALLMAXES.MAX file.
- -f: Specifies the output path.
- -o: Directs log messages to the specified file.
- -ZIF: Sets full path to the and name of the SQR initialization file, SQR.INI.

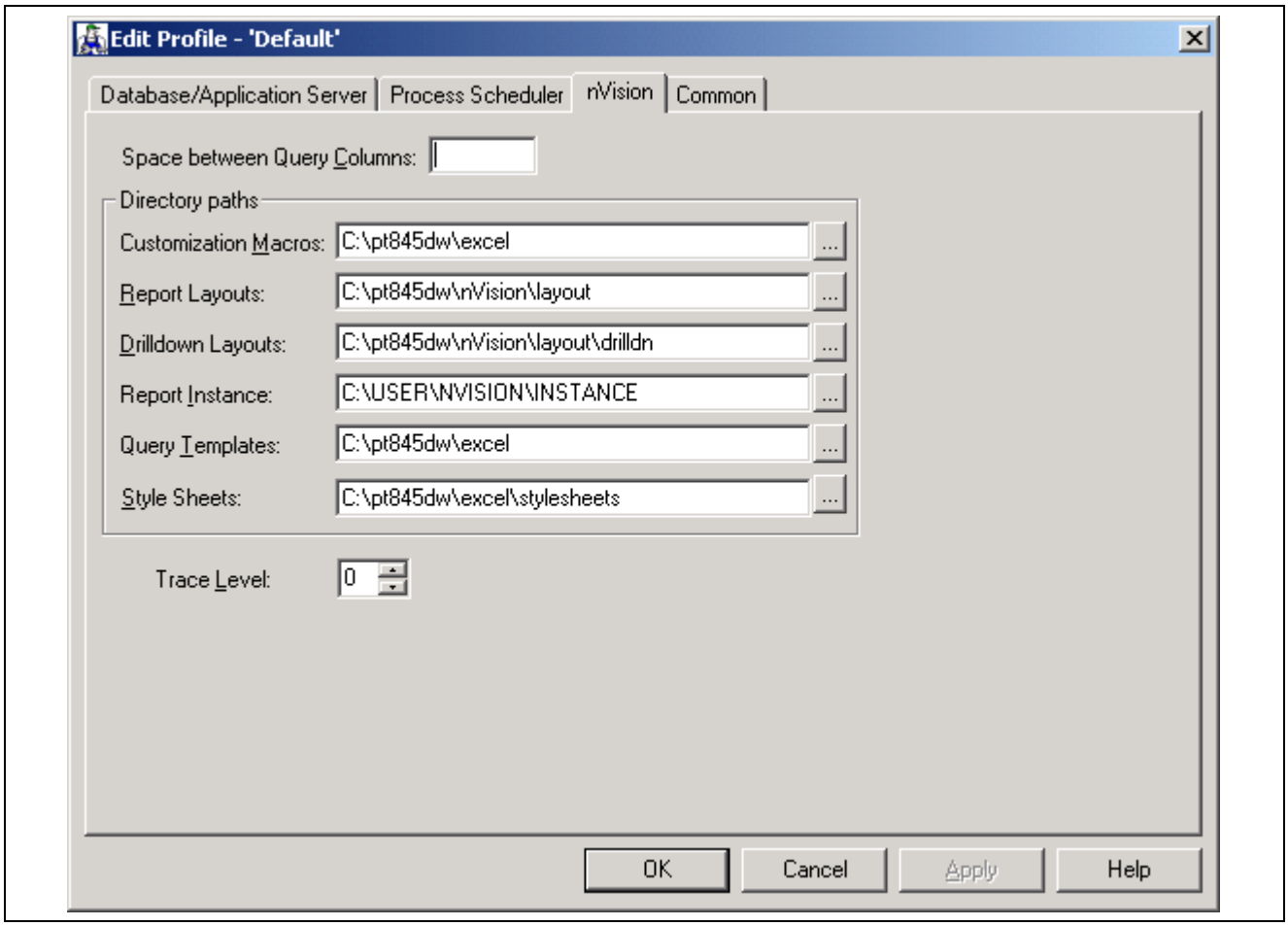
SQR Report Search 1, **SQR Report Search 2**, **SQR Report Search 3**, and **SQR Report Search 4** — Enter the directory paths that the SQR executable should use to locate SQR reports. SQR Report Search 1 is searched first, followed by SQR Report Search 2, and so on.

COBOL

COBOL Executables — Enter the path to \CBLBIN, where COBOL executables reside.

Configuring nVision

Access the nVision tab.



Edit Profile - nVision tab

Use this tab to specify where PS/nVision should look for files and how it should operate. PeopleSoft Query Link, the feature that enables you to send PeopleSoft Query output to a spreadsheet, also uses these settings.

Space between Query Columns

This parameter sets the number of blank Microsoft Excel characters that PeopleSoft Query Link places between query output columns. To eliminate column spacing, set Space between Query Columns to zero.

Directory Paths

Specify the locations of directories associated with PS/nVision jobs.

Customization Macros Enter the directory containing macros for PS/nVision and PeopleSoft Query Link. It is usually *PS_HOME*\excel.

Report Layouts Enter the location of PS/nVision layout fields.

Drilldown Layouts Enter the location of PS/nVision drilldown files, for example, *c:\user\nvision\layout\drilldn.*

Report Instance Enter the directory in which PS/nVision places report instances; for example, *c:\user\nvision\instance.*

Query Templates Enter the directory to look for the QUERY.XLT file, which defines the Microsoft Excel styles used to format output. The default is *PS_HOME*\excel.

Style Sheets Enter the directory where the NVSUSER style wizard locates nPlosion style sheets.

Trace Level Indicate whether you want PS/nVision to generate independent trace log files of two-tier activity, and at what level, for each nVision process. Select one of the following values:

- *0:* Disable tracing. This is the default value.
- *1:* Generate basic high level information.

  This setting can be used to check whether nVision has successfully launched and is able to connect to Excel and process the request. Some of the key entries in a level 1 trace log are:

  - Command Line Arguments.
  - Trace Level.
  - Excel Pid.
  - Run Control Name.
  - Report Id.
  - Business Unit.
  - Drill Layout.
  - Report Id.
  - Instance Name.
- *2:* Generate level 1 tracing plus high level code flow.
- *3:* Generate level 2 tracing plus runtime SQL statements.
- *4:* Generate level 3 tracing plus most function calls and output values.

  Use this setting to identify problems that are intermittent and hard to predict.

The trace log files are generated in the c:\temp directory, named with the format psnvs_*process_id*.nvt, for example, psnvs_1024.nvt. You can view these log files in a text editor.

See *Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Process Scheduler*, "Using Process Monitor," Viewing Process Detail Actions.

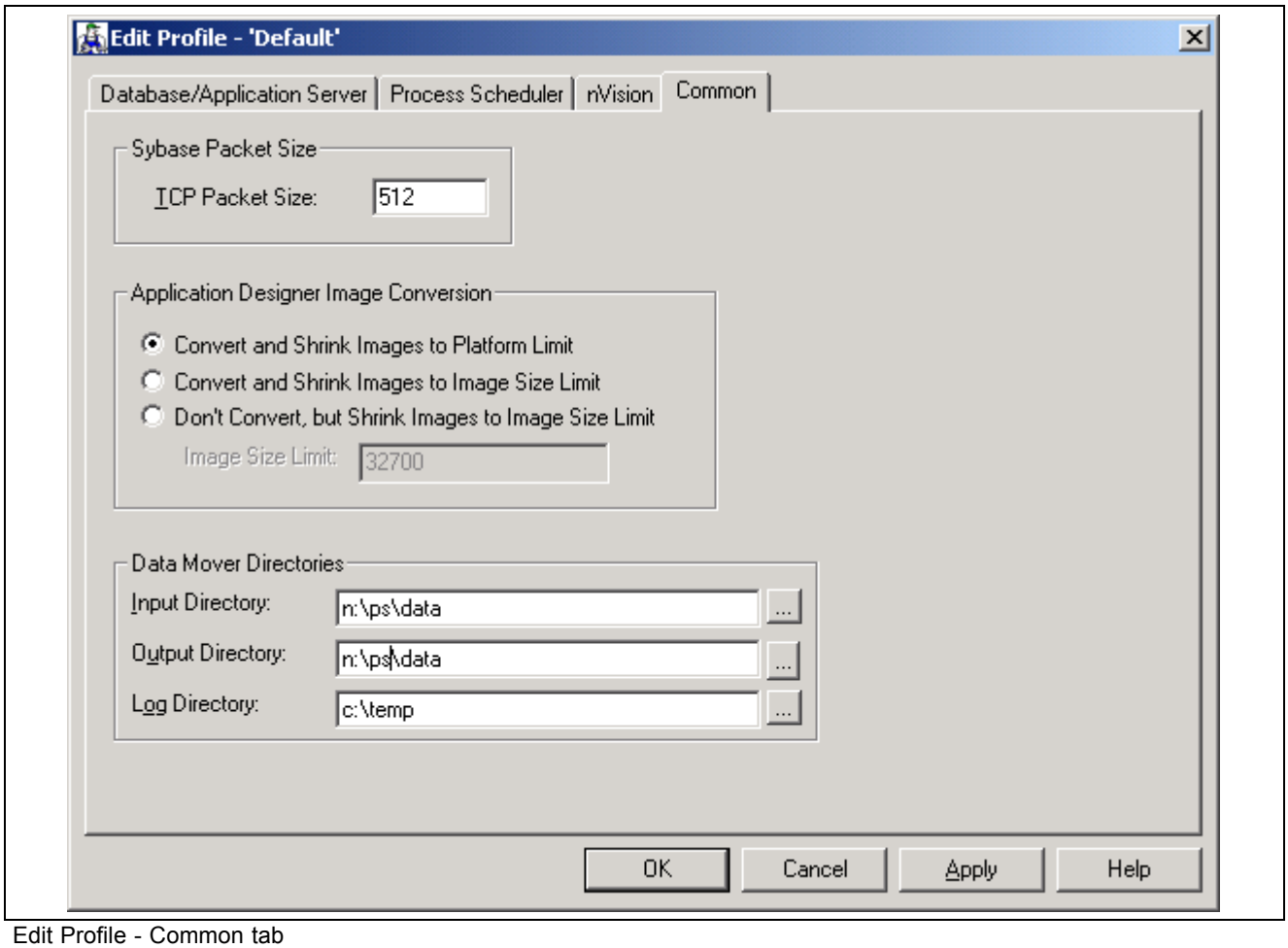
Note. Extensive tracing will affect PS/nVision performance. Two-tier log files aren't automatically purged by PS/nVision. Users must manually delete them from the temp directory to save disk space.

See Also

Enterprise PeopleTools 8.46 PeopleBook: PS/nVision

Specifying Common Settings

Select the Common tab.



Edit Profile - Common tab

Sybase Packet Size

Specify a TCP packet size. The minimum value is *512* and the maximum value is *65538.* The default packet size is *512.* If you change the packet size, make sure to make the corresponding changes to the Sybase server. See the material on Sybase administration and tuning on the PeopleSoft Customer Connection website, as well as your Sybase documentation.

See Your Sybase reference manuals.

Application Designer Image Conversion

When you upgrade to newer version of PeopleTools, you'll need to convert images to a new format, which may require more storage space. If the images exceed the record size limit of your platform, you can shrink the images to conform to this limit.

Convert and Shrink Images to Platform Limit Select to convert and shrink images to fit your selected database platform limit, as shown in the Image Size Limit field.

Convert and Shrink Images to Image Size Limit If you are upgrading to a different database platform, select this option and specify the correct value in the Image Size Limit field.

Don't Convert, but Shrink Images to Image Size Limit Select for images that have already been converted, but need to be converted so they meet the platform size limits.

Data Mover Directories

You can control several PeopleSoft Data Mover settings through PeopleSoft Configuration Manager.

Input Directory Enter the directory where PeopleSoft Data Mover should search for its input data (.DB) files. If no path is specified for the file named in the set input lines when running a PeopleSoft Data Mover script, Data Mover searches directories for the database file in the following order.

1. Specified output directory.
2. C:\TEMP.

Output Directory Enter the directory where PeopleSoft Data Mover scripts will be created. The default is *PS_HOME*\data.

Log Directory Enter the location of PeopleSoft Data Mover log files. The default is *PS_HOME*\data.

See Also

Enterprise PeopleTools 8.46 PeopleBook: Data Management, "Using PeopleSoft Data Mover"

Specifying Command Line Options

In addition to its GUI interface, PeopleSoft Configuration Manager offers command line options. Syntax for PeopleSoft Configuration Manager command line options is as follows:

pscfg -*command*

For example:

pscfg -import:n:\config\hr840.cfg

Import File

To import configuration settings from a named file, enter -import: *filename*.

Export File

To export the current configuration settings, enter -export: *filename*.

Run Client Setup

To run the Client Setup process, enter -setup.

Note. You must use the -setup command in conjunction with the -import command if you are setting up a new workstation.

Run Client Setup Without Displaying Messages

To run the Client Setup process without displaying messages or dialog boxes, enter -quiet.

Note. Output messages are written to a log file called %temp%\PSINSTAL.LOG.

Install ActiveX controls

To register ActiveX controls, enter -activex.

Note. ActiveX controls are registered during the Client Setup process. The -activex command enables you to register the ActiveX controls without running the entire Client Setup process.

Install Crystal Reports Runtime Files

To install Crystal Reports runtime files, enter -crystal.

Install MSS DSN

To install MSS DSN, enter -dsn.

Note. For Microsoft SQL Server, the server name value is used to automatically create your ODBC data source name.

Disable ODBC Driver Manager Installation

This command is only valid when used in conjunction with the -setup command. It disables the installation of the ODBC drivers during the Client Setup process. Use this command when you do not want to install the ODBC drivers on the client workstation when using the -setup command. To use this command, enter -noodbc.

Disable PeopleSoft ODBC Driver Installation

This command is only valid when used in conjunction with the -setup command. It disables the installation of the PeopleSoft ODBC driver during the Client Setup process. Use this command when you do not want to install the PeopleSoft ODBC driver on the client workstation when using the -setup command. To use this command, enter -nopsodbc.

Uninstall Workstation

To clear the PeopleSoft settings from the registry or uninstall the PeopleSoft workstation, enter -clean.

The -clean command removes the following items from the workstation:

- PeopleSoft registry settings.
- All cache files from the current \CACHE directory.

- Shortcut links.
- PeopleSoft program group.

Make sure that removing all of these items is acceptable before issuing the -clean command.

Help

To view PeopleSoft Configuration Manager command-line options online, enter -help or a question mark (?).

Setting Up the PeopleTools Development Environment

This section provides overviews of the PeopleTools development environment and the client setup process and discusses how to:

- Verify PS_Home access.
- Verify connectivity.
- Verify supporting applications.
- Use the Configuration Manager pages.
- Run the Client Setup process.

Understanding the PeopleTools Development Environment

Most user workstations are equipped with supported web browsers, but with no special PeopleSoft software installed. The traditional Microsoft Windows client, now called the PeopleTools development environment, is still supported.

The PeopleTools development environment runs on Microsoft Windows NT 4.0, Windows 2000, and Windows XP. This chapter describes how to configure these Windows-based clients using PeopleSoft Configuration Manager. As before, such clients can connect to the PeopleSoft database directly using client connectivity software (a two-tier connection), or through a PeopleSoft application server (a three-tier connection).

Understanding the Client Setup Process

Before running the Client Setup process, create all the profiles you need.

The Client Setup process does the following:

- Installs a PeopleSoft program group on the workstation.
- Sets up a Microsoft SQL Server system data source name using the server and database name information from the Startup tab.
- Installs the PeopleSoft ODBC driver required for Open Query and Crystal Reports.
- Installs Crystal Reports DLLs on the workstation.
- Configures a PeopleSoft ODBC data source name.

If the Install Workstation check box on the Client Setup tab is selected, these Client Setup functions are performed when you click OK or Apply from PeopleSoft Configuration Manager.

See Chapter 9, “Using PeopleSoft Configuration Manager,” Configuring Developer Workstations, page 195.

Note. Any files installed by the Client Setup process on the workstation from the file server, including ODBC driver files, use the paths specified in the default profile.

Verifying PS_HOME Access

To use the PeopleTools development environment, each workstation must have access to the file server *PS_HOME* directory (the high-level directory where PeopleSoft client executables were installed) and have a drive mapped to the directory. Workstation users must have read access to the *PS_HOME* directory.

Verifying Connectivity

Database connectivity is required on all Microsoft Windows-based clients that make two-tier connections to the database. A two-tier connection is required if any of the following is true:

- You sign in to the application in two-tier.
- You run PeopleSoft Data Mover scripts.
- You run COBOL and SQR batch processes on the client.

Verify Supporting Applications

Supporting applications must be installed on any Microsoft Windows-based client on which batch processes are run locally.

SQR

On Microsoft Windows-based clients, you can install SQR locally, or you can map to a copy installed on the file server. Because SQR does not require local registry settings, you can execute SQR from any Windows-based client once SQR has been installed to a shared directory. Installing SQR locally results in improved performance; over a slow network connection, the improvement is significant.

Crystal Reports

Optionally install Crystal Reports on Microsoft Windows-based two-tier clients. As with SQR, you can install Crystal Reports locally, or you can map to a copy installed on the file server. Because Crystal Reports does not require local registry settings, you can run Crystal Reports from any two-tier client once it has been installed to a shared directory. Installing Crystal Reports locally results in improved performance; over a slow network connection, the improvement is significant.

Crystal Reports requires that you install the PeopleSoft ODBC driver on the workstation where Crystal Reports processes run.

Microsoft Office

Install Microsoft Office on any two-tier client that runs PS/nVision or Microsoft Word batch processes. Microsoft Office must be installed locally, because it requires registry settings.

Using the Configuration Manager Pages

The following PeopleSoft Configuration Manager tabs apply to workstation users:

- Startup.

Controls the default values that appear in the PeopleSoft signon screen, as well as the location of the PeopleSoft cache on the client.

- Display.

  Controls language preference and other display options.

  Note. The language setting in PeopleSoft Configuration Manager determines language preferences for the PeopleTools development environment, regardless of user language preferences.

- Crystal/Bus. Interlink.

  Specifies the locations of Crystal Reports executables and the default location for reports generated using PeopleSoft Query. This location must be a directory to which the user has write access. This tab also specifies the location of PeopleSoft Business Interlink drivers.

- Trace.

  Controls SQL, PeopleCode, PeopleSoft Application Engine, message agent, and PeopleSoft API trace options.

- Profile.

  If multiple users will sign in to the workstation, you may need to set these options once for each user using the Profile feature.

Running the Client Setup Process

To run the Client Setup process:

1. Select the Client Setup tab.
2. In the Group Title text box, enter the name of the program group for the icons you want on the client workstation.

   The default name is *PeopleSoft 8*.
3. Select check boxes to create shortcut links for PeopleSoft applications that you want to access from the workstation.

   When you run the Client Setup process, it removes existing shortcuts in the PeopleSoft 8 program group and installs shortcuts for the applications that you have selected. If you later want to install or uninstall shortcuts, you can always run Client Setup again.
4. Select the Install PeopleSoft ODBC Driver check box to install the PeopleSoft ODBC driver and set up a user ODBC data source name as required by PeopleSoft Open Query and by Crystal Reports.
5. Select the Install Workstation check box.

   Client Setup runs when you click Apply or OK in PeopleSoft Configuration Manager. If this check box is not selected, the Client Setup process creates or updates settings in the registry, but it doesn't set up the PeopleSoft 8 program group or install local DLLs.
6. Click Apply to run the Client Setup process and apply other PeopleSoft Configuration Manager settings.

   Click ODBC Administrator to directly access the Microsoft ODBC Administrator to verify the installation and configuration of the ODBC DSN.

   If you install ODBC Driver Manager 3.5, reboot the workstation after running the Client Setup process.
7. To view a list of the files installed and actions taken by the Client Setup process, open the psinstal.log file in your Temp directory.

See Also

Chapter 9, “Using PeopleSoft Configuration Manager,” Configuring Developer Workstations, page 195

CHAPTER 10

Using PeopleTools Utilities

This chapter provides an overview of the PeopleTools Utilities and discusses how to:

- Use the System Information page.
- Use administration utilities.
- Use audit utilities.
- Use debug utilities.
- Use international utilities.
- Use optimization utilities.
- Use PeopleSoft Ping.

Understanding the PeopleTools Utilities

As you work with the PeopleSoft system, you find that there are some administrative tasks that you only need to perform occasionally. These tasks include such things as maintaining error messages and setting DDL model defaults. The PeopleTools Utilities menu is where you find tools for accomplishing some of these more infrequent tasks.

The documentation of the utilities matches the menu structure of the Utilities interface. For example, the PeopleTools Options utility is under the Administration menu in the Utilities interface; therefore, the documentation for PeopleTools Options is in the Using Administration Utilities section in this chapter.

Also, in many cases this book refers to other PeopleBooks for the detailed documentation of a utility.

Using the System Information Page

This section provides an overview of the system information page and discusses how to view the system information page

With the combination of accessing PeopleSoft applications with a browser, single signon between databases, and the PeopleSoft Portal, users and system administrators need a quick tool to provide orientation information and information regarding the current environment. For this reason, PeopleSoft provides the system information page.

Understanding the System Information Page

With single-signon and the portal, it may not be apparent to all end users just exactly what databases or applications they are currently accessing. Viewing environment information can help end users orient themselves.

In most cases, the administrators use the system help page to aid in troubleshooting. If a user has trouble accessing a particular application, the system administrator can instruct the user to provide the system information that appears in the help page so that the administrator can immediately identify the current application server, database, software version, operating system, and so on.

Viewing the System Information Page

To view the System Information help page, you press the CTRL+J hotkey while a PeopleSoft page is active. The following example illustrates the type of information that appears.



System Information help page

To return to the previous page, click continue.

The following table briefly describes each item:

Item	Description
Browser	The browser version and type, such as Internet Explorer or Netscape.

Item	Description
Operating System	The operating system that runs on the computer on which the browser is running. For example, this refers to the operating system of the end user's workstation or the operating system running on a kiosk machine. It does not refer to the operating system that runs on the application server, web server, or database server.
Browser Compression	Indicates if browser compression is enabled in the Compress Responses field on the General page of the current web profile. Values are: • ON: The flag is on in the web server configuration and the page is compressed. The compression type is either gzip or zip. • OFF: The page is not compressed because the flag is cleared in the web profile. • OFF (not supported): The page is not compressed because the browser doesn't support compression, however the flag is turned on in the web profile.
Tools Release	The version of PeopleTools that is currently installed at the site. For example, PeopleTools 8.4, 8.40.01, and so on.
Application Release	The version of PeopleSoft applications that are currently installed at the site.
Service Pack	Typically, updates to PeopleSoft applications arrive in the form of a service pack. This item shows the current service pack that is applied to the applications.
Page	The current page that the user is accessing.
Component	The component to which the current page belongs.
Menu	The name of the menu under which the component appears.
User ID	The user ID of the user that is currently accessing PeopleSoft.
Database Name	The name of the database that the user is currently performing a transaction in.

Item	Description
Database Type	The type of the current database, as in Microsoft, Oracle, DB2 UDB, and so on.
Application Server	The domain name server name or Internet Protocol (IP) address and the JSL port number.

Depending on the site's policy, you may not want to reveal the user ID, database name, database type, and application server information that is readily available. You use the Show Connection Information check box on the Debugging page of the current web profile to determine what appears when a user presses CTRL+J.

If you select Show Connection Information, all information appears on the System Information help page. However, if you clear this check box, the User ID, Database Name, Database Type, and Application Server information don't appear on the page.

See Also

Enterprise PeopleTools 8.46 PeopleBook: Internet Technology, "Configuring the Portal Environment," Configuring Trace and Debug Options

Using Administration Utilities

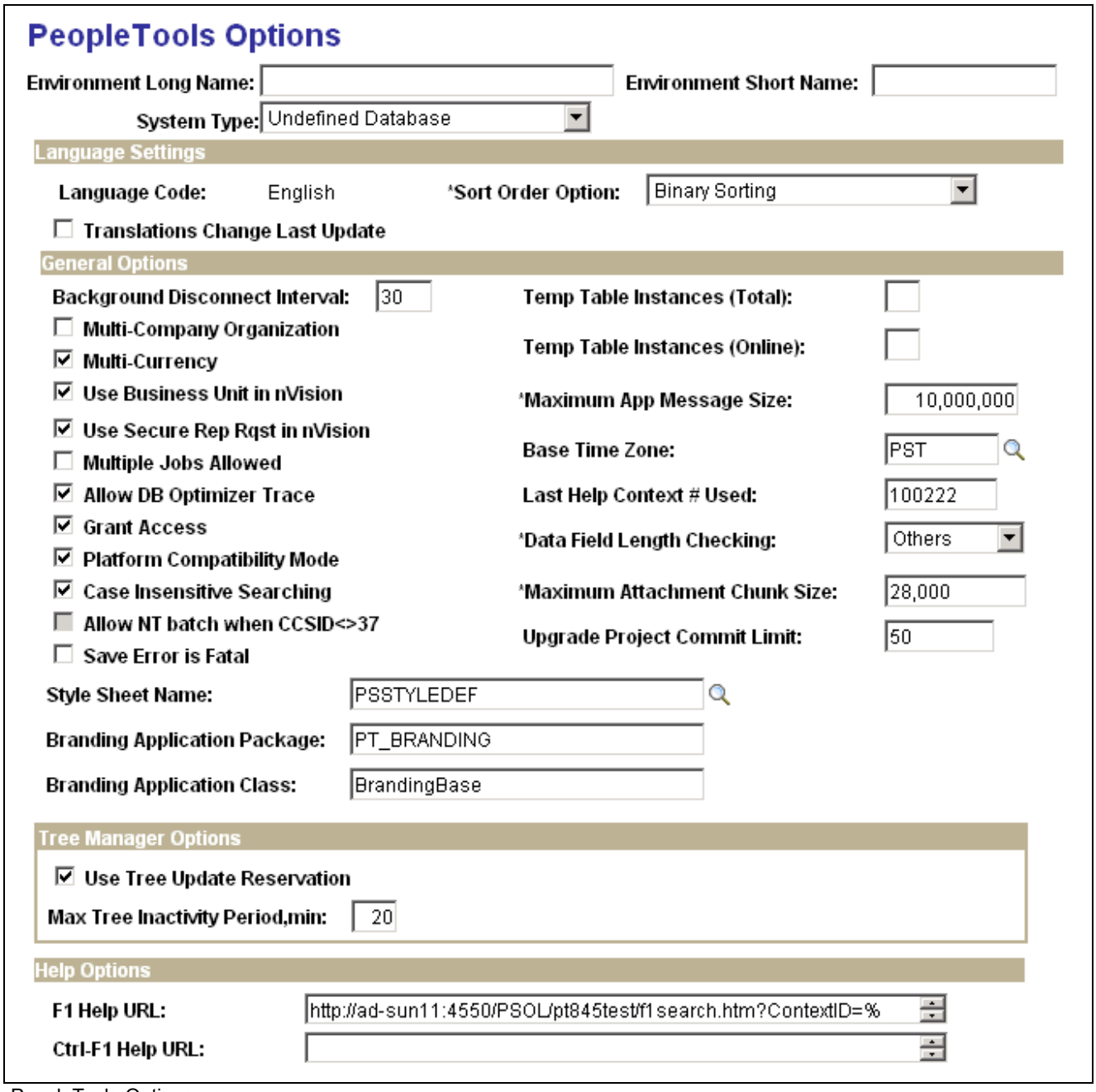
This section discusses:

- PeopleTools Options.
- Message Catalog.
- Spell Check System Dictionary.
- Translate Values.
- Load Application Server Cache.
- Tablespace Utilities.
- Tablespace Management.
- DDL Model Defaults.
- Strings Table.
- XML Link Function Registry.
- Merchant Integration Utilities.
- TableSet IDs.
- Record Group.
- TableSet Control.
- Convert Panels to Pages.
- Update Utilities.

- Remote Database Connection.
- URL Maintenance.
- Copy File Attachments.
- Query Monitor.
- Sync ID Utilities.
- Gather Utility.

PeopleTools Options

Select PeopleTools, Utilities, Administration, PeopleTools Options to access the PeopleTools Options page. Use this page to set a number of options that affect multiple PeopleTools and applications, such as language options and change control settings:



PeopleTools Options page

Environment Long Name and Environment Short Name

Enter a long name and a short name for the current PeopleSoft environment. PeopleSoft software update tools use this information to identify the database when searching for updates. For example, enter *Customer HR Demo Environment* for the long name, and *HR Demo DB* for the short name.

System Type

Select an appropriate system type from the dropdown list, for example, *Demo Database*. This information helps to further identify the current environment for the purpose of searching for and applying software updates.

Language Settings

Language Code

The base language of an application is the application's primary language, normally the language that is used most commonly throughout the enterprise. A database can have only one base language. All other language translations that are stored in the database are referred to as nonbase languages (or sometimes as foreign languages).

You can't change the Language Code setting on this page. This field is for display purposes only. To change the base language, use the SWAP_BASE_LANGUAGE Data Mover command.

The Language Code field box identifies the database's base language.

Translations Change Last Update

If you select the Translations Change Last Update check box, and you use the PeopleTools translate utilities to translate objects, the system updates the Last Updated information of the translated object to the date/time/userid of the translation. If it's turned off, then the date/time/userid of the object does not change when it's translated.

Note. This only applies when you're using the page-based PeopleTools translation utilities; the Translation Workbench always updates the last updated information.

Sort Order Option

Select the sort order that is appropriate for the site.

See the Global Technology PeopleBook for descriptions of the options.

General Options

Background Disconnect Interval

The value in seconds that you enter here acts as the default for Security Administrator profiles.

Multi-Company Organization

Turn on Multi-Company Organization if more than one company makes up the organization.

This option affects how Application Processor displays company-related fields in search dialogs and pages. See the HRMS documentation for more details.

Multi-Currency

The Multi-Currency setting is a systemwide switch that enables automatic formatting of currency amount fields that have associated currency control fields. Another function of this setting is to globally display currency control fields. If you turn off this option, automatic formatting based on currency control fields is no longer active and all currency control fields are thus hidden.

When the Multi-Currency setting is on, it also validates user-entered currency data against the currency's defined decimal precision. This validation causes the system to issue an error if a user attempts to enter a decimal precision that is greater than that which is allowed by the currency code definition.

Under most circumstances, leave Multi-Currency selected.

Use Business Unit in nVision

Deselect the Use Business Unit in nVision option if you're using an HRMS database. Otherwise, select it.

Use Secure Rep Rqst in nVision

Select this check box if you want the report request in nVision to be secure. The default setting is selected.

Multiple Jobs Allowed

Selecting Multiple Jobs Allowed enables HRMS systems to support employees holding concurrent jobs with more than one set of enrollments.

This option affects how Application Processor displays employee-record-number-related fields in search dialogs and pages. See the HRMS documentation for more details.

Allow DB Optimizer Trace

Typically, you turn on this trace only during periods in which you are collecting detailed performance metrics. When you are not tuning your performance, the DB Optimizer trace should be turned off.

Grant Access

When adding a new operator by using PeopleTools Security Administrator, the system automatically grants the new operator select-level access to the three PeopleTools SQL (Standard Query Language) tables that she needs to log on to. If you are using a SQL security package and do not want PeopleTools Security Administrator to perform any SQL grants, turn off Grant Access.

Platform Compatibility Mode

Enables you to add the capability to set a database compatibility mode as an overall database setting, forcing developers to create applications by using all platforms as the least common denominator. This option enables developers, who create applications for multi-platform deployment, to catch platform-specific issues at design time rather than during testing.

Note. This option is used mainly by PeopleSoft development teams that need to develop applications to run on all supported database platforms. To support numerous database platforms, PeopleSoft needs to have a tablespace for each physical table record definition.

If platform compatibility is enabled for a database, the system forces developers to enter a tablespace name when saving a record definition regardless of the current platform. If this option is disabled, you are only prompted for a tablespace name if you are developing on a platform that utilizes tablespaces. This prevents table record definitions being added to the database without a tablespace name.

Case Insensitive Searching

Enables you to enable case-insensitive searching for the PeopleSoft search records.

Note. This is not associated with the Verity search technology.

Allow NT batch when CCSID <>37

Enables you to override non-z/OS COBOL batch restrictions. If the DB2 z/OS database's CCSID is NOT 37, PeopleSoft blocks batch COBOL from running against z/OS Databases on Windows unless you choose this override.

Note. Even if you choose this override, if you use %BINARYSORT() in the COBOL, the system issues an error on Windows. RemoteCall COBOL can run on Windows and UNIX regardless of this option setting, even if CCSID is NOT 37, but the system issues an error.

Save Error is Fatal

Select this option when you have non-repeatable PeopleCode logic in your application's SavePreChange or Workflow. In previous releases, PeopleSoft applications were coded to assume that errors during save are always fatal, but the current PeopleTools release no longer behaves this way. Use this option to ensure predictable behavior with your application without having to modify your older application code.

This check box is cleared by default. If you get an error during save processing, the transaction continues and you're allowed to attempt to save again. When this option is selected, if you get an error during save processing the transaction is aborted and all changes are lost. This applies to errors that occur between and including the SavePreChange event to the SavePostChange event. It also includes the component processor save processing. It doesn't include errors from the SaveEdit event.

For example, suppose you have some calculations that occur in SavePreChange which are based on the buffers and also modify the buffers. If there's an error during the save and you attempt to save again, the calculations are repeated, but this time based on the buffers that were already modified by the first time the calculations where done. Therefore the second time the calculations are done they will be incorrect, which could lead to incorrect data being saved to the database. In this case you would want to turn on the Save Error is Fatal option, because a fatal error on save is more desirable than incorrect data being put into the database.

Style Sheet Name

All PeopleSoft applications reference the PSSTYLEDEF style sheet by default. You can set the individual style sheets in PeopleSoft Application Designer, and these override the general style sheet for the application, which is set here.

Temp Table Instances (Total):

The value that you specify in the Temp Table Instances (Total) edit box controls the total number of physical temporary table instances that PeopleSoft Application Designer creates for a temporary table record definition when you perform the Build process.

This value indicates the total number of undedicated temporary table instances. The maximum number of temporary table instances that you can specify is 99.

Temp Table Instances (Online)

Enter the available online instance values. When you invoke a process online, PeopleTools randomly allocates a single temporary table instance number to programX for all of its dedicated temp table needs. The higher the number of online instances that is defined, the less likely it is for two online processes to get the same value.

Maximum App Message Size

There is practical limit to how large a message can be. Enter the maximum message size; this does not set individual message definition, but defines the size for all application messages.

Base Time Zone

Although you can display time data a number of different ways, PeopleSoft databases store all times relative to a systemwide base time zone. You can adjust the display of the time that an end user sees using the Use Local Time Zone (LTZONE) setting in PeopleTools, Personalizations.

This base time zone is the one that the database server uses. In order for PeopleSoft to properly manage time data, the system needs to know which time zone that is. Set the Base Time Zone to the time zone that the database server's clock uses.

Note. After changing this setting, reboot any application servers that are connected to the database. It is critical for the correct operation of the system that this time zone match the time zone in which the database is operating. Any discrepancy in the base time zone as defined in this page and the time zone in which the database system is operating leads to inaccurate time processing.

Last Help Context # Used This field is no longer used.

Data Field Length Checking Normally, field length validation is based on the number of characters that are allowed in a field. For example, a field defined as CHAR(10) in PeopleSoft Application Designer holds ten characters, regardless of which characters you enter. In a Unicode database, double-byte characters, such as those found in Japanese, are counted the same as single-byte characters, such as those found in the Latin alphabet.

If you create a non-Unicode database, the field length in PeopleSoft Application Designer represents the number of bytes that are permitted in the field, not the number of characters. When the non-Unicode database uses a single-byte character set (SBCS), you can only enter single-byte characters, so the number of characters and the number of bytes are the same. However, because double-byte character sets (DBCS) typically allow a mix of single- and double-byte characters, the number of characters that are allowed in a field in a non-Unicode DBCS database varies. This is true for both shifting and non-shifting double-byte character sets.

For example, a if a user enters ten Japanese characters into a field that is defined as CHAR(10) in PeopleSoft Application Designer, this string needs 20 bytes of storage in a nonshifting double-byte character set and 22 bytes of storage in a shifting double-byte character set. This ten-character input fails insertion into both these databases.

Use the Data Field Length Checking option to ensure field length validation appropriate to the database's character set. Values are DB2 MBCS, MBCS, and Others.

Choose Others if you are using a Unicode-encoded database or a non-Unicode single-byte character set database. This prevents special field length checking. As discussed above, these types of databases do not require such checking.

Choose DB2 MBCS if you are running a Japanese database on the DB2 UDB for z/OS platform. This enables field length checking based on a shifting DBCS character set.

Choose MBCS if you are running a non-Unicode Japanese database on any other platform. This enables field length checking based on a nonshifting DBCS character set.

The non-Unicode DBCS settings are specifically oriented towards Japanese language installations, as Japanese is the only language that PeopleSoft supports in a non-Unicode DBCS encoding. All other languages requiring double-byte character sets are only supported by PeopleSoft by using Unicode encoded databases.

Maximum Attachment Chunk Controls the size of the file attachments that you store in the database. The default is 28000 kilobytes.

Upgrade Project Commit Limit Sets the limit on how many rows can be modified by an upgrade project before the system issues a COMMIT statement.

Branding Application Package Specifies the application package that contains the branding application classes to generate the portal headers, footers and menu pagelet icons. The default is the standard PeopleTools branding, PT_Branding. For Enterprise Portal, a different branding application package is specified.

Branding Application Class The main branding application class that generates header, footer, and menu pagelet icons. The default is the standard PeopleTools branding, BrandingBase. For PeopleSoft Enterprise Portal, a different branding application class from a different branding application package is used. It generates different header, footers , and menu pagelet icons dynamically, based on the user role or security.

Help Options

FI Help URL This setting only applies to the Windows environment (such as PeopleSoft Application Designer) when the user presses F1 or selects Help, PeopleBooks Help while in PeopleTools.

The F1 Help URL can direct users to any location that is on the web, such as a custom help system or the website for the company's help desk. It can be a fully qualified uniform resource locator (URL), which is passed literally to the browser, or it can contain one or both of these system variables.

%CONTEXT_ID% is the object name or context ID of the currently displaying page or dialog box.

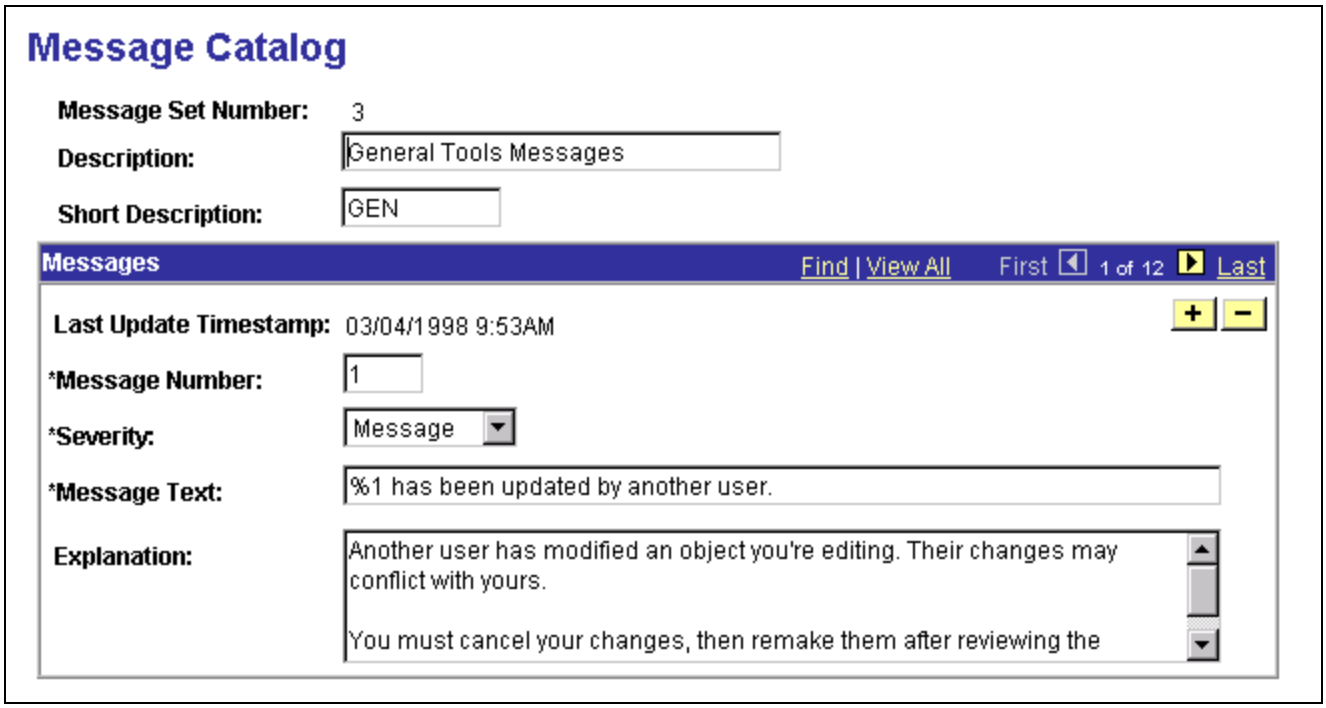
%LANG_CD% is the three-letter language code for the user's preferred language.

Ctrl-F1 Help URL This setting only applies to the Windows environment (such as PeopleSoft Application Designer).

The Ctrl+F1 URL allows you to provide an alternate location for help. For example, you may set the main F1 Help URL to the PeopleBook and the Ctrl+F1 for the company's help site.

Message Catalog

Select PeopleTools, Utilities, Administration, Message Catalog to access the Message Catalog page.



Message Catalog page

You add and maintain system messages by using the Message Catalog page. PeopleSoft error messages are stored in the Message Catalog, and organized by message set number. Each message set consists of a category of messages, ranging from PeopleTools Message Bar Items and PeopleCode Runtime Messages to PeopleSoft Payroll and PeopleSoft General Ledger application messages.

Message Set Number Identifies the message set.

Description The Message Set Description is a reference that is used on reports and pages for easy identification.

Short Description The Message Set Short Description is a reference that is used on reports and pages for easy identification.

Message Number Each message set consists of one or more rows of messages that are identified by a message number.

Severity You assign each message a severity, which determines how the message appears and how the component processor responds after the user acknowledges message. The severity levels are:

Cancel: This severity should be reserved for the most severe of messages, as when a critical error occurs and the process must be aborted or a machine needs to be shut down. To indicate how rarely this severity level is appropriate, of all PeopleTools messages only five or so have a severity level of Cancel. In almost all cases, you use one of the other severity levels.

Error: Processing stopped, and data cannot be saved until the error is corrected.

Message: This is an informational message and processing continues normally.

Warning: User can decide to either stop or continue processing despite the error.

Message Text In the Message Text edit box, you see the message text. Any reference to the characters %n, as in %1 or %2, is replaced by parameter values that the system provides.

Explanation The Explanation text provides a more in-depth explanation of why the message is generated and how to fix the problem. This text appears below the Message Text when the message appears.

PeopleTools uses some messages, but the applications use the other messages, which get called by the Error, Warning, Message Box, MsgGet, and MsgGetText built-in PeopleCode functions.

Note. You can create messages and message sets to support new or customized functionality in the system. You can also edit the messages that PeopleSoft delivers. In both of these cases, remember that PeopleSoft reserves all message set numbers up to 20,000. If you add a message set or edit a message set with a number that is less than 20,000, it may be overwritten in future upgrades:

To add a message set:

1. Select Utilities, Administration, Message Catalog, and on the search page click Add New Value.
2. Enter the value of the new Message Set Number and click OK.
3. Enter a description and short description of the type of messages that this message set contains.

   Try to group the messages logically. For instance, create one message set for the new budgeting application and a different one for the customized billing pages.
4. Add messages.
5. Save your work.

To add a message

1. Open the desired message set.
2. In the Message Catalog page, click the plus sign button to add a new row.

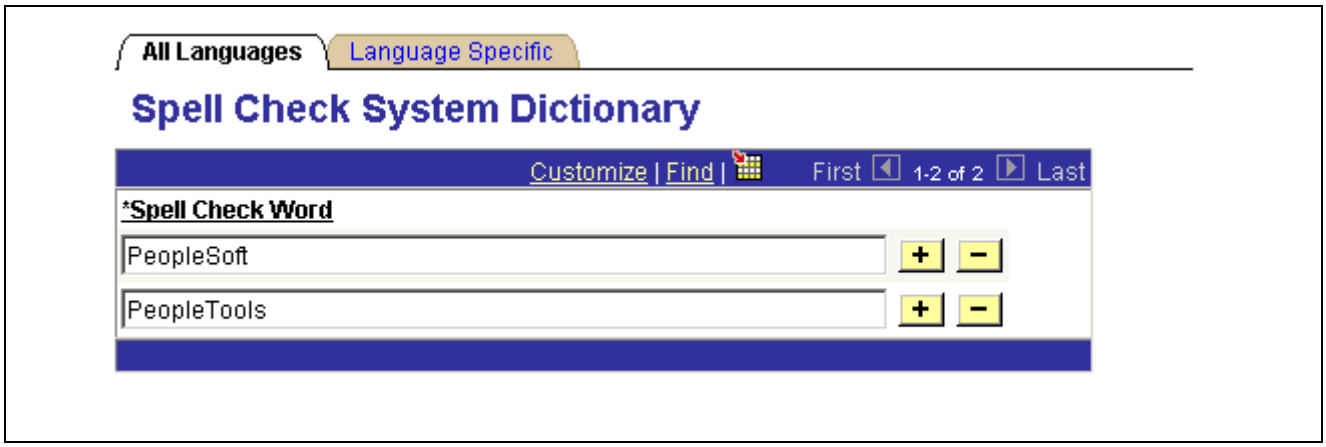
   The Message Number value is automatically set to the next unassigned number in the message set.
3. Select a Severity level, enter message text and a detailed explanation.
4. Save your work.

Spell Check System Dictionary

PeopleSoft PeopleTools provides personal and system-level dictionaries. End users and system administrators can add words to the dictionary for use with the spell check feature. Typically, system administrators add words to the system-level dictionary that are used company-wide; end users add additional role-specific terminology to their personal dictionaries.

Select PeopleTools, Utilities, Administration, System Dictionary to access the system-level dictionary.

Select the All Languages page to enter words that are valid across all languages. Select the Language Specific page for those words that are valid to a specific language:



Spell Check System Dictionary page

To add words to the system dictionary by language:

1. Select Spell Check System Dictionary, Language Specific.
2. Select the desired language from the Spell Check Language drop-down list box.
3. Select Session to add a word to the current session's spell check dictionary. After saving this word, the language field refreshes to the current spell check language.
4. Enter the word (maximum 40 characters) that is to be added in the Spell Check Word field.
5. Save your changes.

Case Sensitivity for Spell Check

The words that you add to your personal dictionary are case-sensitive and are validated by the following rules:

1. If the added word is all lower case, such as worklist, then the following are considered valid:
   - Exact match, all lower case (worklist).
   - All uppercase (WORKLIST).
   - Initial capitals (Worklist), regardless of its position in the sentence. Mixed case (WorkList) is considered incorrect.
2. If the added word is all uppercase, such as CRM (customer relationship management), then only an exact match is valid.
3. If the added word is in initial capitals, such as California, then only an exact match and all upper case (CALIFORNIA) are considered valid.
4. If the added word contains an embedded capital letter, such as PeopleSoft, then only an exact match is valid. Therefore, if case is not relevant to the validity of the word, use all lower case.

Table Structure for Word Storage

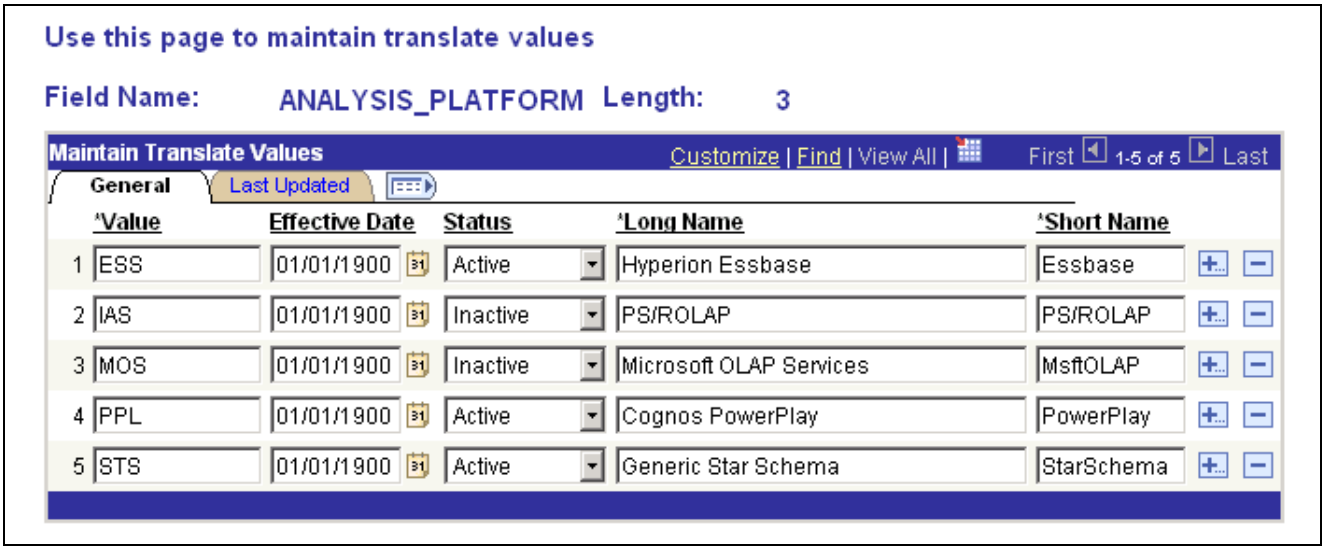
System and personal words are stored in the database in the PSSCWORDDEFN table with the following fields:

- SCOPRID indicates whether a word is a system word or a user's personal word.
- SCLANG stores the dictionary language for which the word is considered valid. If the system administrator chooses to store the word for all languages, this field is left blank.
- SCWORD stores the actual word, with a maximum length of 40 characters.

Translate Values

You use the Translate Values interface to maintain the values in the translate table. If it's allowed by site security administrators, power users can now learn to add their own pick lists (translate values) to an application:

Select PeopleTools, Utilities, Administration, Translate Values to access the Maintain Translate Values page.



Maintain Translate Values page

Value Enter the value for the translate selection.

Effective Date Specify a date for the value to become active.

Status Specify whether the value is active or not.

Long Name Enter a long description for identification. There is a 30-character limit.

Short Name Enter a shorter description for identification. There is a 10-character limit.

See Also

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Application Designer, "Creating Field Definitions," Using the Translate Table

Load Application Server Cache

The Load Application Server Cache page enables you to invoke an Application Engine program, called LOADCACHE, which preloads the cache for the application server. You need to run this program only if you intend to implement shared caching on the application server, which you configure by setting the ServerCacheMode parameter in the application server configuration file.

Load Cache and Application Server Caching

Each PeopleTools server process has two types of cache: memory cache and file cache. Memory cache is always enabled for all processes, but file cache can be configured by an administrator. This section describes populating and using a shared file cache.

The LOADCACHE program caches all of the PeopleTools object metadata into the cache directory that you configure. This is the equivalent of having a user access every page in the system once so that all the metadata is stored in cache. The shared cache also contains metadata for other application objects, such as application messages and Application Engine programs

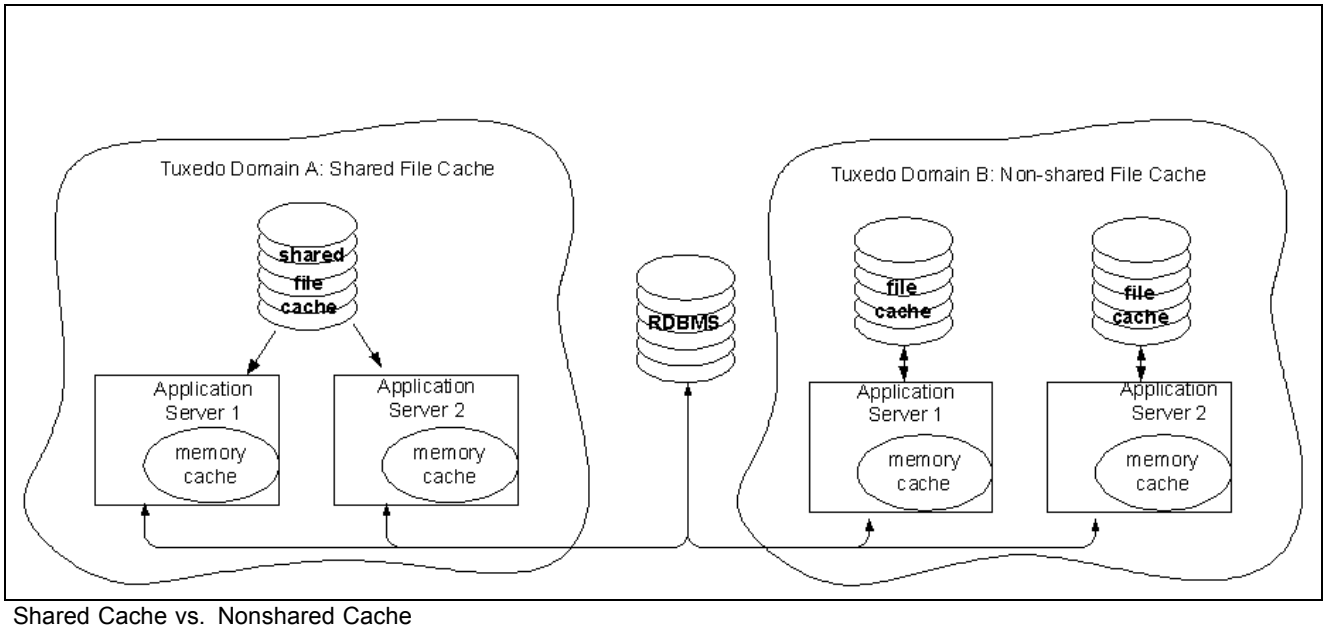
Using the cache options, the application server is recommended for optimal performance, but the underlying benefit of preloading the cache and using shared cache on the application server is predictable performance. For instance, by preloading the cache, users don't have to wait for the system to cache an object if it's the first time that the system accesses the object. Because the cache is preloaded with all the database objects, the system retrieves all of the required objects from the cache. This provides a significant improvement in first-time transactions and large transactions.

If you elect to implement the shared cache option on the application server, consider the following items:

- You need to run the LOADCACHE program at least once. As the PeopleTools metadata objects change, items that are in the shared cache are marked invalid but are not rewritten. This includes design time changes, upgrades, patches, and so on.
- The first time that you run the LOADCACHE program, it can take from 2 to 30 hours to complete. The time of the program run depends on the number of active languages that are set in the PSLANGUAGES table, the size of the database, and the performance of the machine. Subsequent program runs complete in less time if there is already valid cache data in the target cache directory, as the program is designed only to update the changed objects after the staging directory is already loaded.
- If you update PSSTATUS.LASTREFRESHDTTM, the system marks all items in the shared cache as invalid and you need to rerun LOADCACHE from scratch.

Note. The output is not portable to different operating systems. For instance, if you generate the cached metadata onto a Windows machine, you can't copy the cache files to a UNIX machine.

The following example graphically depicts the shared cache and the nonshared cache architecture:



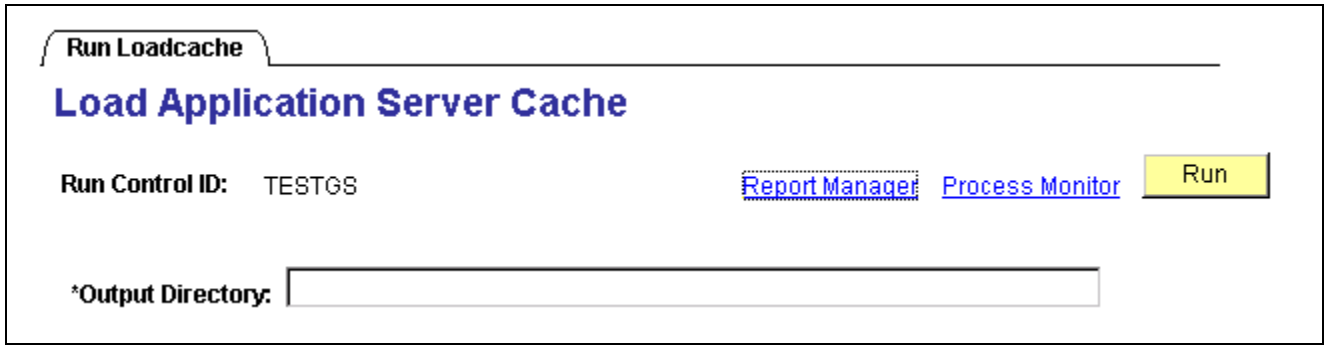
Shared Cache vs. Nonshared Cache

Note. Shared cache can be used without running the LOADCACHE program, however you still need to load the cache through some other mechanism. If you do not preload the cache, then shared cache is equivalent to having no file cache at all. Using the LOADCACHE program to load the file cache is the best option.

Running the LOADCACHE Program

This page enables you to run the Application Engine LOADCACHE program.

Select PeopleTools, Utilities, Administration, Load Application Server Cache to access the Load Application Server Cache page.



Load Application Server Cache page

To create and deploy a shared cache:

1. Make sure that the database that the application server runs against produces a clean SYSAUDIT report.

   If SYSAUDIT is not clean, the LOADCACHE program might fail.

2. Ensure that server file caching for all types is enabled.

   In the PSAPPSRV.CFG (application server configuration) file, the EnableServerCaching parameter should be set to *2* (the default) or commented out.

   The LOADCACHE program reads this setting and caches metadata according to the value specified.

   Note. Leave the ServerCacheMode parameter set to its default value of *0*.

   ---

3. Access the Load Application Server Cache page with an appropriate Run Control ID.
4. In the Output Directory field, enter the value of *PS_HOME* (the location of your PeopleSoft root directory); for example, C:\psft846.
5. Click Run.

   The Process Scheduler Request page appears.

6. Specify the name of the server that you want to run the process, and click OK to launch the LOADCACHE program.

   The first time that you run the program, the process may take four to five hours. The LOADCACHE program creates the cache files in the following directory:

   PS_HOME\appserv\prcs*ProcessScheduler_domain*\cache\cache\stage\stage

   Where *ProcessScheduler_domain* is the Process Scheduler domain in which you ran this program. After you invoke the program, you can use the Report Manager and Process Monitor links to monitor the progress of the program that is run.

7. Shut down the application server domain.
8. Enable shared caching with the ServerCacheMode parameter (ServerCacheMode=1), and reconfigure the domain so that the changes are reflected.

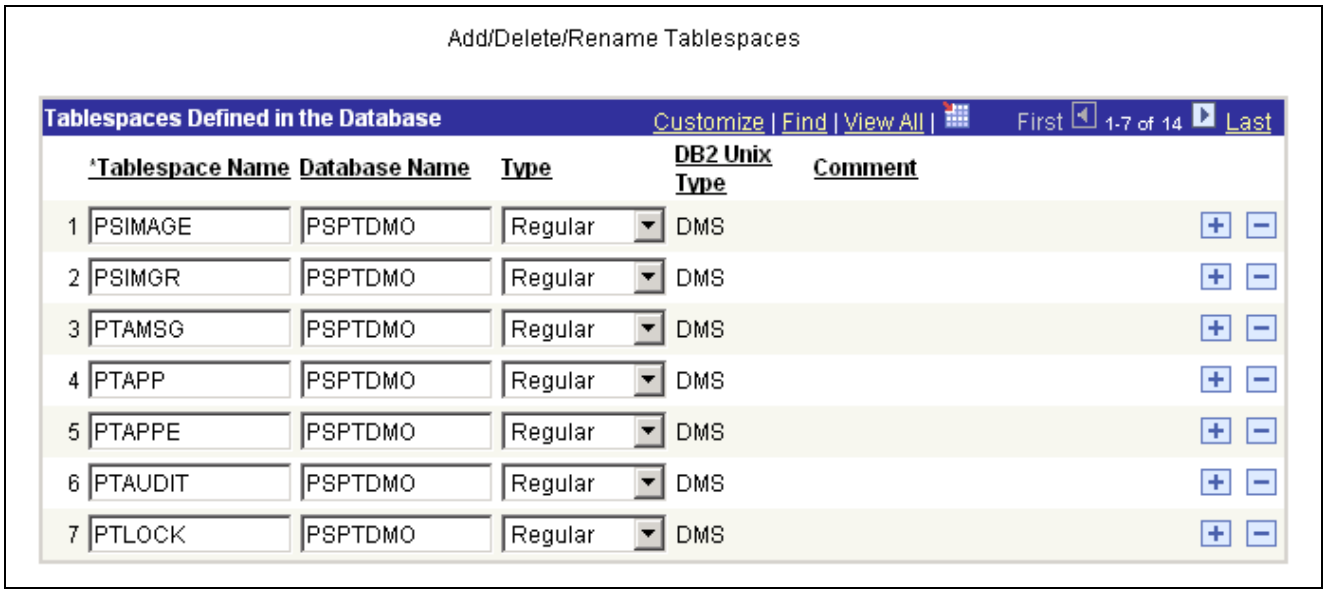
Note. When you enable share cache but the SHARE directory is not set up properly, a warning message is generated in the application server log file to alert you that there are no cache files in the cache directory.

9. Copy the contents of the output directory into the \cache\share directory for the appropriate application server domain.
10. Reboot the application server domain.

Tablespace Utilities

Select PeopleTools, Utilities, Administration, Tablespace Utilities to access the Tablespace Utilities page.

To comply with requirements for DB2 UDB for z/OS, the Tablespace Utility now includes both tablespace name and database names when you define a tablespace using the Tablespace Management page. Use the Add/Delete/Rename Tablespaces page to change the list of tablespace and database names.



Tablespaces Defined in the Database page

Add SQL Space

SQL Space Name Enter the name of the SQL space that you want to add.

Database Name Enter the database name into which you want to add the space.

Comment Enter any internal documentation that is required to identify the space and its purpose.

Add Adds the SQL space to the database.

Delete SQL Space

Existing SQL Space Name Enter or look up the name of the SQL space that you want to delete.

Delete Deletes the specified SQL space.

Rename SQL Space

Existing SQL Space Name	Enter or look up the name of the SQL space that you want to rename.
New SQL Space Name	Enter the new name for the SQL space.
Comment	Enter any internal documentation that is required to identify the space and its purpose.
Rename	Renames the specified SQL space.

Tablespace Management

Select PeopleTools, Utilities, Administration, Tablespace Management to access the Tablespace Management component (PSTBLSPCCOMP).

These pages enable you to modify the tablespace definition.

Tablespace Defn Page

This page shows the identification values for the tablespace.

Tablespace List Page

This page is where you add records to a particular tablespace. Use the plus and minus buttons to add and delete rows from the list.

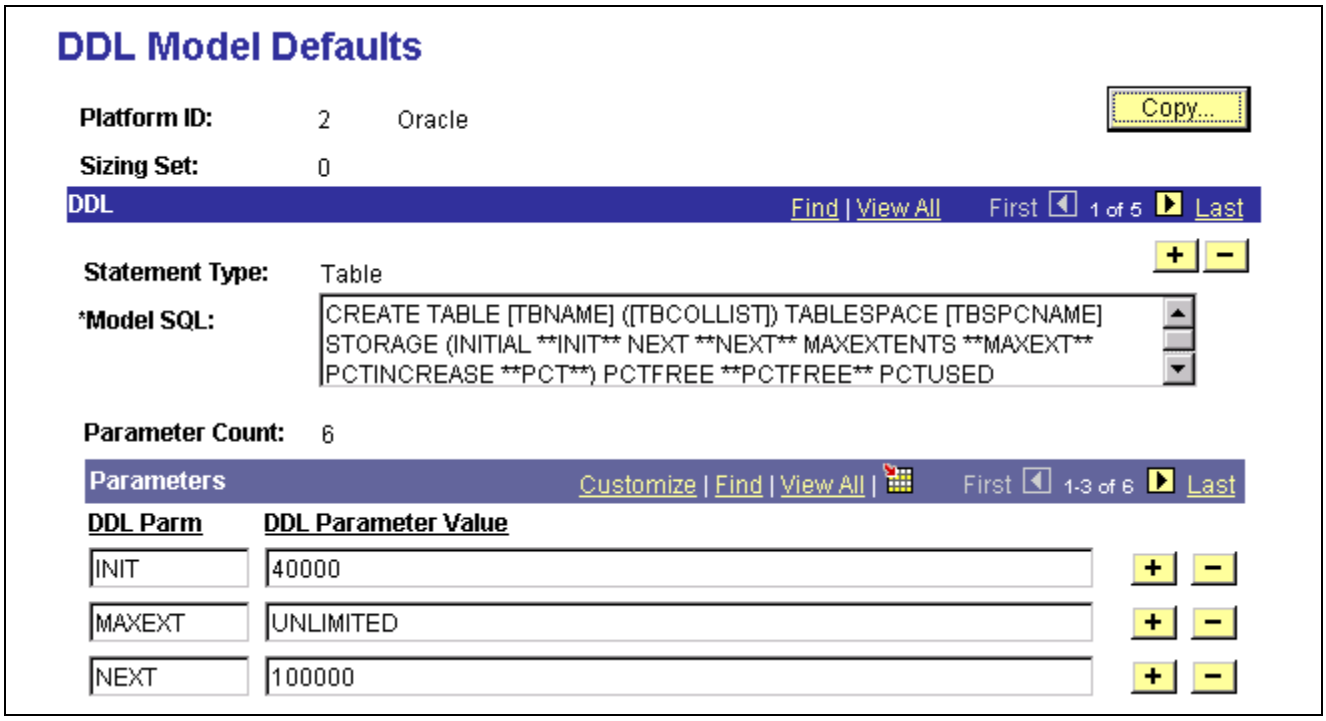
Tablespace DDL Page

This page enables you to view and override DDL parameters if needed. View the default DDL in the Default Tablespace DDL list. You override specific parameters, if needed, in the Override Tablespace DDL list. Enter the parameter that you want to override in the Parameter Name column, and enter the override value in the Override column.

DDL Model Defaults

Select PeopleTools, Utilities, Administration, DDL Model Defaults to access the DDL Model Defaults page.

This page is used to view and edit the DDL for creating tablespaces, indexes and tables. Any changes that you make here are global.



DDL Model Defaults page

Platform ID Identify the type of platform that you are running on.

Sizing Set Specify multiple Sizing Sets if needed. Sizing Sets are a way to maintain multiple versions of the DDL Model statements for a particular database platform. For example, you could have one sizing set to be used during a development phase, when tables only have test data, and you could have separate sizing set to be used during production, when tables have much more data.

Copy Copies information from one sizing set to another.

Statement Type Indicates the type of statement that's entered in the Model SQL edit box. Values for this field can be *Table*, *Index*, and *Tablespace*.

Model SQL This field displays the model SQL statements, which you can edit. Valid statements are CREATE TABLE, CREATE INDEX, CREATE TABLESPACE, and a platform-specific statement for updating statistics.

Some platforms have all the statements, some do not. For example, DB2 UDB has all four statements. SQL Server has only CREATE TABLE and CREATE INDEX.

Parameter Count The Parameter Count is calculated based on how many nonblank DDL parm rows that you define.

DDL Parm The DDL Parm value is a value that the user can change.

DDL Parameter Value The DDL Parameter value is a value that the user can change. Here you can override the DDL parameter default values with your own for the selected statement type. The statement type that you want to change must be open and have the focus in the PeopleSoft Application Designer.

For example, if you want to change the DDL Parm Values for Indexes, set the statement type to Index, then open the record where the index is located in Application Designer, and then change the DDL Parm Value for the index in the chosen record.

Using the DDL Model Defaults page, you can maintain DDL model statements and default parameters for Data Mover. The options that you select on this page also apply to the build function in PeopleSoft Application Designer.

Using this utility, you can:

- Scroll through all the statement types and platforms that are defined in the PSDDLMODEL table.
- Change DDL model statements.
- Add, delete, or change DDL parameters and values.

The Platform IDs are as follows:

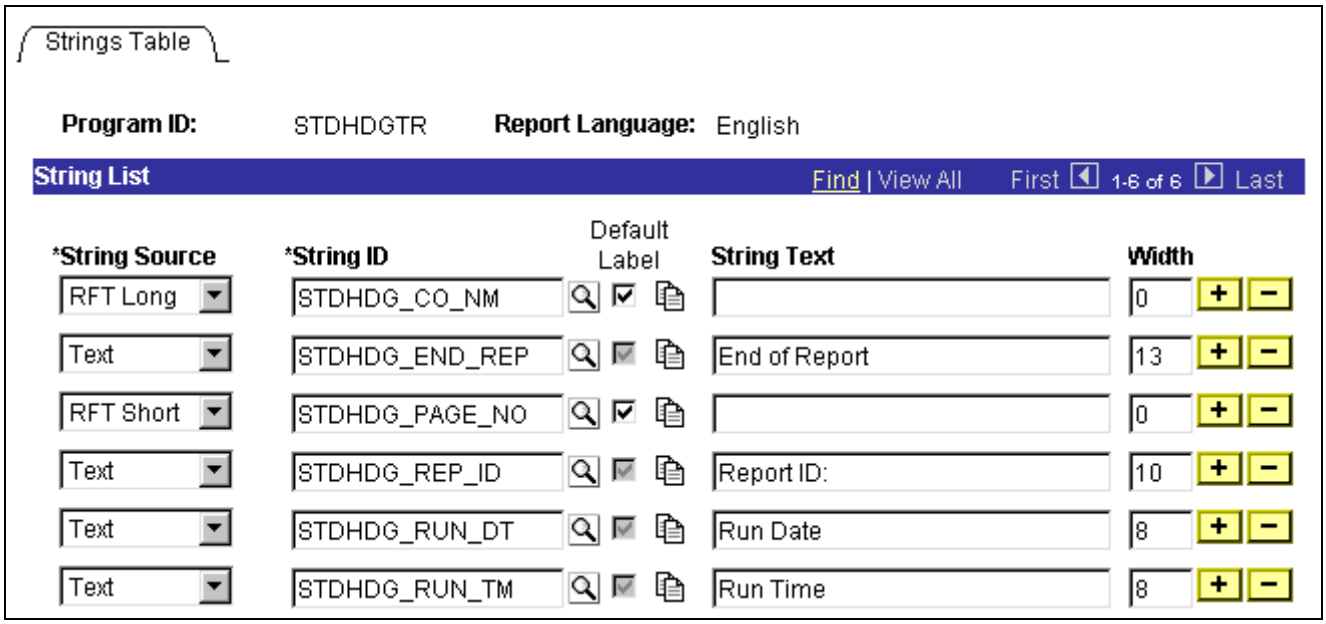
Number	Platform
0	SQLBase (no longer supported).
1	DB2.
2	Oracle.
3	Informix.
4	DB2/Unix.
5	Allbase (no longer supported).
6	Sybase.
7	Microsoft.
8	DB2/400 (no longer supported).

Note. There is no validation performed on the Model SQL statement, the DDL Parm syntax, or the relationship between the statement and the parameters.

Strings Table

Select PeopleTools, Utilities, Administration, Strings Table to access the Strings Table page.

The Strings Table page enables you to customize the column headings in the Structured Query Reports (SQRs):



Strings Table page

String Source Options are:

RFT Long: Select if you want the long description of the field to be displayed in the column heading as set in PeopleSoft Application Designer.

RFT Short: Select if you want the short description of the field as set in the Application Designer to be displayed in the column heading.

Text: Select to enter a custom column heading for the report.

String ID Use the browse button to select the string ID that is to be used for the column heading in the SQR report.

Default Label The default label is enabled if you select the RFT Long or RFT Short string source, otherwise, the check box is disabled.

Remember that fields can have multiple labels. Select the Default Label option to ensure that the default label is used. If you do not use the field's default label, you must select which of the field's labels to use using the label properties button.

String Text Enter the text for the custom column heading, This is the text that is displayed if you set the string source to Text.

Width The default value is the current width of the string that you enter or select. Be sure to update the width based on the actual space that is available on the report layout to avoid limiting a translator to an artificially short length, which is likely to degrade the quality of the translation.

XML Link Function Registry

The XML Link Function Registry is used exclusively in conjunction with the XML Link technology. This utility is documented in the *PeopleSoft Business Interlinks PeopleBook*.

Merchant Integration Utilities

There are two utilities that are related to the Merchant Integration technology that are provided for upgrade support only: Merchant Categories and Merchant Profile.

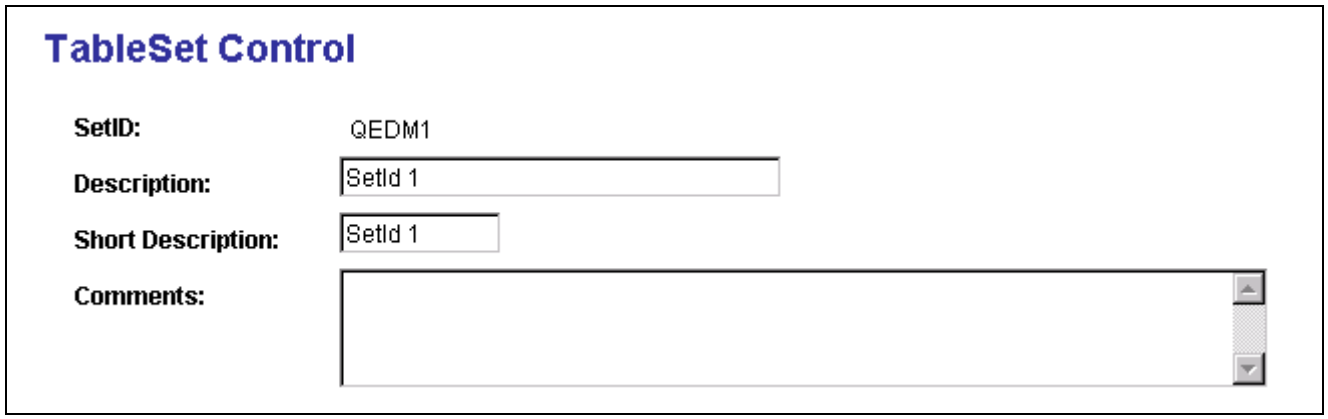
Refer to PeopleSoft documentation from previous releases for information regarding these utilities. These utilities are not intended for any new development purposes.

TableSet IDs

Select PeopleTools, Utilities, Administration, TableSet IDs to access the Tableset Control page.

Use this utility to create Set IDs. Before doing this:

- Add the SETID field (as a key field) to the record definition for that table.
- Define a Set Control Field as the field controlling the assignment of table sets.



TableSet Control page

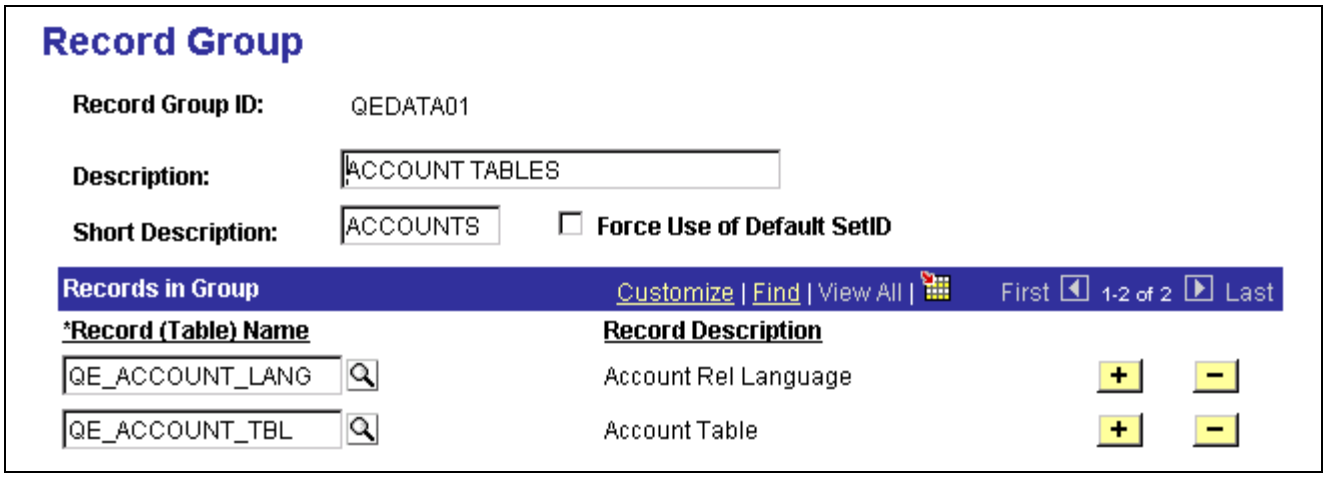
SetID Enter the setID as defined in the record definition.

Description/Comments Add any descriptions and comments that are necessary for identification and internal documentation.

Record Group

Select PeopleTools, Utilities, Administration, Record Group to access the Record Group page.

Used to group record definitions for the tables that you want to share, as well as any dependent record definitions:



Record Group Table page

Description The Record Group ID description should provide enough information to encompass a category of related tables, not just the table that you are specifically sharing.

Short Description Enter a short description.

Force Use of Default SetID This overrides alternate setIDs that are entered so that the default is used.

Record (Table) Name This prompt list comes from a SQL view of record definitions that are defined with that Set Control Field that aren't already associated with a record group.

Record Description Automatically populated when the Record (Table) Name is selected.

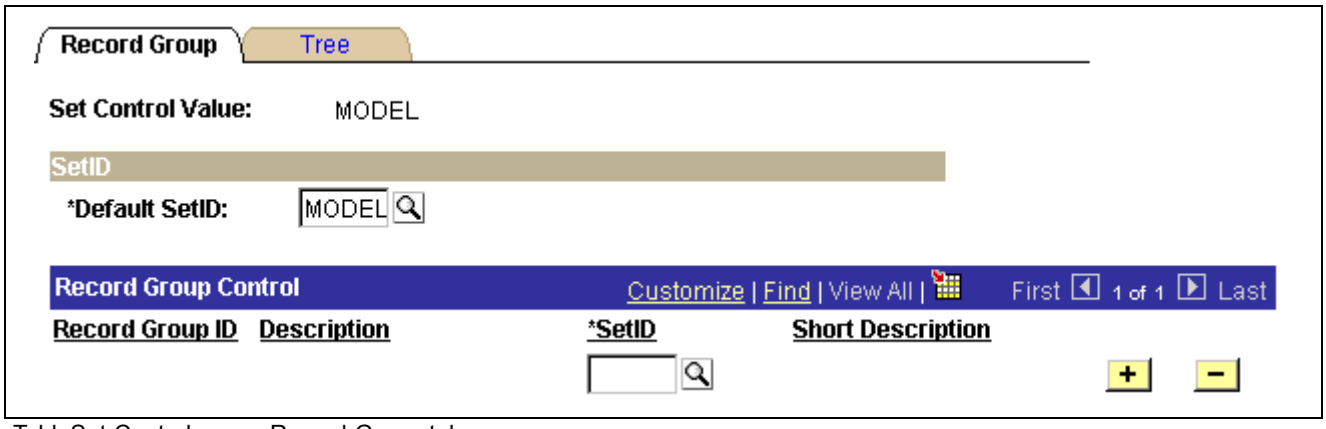
TableSet Control

The following pages are used to control table sets.

Record Group Page

Select PeopleTools, Utilities, Administration, TableSet Control to access the Record Group page.

Used to define which record groups use which table set:



TableSet Control page: Record Group tab

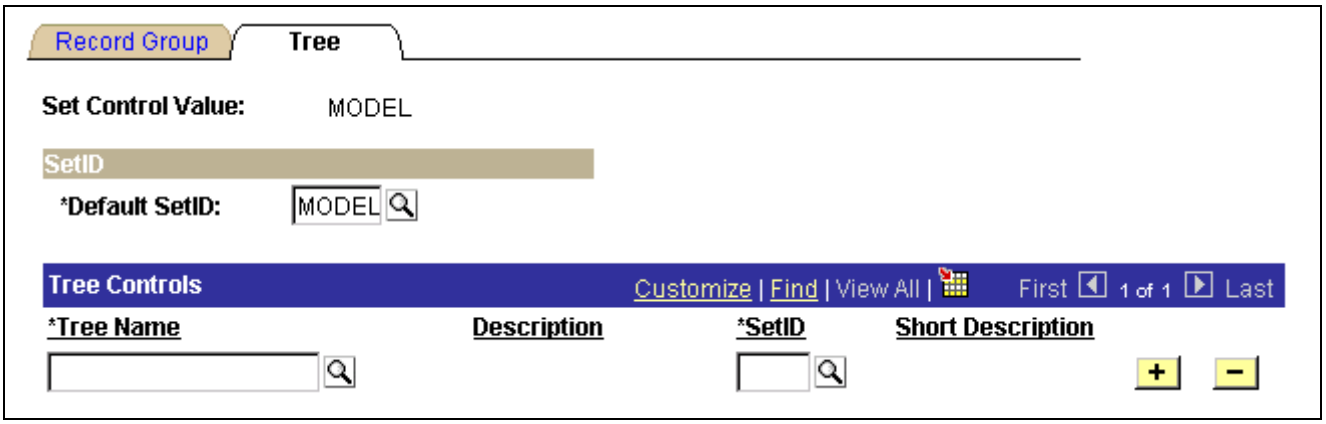
Default SetID This is the setID that the system uses as you add additional record definition groups to be shared within this tableset.

SetID Although this database is set up to share only one accounting-related record group, you may have multiple record groups to which you assign default unique Set IDs.

Tree Page

Select PeopleTools, Utilities, Administration, TableSet Control, Tree to access the Tree page.

Used to share Trees as well as tables and views:



TableSet Control page: Tree tab

Default SetID The Default setID that you assign to this field value automatically appears. If you create another tableset for sharing trees, you can change this value.

Tree Name Use the browse button to select from a list of only the tree definitions that are defined with the same Set Control Field.

SetID Use the browse button to select the appropriate SetID.

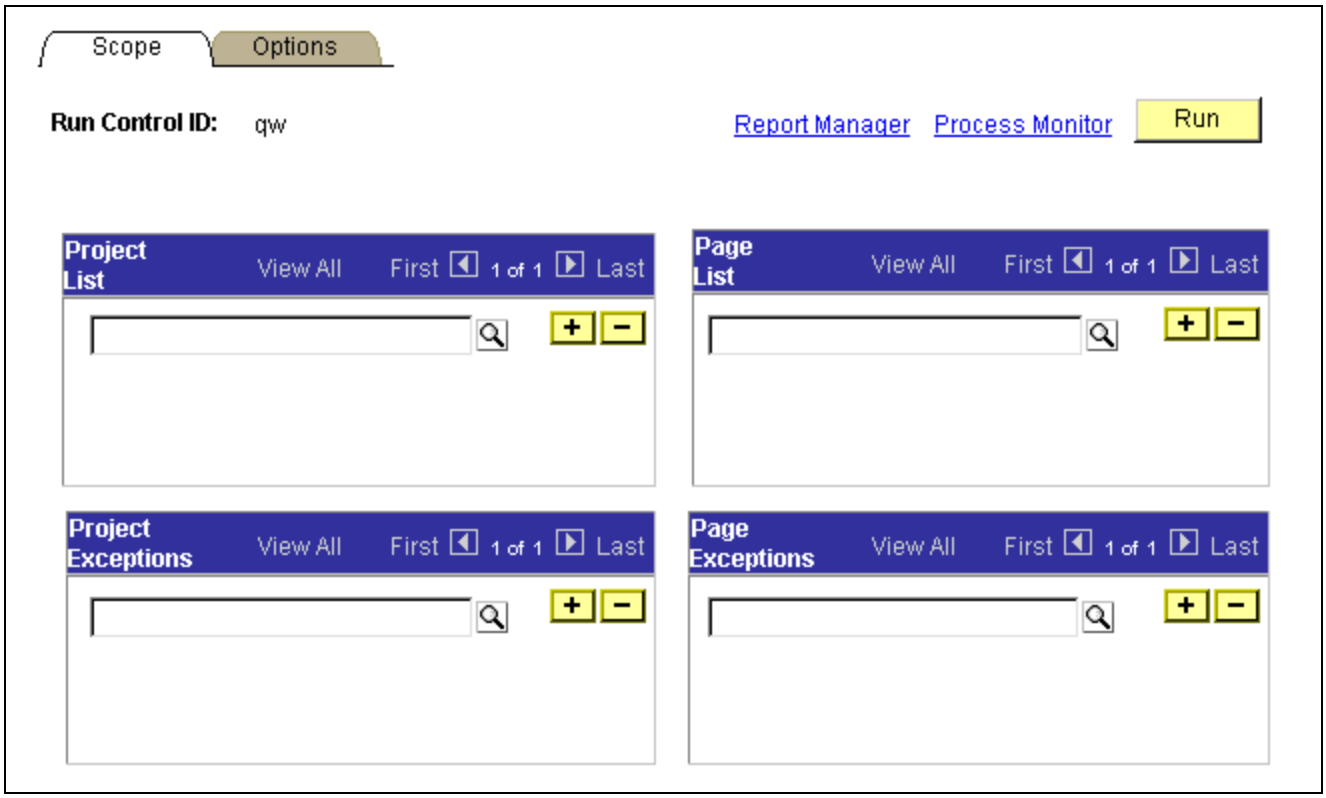
Convert Panels to Pages

The following pages are used to convert panels that are used in previous PeopleSoft Windows applications to pages that are used for browser access.

Scope Page

Select PeopleTools, Utilities, Administration, Convert Panels to Pages to access the Scope page.

This utility helps you update panels that you develop for previous PeopleSoft releases to reflect the pages that are used for the internet architecture.



Convert Panels to Pages page: Scope

Project List Insert projects, containing panels that you want to convert, into this scroll. In addition, if you use the Apply Panel Group Defaults option, any panel group that is contained in projects in this scroll are processed. Note that exceptions may be defined see the task titled, Project Exceptions.

Page List Insert panels that you want to convert to pages into this scroll.

Project Exceptions If you want to ensure that a group of panels or panel groups is never processed for conversion, you can insert them into an application upgrade project and insert the project name in this scroll.

Page Exceptions Panels that are inserted into this scroll are not be processed.

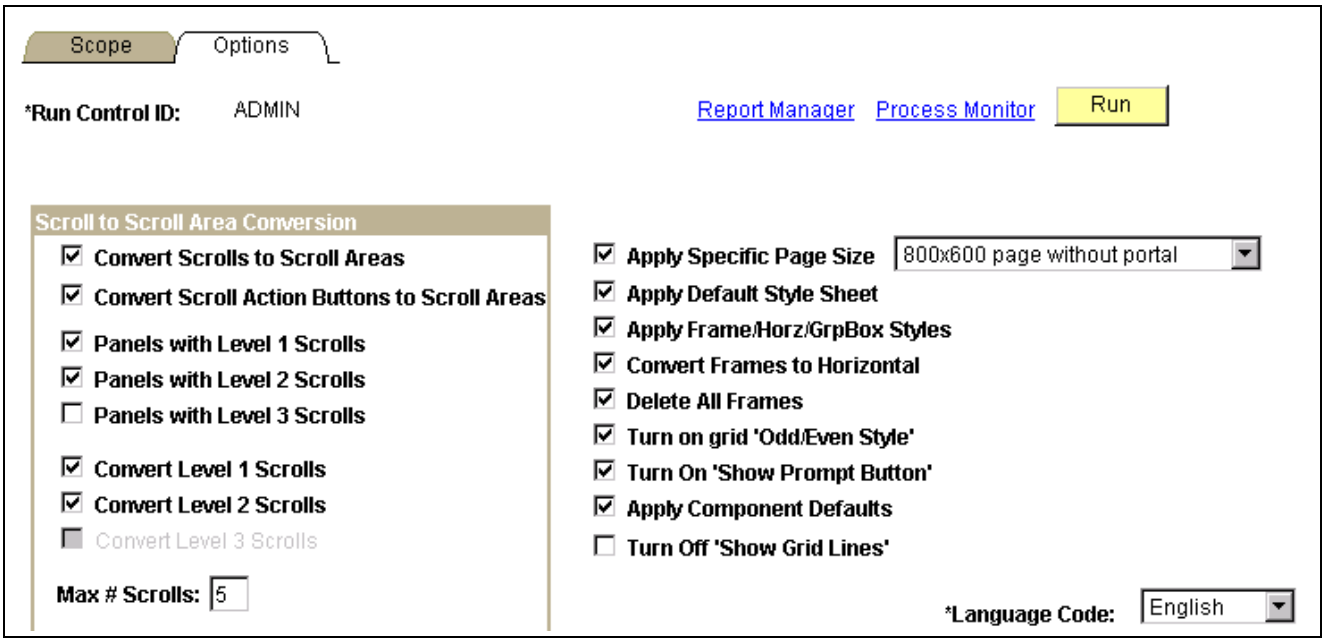
See PeopleSoft ugrade documentation.

documentation for more information.

Options Page

Select PeopleTools, Utilities, Administration, Convert Panels to Pages, Options to access the Options page.

Specify the options for the conversion process:



Convert Panels to Pages: Options page

Convert Scrolls to Scroll Areas If you select this option, scroll-to-scroll area conversions take place for panels with scroll bars. If this is unchecked, no scroll-to-scroll area conversion takes place.

Convert Scroll Action Buttons to Scroll Areas Some scroll bars may exist with scroll action buttons that are already defined. This option determines whether these scrolls should be converted or ignored. If they are converted, the scroll action buttons are removed before the scroll bar is converted to a scroll area.

If you select this option, scrolls with scroll action buttons are converted. If this options is not checked, scrolls with scroll action buttons are ignored.

Panels with Level 1 Scrolls If you select this option, panels with level 1 scrolls are processed for scroll conversion.

Panels with Level 2 Scrolls If you select this option, panels with level 2 scrolls are processed for scroll conversion.

Panels with Level 3 Scrolls If you select this option, panels with level 3 scrolls are processed for scroll conversion.

Convert Level 1 Scrolls If you select this option, level 1 scrolls are converted to scroll areas.

Convert Level 2 Scrolls If you select this option, level 2 scrolls are converted to scroll areas.

Convert Level 3 Scrolls If you select this option, level 3 scrolls are converted to scroll areas.

Max # Scrolls This parameter is a general scroll count limit for scroll conversion processing. For example, if this is set to 5, any panel with more than five scrolls that are not invisible is ignored. This is a simple way of eliminating complex panels from automatic scroll conversion.

Apply Specific Page Size This option is used to define whether a specific size should be assigned to a panel. If you select this option, the panel size that is defined in the drop-down list box is applied to the panel. If this is unchecked, no changes are made to the panel size.

Note. Note. When you select a specific panel size, the panel size is applied to standard panels only (secondary panels and subpanels are not sized automatically).

Apply Default Style Sheet If you select this option, the style sheet that is associated with a panel is updated with a blank value, so that the panel's style sheet appears by default from PSOPTIONS.STYLESHEETNAME ('PSSTYLEDEF').

Apply Frame/Horz/GrpBox Styles If you select this option, the conversion process looks for frames, group boxes, and horizontal rules that have no styles associated with them, and that appear to be associated with a specific scroll area by virtue of their position within a scroll area. It then assigns level-specific styles, based on the occurs level of the scroll area.

Convert Frames to Horizontal Horizontal lines are a new page object for PeopleSoft 8. If you select this option, the conversion process looks for frames on the panel with upper and lower coordinates less than 9 grid units apart. These frames are then converted to horizonal lines.

Delete All Frames If you select this option, the process removes all frames on the converted panel.

Note. If Convert Frames to Horizontal and Delete All Frames are both checked, the conversion from frame to horizontal takes place first, then any remaining frames are deleted.

Turn On Grid 'Odd/Even Style' This applies to grids that are on a panel being converted. If you select this option, the conversion process determines if grids on the panel have their 'Odd/Even Style' turned on. If it is not turned on, the conversion process turns on this option.

Turn On 'Show Prompt Button' This option applies to edit box fields that are not invisible and are not display-only. If you select this option, the conversion process turns on the Show Prompt Button option for edit box fields that have it turned off.

Apply Component Defaults Used to apply standard defaults to component definitions. The defaults that are set are dependent on the Use characteristics of the component. See Application Designer, Component Properties/Use and Component Properties/Internet tabs.

Turn Off 'Show Grid Lines' Turns off the Show Grid Lines option for grids that have it checked on.

Language Code Enables you to convert panels whose language code differs from that in PSOPTIONS. Select a language code from the drop-down list box.

Update Utilities

The Update utilities enable you to keep track of the PeopleSoft updates that you apply to the database.

Updates By Release Label

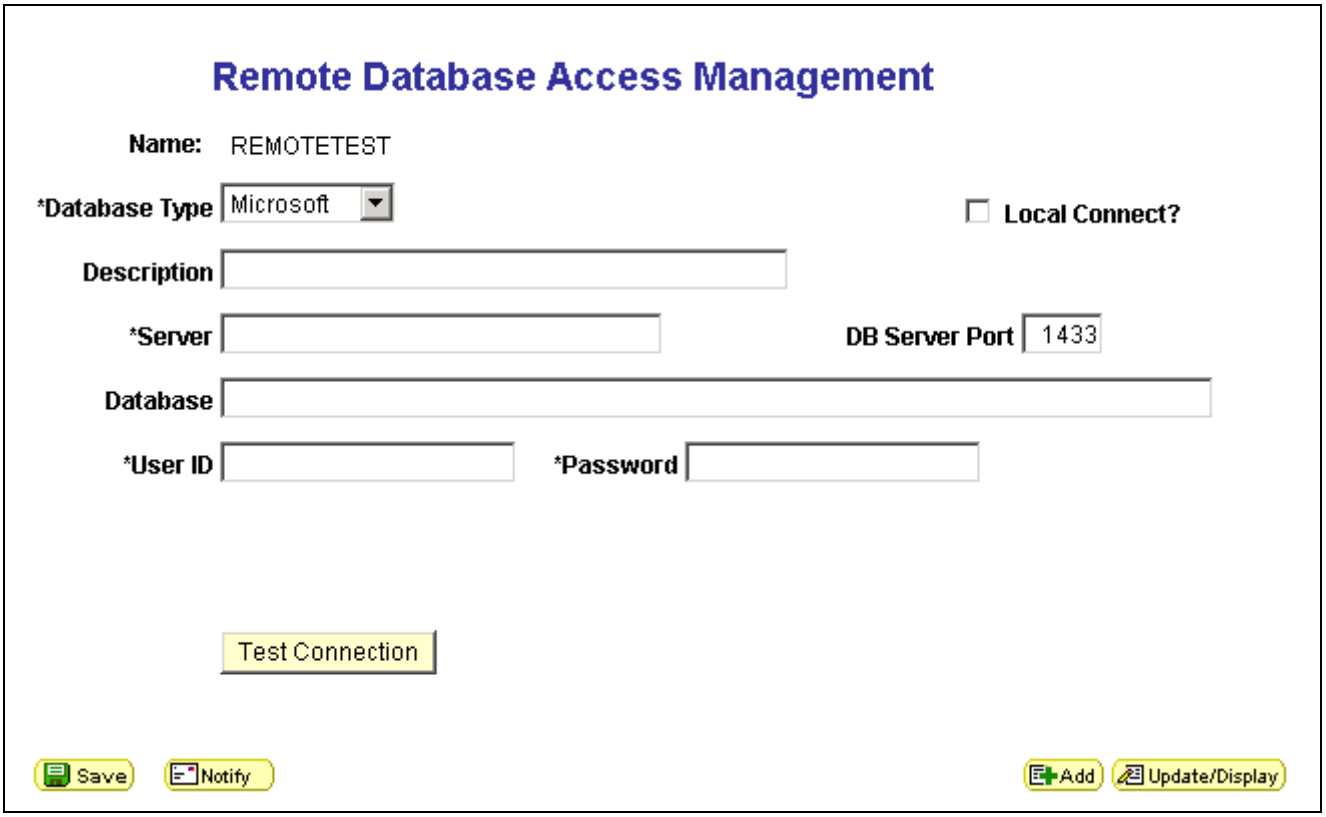
The release label refers to the official release name, such as PeopleTools 8.40.00

Updates By Update ID

The update ID refers to the patch or project name that you apply to the system. The update ID is typically the report ID for a TPRD incident.

Remote Database Connection

Use the Remote Database Connection page to set up remote databases for use with the Remote Data Access (RDA) feature. Select PeopleTools, Utilities, Administration, Remote Database Connection to access the Remote Database Access Management page.



Remote Database Access Management page

Name Enter the name of the remote database connection.

Database Type Available types are Microsoft, DB2 (z/OS), DB2/UNIX, Sybase, Informix, Oracle, and Sybase.

Description Enter a description of the remote database.

Server Enter the server name where the remote database resides.

Database Enter the remote database name.

Local Connect One connection must defined as the Local Connect for the current PeopleSoft instance (the local database). Check this to specify which database is the local.

DB Server Port This value is automatically populated with a default value that is based on the database type. You may need to change this value depending upon the database server configuration.

User ID Enter the user ID that is needed to connect to the remote database.

Password Enter the password that is associated with the user ID.

Test Connection Select this to test the remote database connection.

Connection Type For Oracle database type only. TNS Names or Specific. TNS Names represent a preconfigured file (tnsnames.ora) that consists of previously defined database connection information. Enter Specific if you want to set up a database that does not already have a TNS entry defined.

TNS Entry For Oracle database type only.

Inf Svr Name For Informix database type only.

Security in Remote Databases

To ensure security and limit the risk of unauthorized access to databases, follow these recommendations:

- The remote system's database administrator should create a user with read-only access to the tables that may be accessed by other systems using PeopleSoft's RDA.

  Use this restricted user ID and password in configuring a source RDA node.

- The local system's database administrator should create a user with insert/update access to the RDA destination tables only.

  Use this restricted user ID and password in configuring the target RDA node.

URL Maintenance

Select PeopleTools, Utilities, Administration, URLs to access the URL Maintenance page.

Use the URL Table to store URL addresses and to simplify specifying and updating URLs. URLs that are saved here can be referenced from page controls such as a push button/link. The associated URL can be either an internet or intranet link.



URL Maintenance page

Description Users can search for URLs by description.

URL Enter the entire URL.

Comments This field can be used to make notations and comments and is not displayed elsewhere.

To add a new URL entry in the URL table:

1. Click the Add a New Value link.

A new page appears, prompting you to enter the URL Identifier. Enter the name that you want to use to identify the new URL address.

2. Select Add.
3. Enter the Description, URL, and Comment, if any.
4. Select Save.

   You must save the page before you can add another URL, or update or display existing URL addresses.
5. Select Add to add another URL.

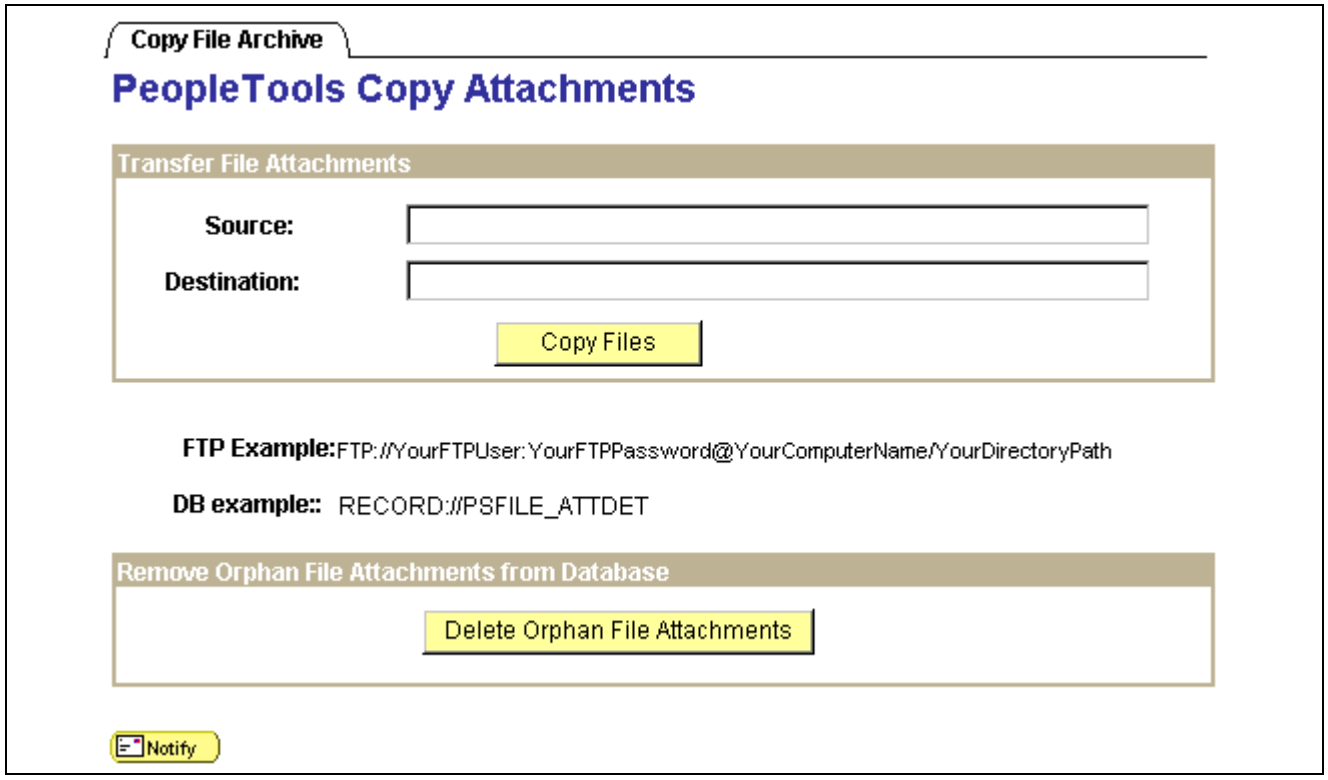
To update or display the URL table:

1. From the URL Maintenance search page, click Search.
2. Select the URL Identifier link that you want to update from the Search Results table.
3. Make changes to the page and save.

Copy File Attachments

Select PeopleTools, Utilities, Administration, Copy File Attachments to access the Copy File Archive page.

Enables you to manage the file attachments that are stored in the database.



Copy File Archive page

Transfer File Attachments

Source When you want to copy the file attachment archive from one location to another, enter the record or directory where the files are currently stored.

Destination	Enter the record or directory where you want to copy the file attachment archive.
Copy	Invokes the PeopleCode function (CopyAttachment) that copies the file attachment archive to another location. For example, you can copy from a file server to a database, and you can copy from a database to a file server.

Remove Orphan File Attachments from Database

The accumulation of file attachments can consume a significant chunk of disk space. On a regular basis, you should make sure that lingering, or orphaned, file attachments are deleted from the database. Click the Delete Orphan File Attachments button to complete this task. This button invokes the CleanAttachments PeopleCode function.

Query Monitor

The Query Monitor is used to track the queries that users execute in the system. It enables you perform such tasks as identify queries that need to have logging turned off or need to be tuned.

See Also

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Query, "Query Administration"

Sync ID Utilities

The Sync ID Utilities are used exclusively with PeopleSoft Mobile Applications technology.

See Also

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Mobile Agent, "Understanding PeopleSoft Mobile Agent," Data Synchronization

Gather Utility

The Gather utility facilitates communications between PeopleSoft and the customer on technical questions or issues. The Global Support Center (GSC) directs the customer to the Gather Utilty when problems arise. Customers can also use a self-service website to run this utility and send in relevant information about their problems or issues.

Using a simple command line interface, the Gather utility is a small Java application that can run on any platform to collect various files from the following environments:

- Application Server.
- Web Server.
- Any additional files that the user chooses (SQL Trace files, PeopleCode Trace Files, etc).

  The collected files are placed in a single jar file with psft.jar as the default name, in the temp directory. Subsequently, these files are sent to PeopleSoft.

Note. For this utility to work, Java must be installed on the target machine. Specifically, JRE 1.4.x and above must be used to take advantage of the Jar Utility Class.

Getting Started

The following files reside in the starting directory:

- Gather.class: The main Java class file
- Helper.class: This class file is called by Gather.class
- Runnit.bat: A MS-DOS batch file that is used by Windows users.

  UNIX users have to run the Gather utility manually.
- Vars.sh: a UNIX shell script.

  Gather calls this automatically if the UNIX operating system is detected.

Windows Users

The following steps are used for Windows (98/98SE/ME/NT/2K):

1. Make sure that you have the PS_HOME environment variable set.

   This saves the user from having to type it in.
2. Go to *PS_HOME*\utility.
3. Type `runnit`.
4. Follow the directions that are on the screen.

UNIX Users

Use the following steps for UNIX:

1. At a command prompt, run the following command where PeopleSoft is installed:

   ```
   ../psconfig.sh
   ```
2. Go to the *PS_HOME*/utility directory.
3. Change permissions for all files:

   ```
   chmod 777 *.*
   ```
4. Enter the following to start the utility:

   ```
   java -cp .:$CLASSPATH Gather
   ```

   Note. UNIX is case-sensitive – Gather is spelled with a capital G.
5. Follow the instructions that are on the screen.

Environmental Data

On Windows, both the set and netstat commands are invoked with the results copied to a file that is collected. On UNIX, the same thing is done with the env command.

Application Server Data

The following files are collected from the Application Server:

- PSAPPSRV.CFG
- PSAPPSRV.UBB
- LOGS/*.*—this usually includes all app serv/tuxedo logs, dump, and replay files.

  This includes all subdirectories under LOGS (8.18).

Web Server Data

The following files are collected from the Web Server:

WebLogic	WebSphere
configuration.properties	configuration.properties
integrationGateway.properties	application.xml
application.xml	ibm-application-bnd.xmi
cookierules.xml	ibm-application-ext.xmi
browserprops.xml	ibm-web-bnd.xmi (for PORTAL, PSIGW, PSOL, PSINTERLINKS, PSEMHUB)
config.xml	ibm-web-ext.xmi (for PORTAL, PSIGW, PSOL, PSINTERLINKS, PSEMHUB)
weblogic.xml (for PORTAL, PSIGW, PSOL, PSINTERLINKS, PSEMHUB)	web.xml (for PORTAL, PSIGW, PSOL, PSINTERLINKS, PSEMHUB)
web.xml (for PORTAL, PSIGW, PSOL, PSINTERLINKS, PSEMHUB)	integrationGateway.properties
gatewayUserProfile.xml	gatewayUserProfile.xml
setEnv.cmd	server.xml
startPIA.cmd	startServer.log
setEnv.sh	stopServer.log
startPIA.sh	SystemErr.log
PIA_access.log	SystemOut.log
PIA_weblogic.log	cookierules.xml
peoplesoft-domain.log	browserprops.xml
PIA_stderr.log	config.xml
PIA_stdout.log	config_prop
emf.log	tools_prop
config_prop	options_prop
tools_prop	emf.log
options_prop	

Additional Files

There is always a need to include files that are not on the above list. These can include PeopleCode Trace files, SQL Trace files, SQL output, and so forth. The command line interface allows you to specify any file that you want to be included in the jar file.

Using Audit Utilities

This section covers the utilities that are used for auditing the system's integrity.

This section discusses how to:

- Use the Record Cross Reference component.
- Perform a system audit.
- Perform database level auditing.

Using the Record Cross Reference Component

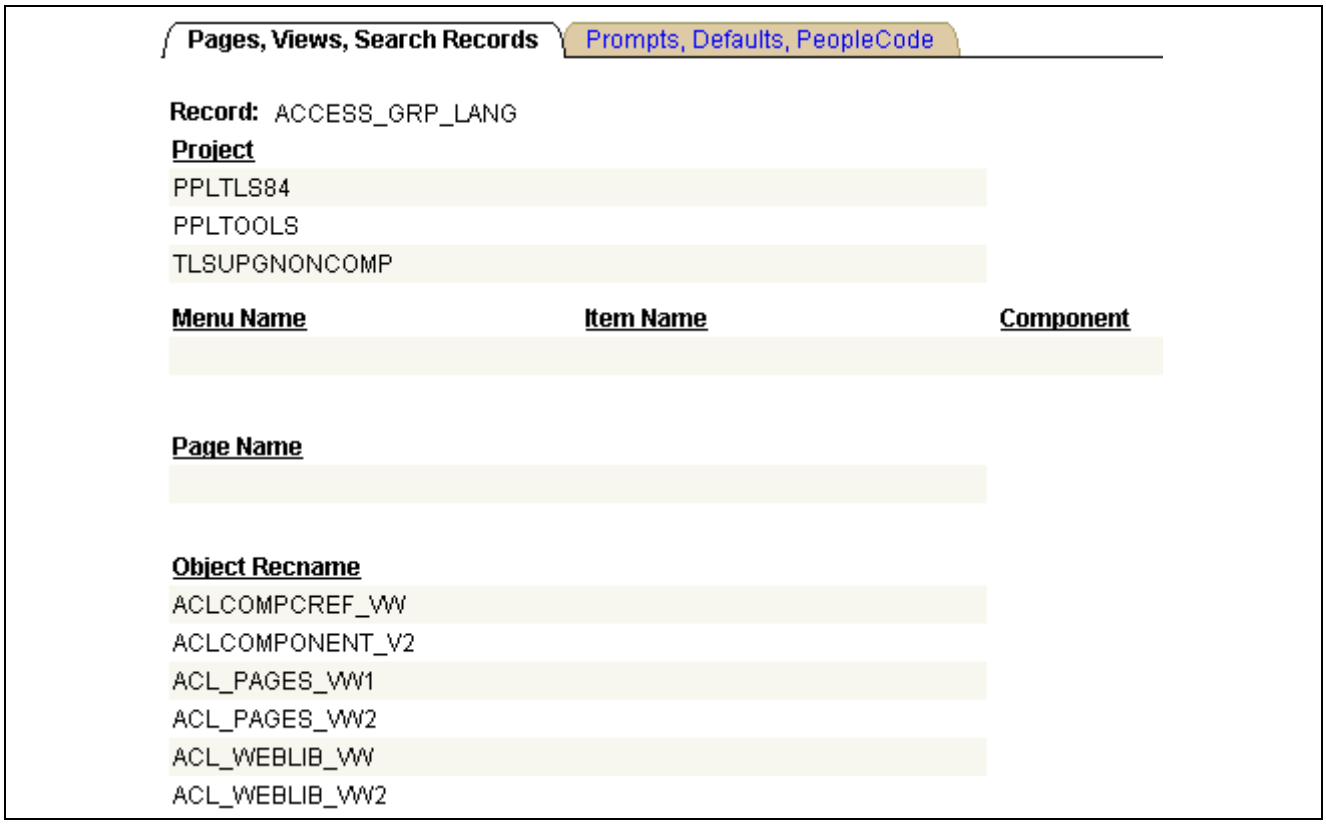
Select PeopleTools, Utilities, Audit, Record Cross Reference.

You use the Record Cross Reference component (XREF_PANEL_01) to view where a record is used throughout the application. There are two pages in this page group:

- Pages, Views, Search Records.
- Prompts, Defaults, PeopleCode.

Pages, Views, Search Records

This is a read-only page that shows which Projects, Menus, Pages, and Objects reference a particular record:



Record Cross Reference: Pages, View, and Search Records page

Prompts, Defaults, PeopleCode

On the Prompts, Defaults, PeopleCode page, the group boxes list the components that refer to the record.

Pages, Views, Search Records | Prompts, Defaults, PeopleCode

Record: ABSENCE_HIST

Used as an Edit Table on: View All First 1 of 1 Last

	Base Record	Field Name
1		

Used as a Default Table in: View All First 1 of 1 Last

	Base Record	Field Name
1		

PeopleCode with Fields from this Record View All First 1-13 of 13 Last

	PeopleCode Reference Name	PeopleCode Fieldname	PeopleCode Recname	PeopleCode Type
1	BEGIN_DT	BEGIN_DT	ABSENCE_HIST	FieldChange
2	BEGIN_DT	DURATION_DAYS	ABSENCE_HIST	SaveEdit
3	BEGIN_DT	RETURN_DT	ABSENCE_HIST	FieldChange
4	BEGIN_DT	RETURN_DT	ABSENCE_HIST	SaveEdit
5	BEGIN_DT	DAY_OF_WEEK	DERIVED_HR	FieldDefault
6	DURATION_DAYS	BEGIN_DT	ABSENCE_HIST	FieldChange
7	DURATION_DAYS	DURATION_DAYS	ABSENCE_HIST	SaveEdit
8	DURATION_DAYS	RETURN_DT	ABSENCE_HIST	FieldChange
9	DURATION_HOURS	DURATION_DAYS	ABSENCE_HIST	SaveEdit
10	RETURN_DT	BEGIN_DT	ABSENCE_HIST	FieldChange
11	RETURN_DT	DURATION_DAYS	ABSENCE_HIST	SaveEdit
12	RETURN_DT	RETURN_DT	ABSENCE_HIST	FieldChange
13	RETURN_DT	RETURN_DT	ABSENCE_HIST	SaveEdit

PeopleCode referring to this View All First 1 of 1 Last

	PeopleCode Recname	PeopleCode Fieldname	PeopleCode Type
1			

Record Cross Reference: Prompts, Defaults, PeopleCode

Used as an Edit Table on Lists pages that use the record for those purposes.

Used as a Default Table in Lists pages that use the record for those purposes.

PeopleCode with Fields from this Record Shows where fields from this record are used in PeopleCode.

PeopleCode referring to this Shows all PeopleCode that references this record.

Performing a System Audit

The System Audit (SYSAUDIT) utility is extensively documented in the Data Management PeopleBook.

See Also

Enterprise PeopleTools 8.46 PeopleBook: Data Management, "Ensuring Data Integrity," Running SYSAUDIT

Performing Database Level Auditing

This utility is used to support database level auditing features, and is extensively documented in the Data Management PeopleBook.

See Also

Enterprise PeopleTools 8.46 PeopleBook: Data Management, "Employing Database Level Auditing"

Using Debug Utilities

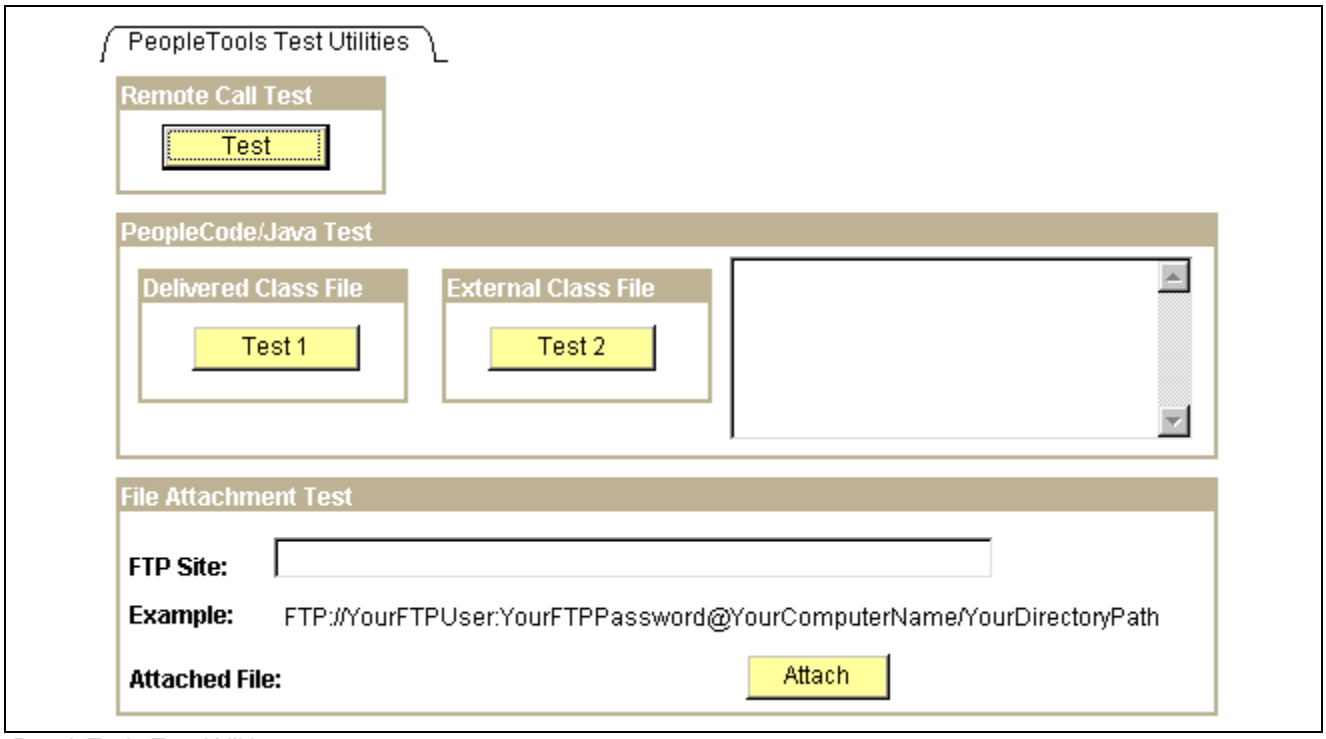
This section discusses how to:

- Use the PeopleTools Test Utilities page.
- Use the Trace PeopleCode utility.
- Use the Trace SQL utility.

Note. The Trace page is no longer actively used or maintained.

Using the PeopleTools Test Utilities Page

Select PeopleTools, Utilities, Debug, PeopleTools Test Utilities to access the PeopleTools Test Utilities page:



PeopleTools Test Utilities page

Remote Call Test You use the Remote Call Test button to test the Remote Call configuration.

Delivered Class File	The Delivered Class File button tests Java PeopleCode integration. It tests to see that Java is being executed correctly through PeopleCode. The Delivered Class File button tests a Java class that is shipped with PeopleSoft 8.
External Class File	The External Class File button tests Java PeopleCode integration. The External Class File button tests a Java class that is created similar to that, which a customer may wish to create.
FTP Site	Enter the full path and password for the test file. For example: FTP://YourFTPUser:YourFTPPassword@YourComputerName/YourDirectory/Path
Attached File	Click this button to attach the file whose path you indicate in the FTP Site.

Using the Trace PeopleCode Utility

The Trace PeopleCode utility is discussed elsewhere in this PeopleBook.

See Also

Chapter 11, “Configuring Trace and Debug Settings,” Setting Up the PeopleCode Debugger, page 257

Chapter 11, “Configuring Trace and Debug Settings,” Configuring PeopleCode Trace, page 260

Using the Trace SQL Utility

The Trace SQL utility is discussed elsewhere in this PeopleBook.

See Also

Chapter 11, “Configuring Trace and Debug Settings,” Configuring SQL Trace, page 261

Using International Utilities

The following sections cover the utilities that you use in globalization efforts.

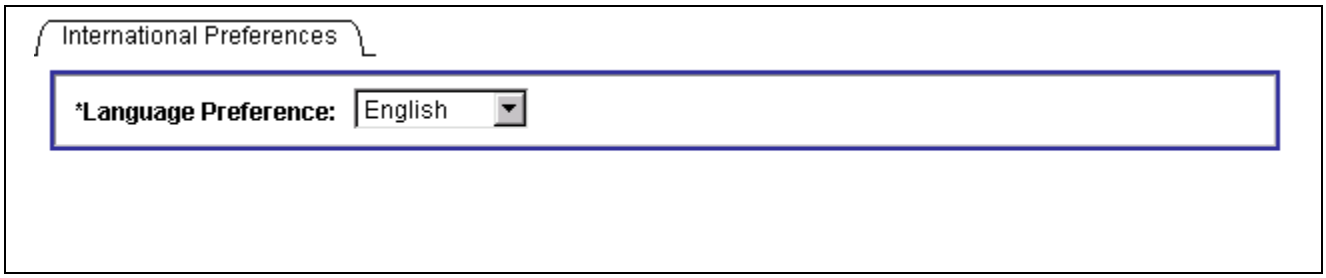
This section discusses how to:

- Set international preferences.
- Set process field size.
- Administer time zones.
- Manage languages.

Setting International Preferences

Select PeopleTools, Utilities, International, Preferences.

Used to override the language that you select when you sign in to the database.



International Preferences page

Language Preference Use the International Preferences page to temporarily change the session's language preference that was specified during signon. This change lasts until you exit the PeopleSoft session or change the language preference again. Only languages that are enabled on the Languages page are available for selection.

Setting Process Field Size

Select PeopleTools, Utilities, International, Process Field Size.

If you process currency values that require large numbers, such as Italian lira, that require fields longer than those that are included in the standard application, you can use the International Field Size page to expand amount fields throughout the application.

After you create or select a run control ID, set the appropriate lengths for a list of fields, then click the Run button to launch the batch program that performs the field size changes.

Field Name Use the Browse button to select the field name.

Current Field Size This is a read-only field indicating the current field size as stored in PSDBFIELDS.

Field Size - International Enter the field size to expand (or contract) the field size for foreign fields.

Administering Time Zones

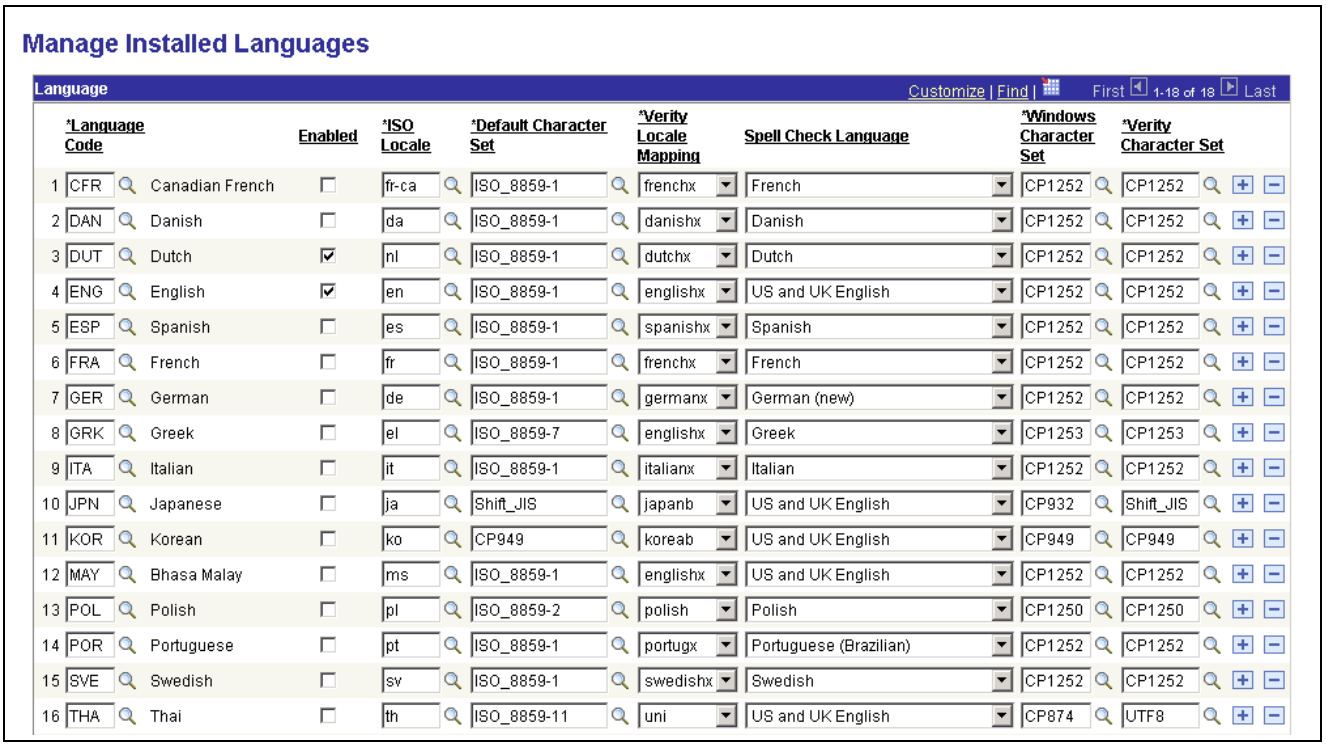
This utility is extensively documented in thePeopleTools Global Technology PeopleBook.

See Also

Enterprise PeopleTools 8.46 PeopleBook: Global Technology, "Setting and Maintaining Time Zones"

Managing Languages

Select PeopleTools, Utilities, International, Languages to access the Manage Installed Languages page.



Manage Installed Languages

Language — Customize | Find | First 1-18 of 18 Last

	*Language Code		Enabled	*ISO Locale	*Default Character Set	*Verity Locale Mapping	Spell Check Language	*Windows Character Set	*Verity Character Set
1	CFR	Canadian French	☐	fr-ca	ISO_8859-1	frenchx	French	CP1252	CP1252
2	DAN	Danish	☐	da	ISO_8859-1	danishx	Danish	CP1252	CP1252
3	DUT	Dutch	☑	nl	ISO_8859-1	dutchx	Dutch	CP1252	CP1252
4	ENG	English	☑	en	ISO_8859-1	englishx	US and UK English	CP1252	CP1252
5	ESP	Spanish	☐	es	ISO_8859-1	spanishx	Spanish	CP1252	CP1252
6	FRA	French	☐	fr	ISO_8859-1	frenchx	French	CP1252	CP1252
7	GER	German	☐	de	ISO_8859-1	germanx	German (new)	CP1252	CP1252
8	GRK	Greek	☐	el	ISO_8859-7	englishx	Greek	CP1253	CP1253
9	ITA	Italian	☐	it	ISO_8859-1	italianx	Italian	CP1252	CP1252
10	JPN	Japanese	☐	ja	Shift_JIS	japanb	US and UK English	CP932	Shift_JIS
11	KOR	Korean	☐	ko	CP949	koreab	US and UK English	CP949	CP949
12	MAY	Bhasa Malay	☐	ms	ISO_8859-1	englishx	US and UK English	CP1252	CP1252
13	POL	Polish	☐	pl	ISO_8859-2	polish	Polish	CP1250	CP1250
14	POR	Portuguese	☐	pt	ISO_8859-1	portugx	Portuguese (Brazilian)	CP1252	CP1252
15	SVE	Swedish	☐	sv	ISO_8859-1	swedishx	Swedish	CP1252	CP1252
16	THA	Thai	☐	th	ISO_8859-11	uni	US and UK English	CP874	UTF8

Manage Installed Languages page

Use this page as a central utility to manage language information for the currently enabled languages.

Language Code Use the search prompt to select the PeopleSoft language code from the PSXLATITEM table. The language description appears to the right of the code field.

Enabled When you select this check box, PeopleSoft Internet Architecture enables you to log in with the language.

ISO Locale Use the search prompt to select the ISO locale code from the PSLOCALEDEFN table. Consists of an ISO 639 language code, optionally followed by an ISO 3166 country code.

Default Character Set Use the search prompt to select the character set from the PSCHARSETS table. Determines the default encoding for input and output files.

Verity Locale Mapping Select the Verity locale code from the PSVERITYLOCALE table. Determines the locale to use for building search collections and searching data.

Spell Check Language Select the spell check language from the PSXLATITEM table. This enables you to select the language of the spell check dictionary that is associated with a given language code.

Windows Character Set Select the Microsoft codepage that is associated with the given language. This defines the codepage to use with certain Microsoft applications.

Verity Character Set Select the character set that the Verity engine uses for its internal encoding in the given language. You should not modify the value in this field under normal circumstances.

Using Optimization Utilities

The Optimization utilities are documented extensively in the Optimization Framework PeopleBook.

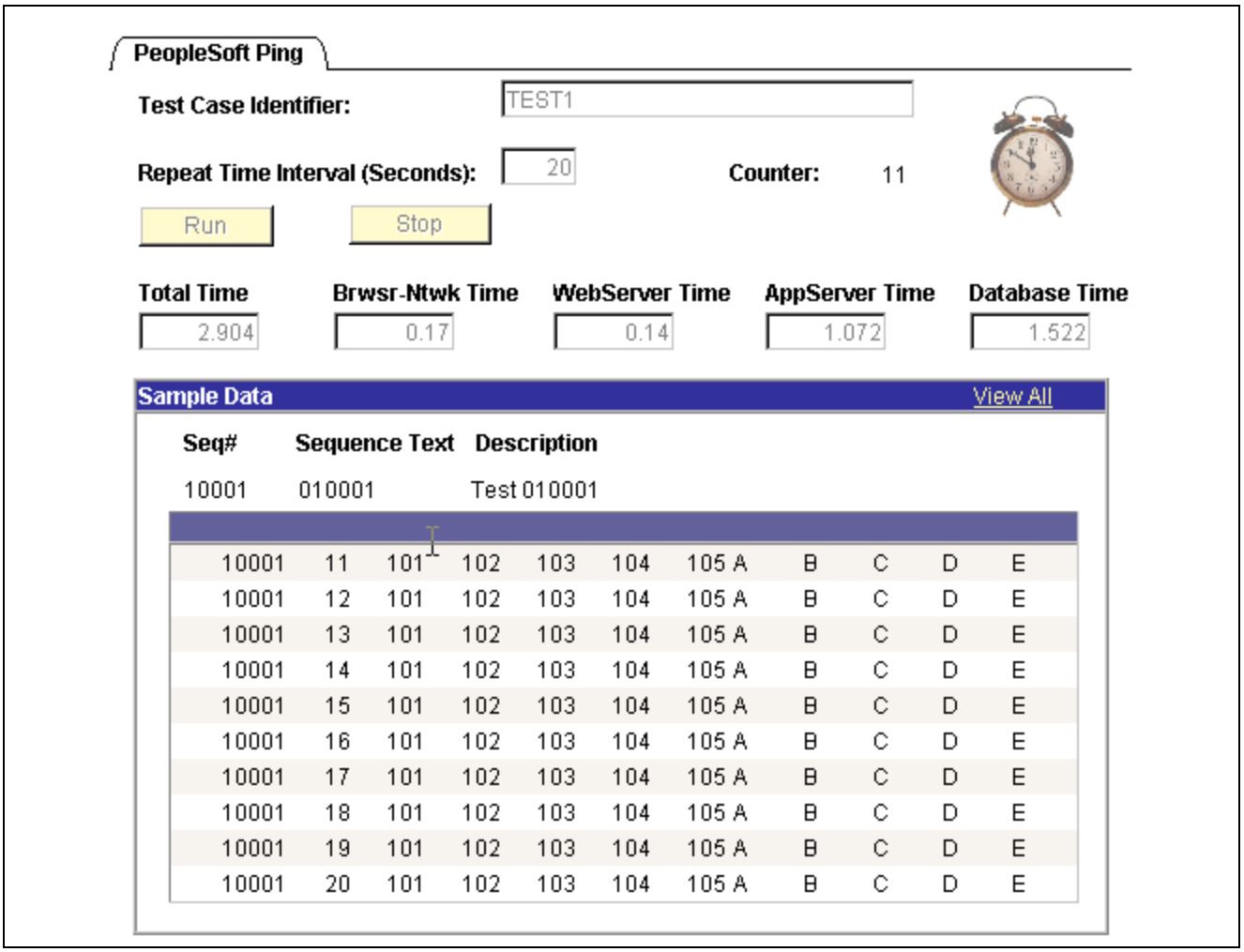
See Also

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Optimization Framework, "Designing Analytic Type Definitions"

Using PeopleSoft Ping

The PeopleSoft Ping feature collects timestamps by sending a specific page to different tiers of the PeopleSoft system, starting at the browser, then going to the web server, the application server, the database and back. The timestamps that are collected are total time elapsed for the round trip, and arrival and departure time at each of the tiers.

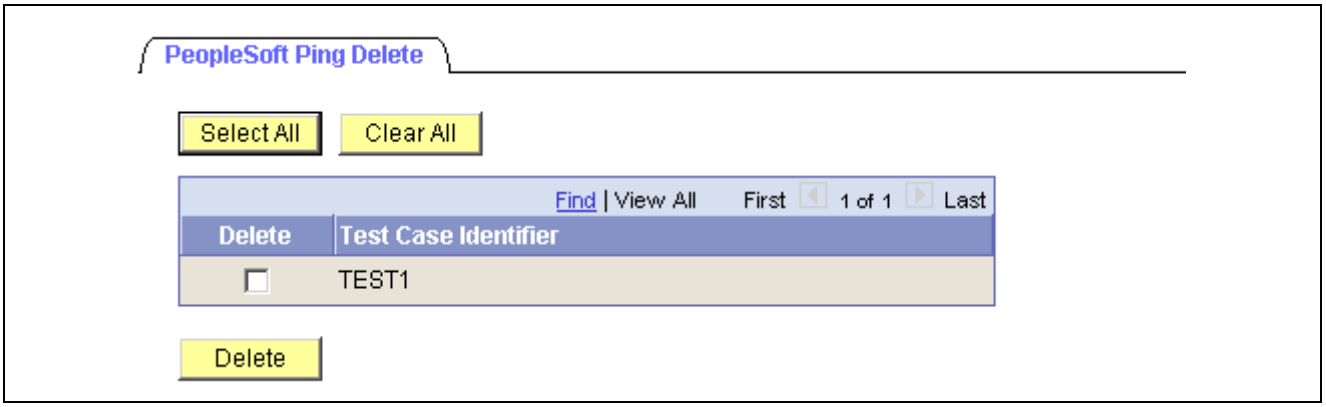
To use the PeopleSoft Ping feature, select PeopleTools, Utilities, PeopleSoft Ping. Enter a new or existing Test Case Identifier to show the following page:



PeopleSoft Ping Page

After you specify a test case identifier, enter a value for Repeat Time Interval.To avoid creating unnecessary traffic and overhead to the PeopleSoft system, set the Repeat Time Interval to a relatively high value, such as 600 to 1800 seconds, during normal operations. You may need to increase the Session Timeout value accordingly.

To delete a ping page test case, select PeopleSoft Ping Delete. Select the appropriate check box:

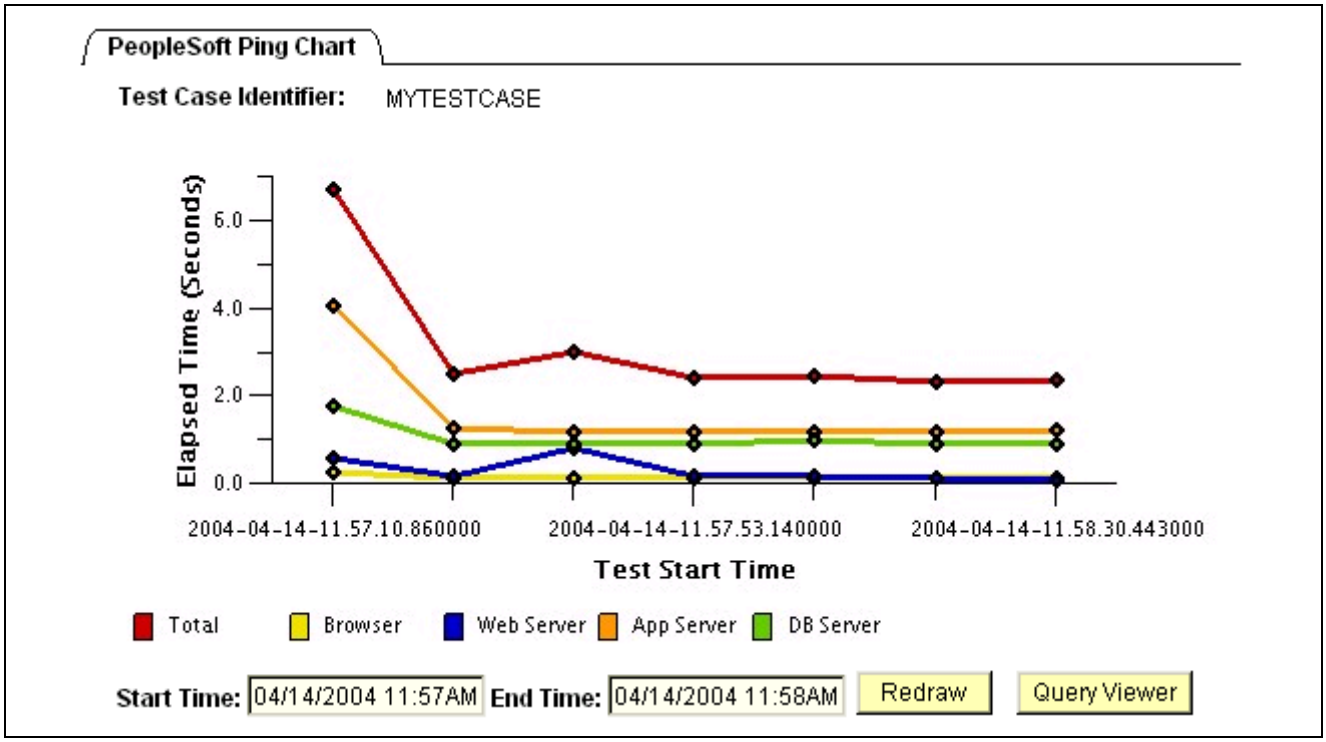


PeopleSoft Ping Delete page

The Delete page lists the test case identifiers. Select the check box that is next to the desired test case identifiers to delete a test case.

PeopleSoft Ping Chart

PeopleSoft Ping includes a charting utility to zoom in to a specific time interval from the ping test. To view a graphic chart of the ping data, select PeopleSoft Ping Chart:



PeopleSoft Ping Chart page

You can change the displayed time interval to a subset of the full ping test period. Edit the start time and end time values, and click Redraw to refresh the chart display with the new time interval.

Click Query Viewer to query the database for the ping data. A new browser window opens, displaying the ping data for the full test period in a table.

CHAPTER 11

Configuring Trace and Debug Settings

This chapter discusses how to:

- Set up the PeopleCode Debugger.
- Configure PeopleCode trace.
- Configure SQL trace.

See Also

Chapter 4, "Setting Application Server Domain Parameters," Trace Options, page 53

Chapter 9, "Using PeopleSoft Configuration Manager," Specifying Trace Settings, page 191

Setting Up the PeopleCode Debugger

This section discusses how to:

- Debug for a two-tier connection.
- Debug for a three-tier connection.
- Use the PeopleCode Debugger.

Note. PeopleCode debugging is not supported on z/OS.

You can debug the PeopleCode program configurations of a two-tier connection to the database or a three-tier connection to the database.

Note. When you debug PeopleCode with an application server, PeopleSoft Application Designer should be run in three-tier mode. PeopleCode debugging by using a two-tier PSIDE and an application server is not supported on multi-homed (multiple Internet Protocol address) workstations.

Debugging for a Two-Tier Connection

Debugging in two-tier connections involves connecting directly to the database, not through the application server. Use this method to debug two-tier Windows applications.

Note. By default, the port number that the PeopleCode debugger uses is 9500. Unless this port number is being used by another application, you do not need to alter any environment settings, and after you sign on to the database, you are able to debug PeopleCode.

If you need to change the PeopleCode Debugger port settings, complete the following procedure.

To change the default PSDBGSRV listener port number:

1. Open PeopleSoft Configuration Manager.
2. Select the Trace tab.
3. Locate the PeopleCode Debugger section, and make sure that the default value for the Local PSDBGSRV Listener Port is suitable for the system.

   For example, make sure that no other applications are configured to listen on the default port number (9500). If so, you must assign a port number that is not being used.

Note. If you're using a personal firewall, you must configure it to enable data packets to flow through the PSDBGSRV listener port. If you can't configure your firewall appropriately, you must shut it down while performing PeopleCode debugging.

Debugging for a Three-Tier Connection

Use three-tier debugging to debug three-tier Windows applications and PeopleSoft Internet Architecture (PIA) applications. For three-tier debugging, use PSADMIN to ensure that the following items are set:

- The appropriate PSDBGSRV listener port is specified in the PeopleCode Debugger section of PSADMIN.
- At least two PSAPPSRV processes are configured to boot in the domain with the service timeout parameter set to zero.
- Enter y for yes at the Enable PSDBGSRV Server Process prompt at the end of the PSADMIN interface.

Debugging on a Multi-Homed System

If you're debugging on a multi-homed (multiple IP address) system, you must explicitly specify an IP address in the Workstation Listener section of the PSADMIN configuration, rather than %PS_MACH%. The address you specify must be one by which the application server identifies the machine on which you're doing the debugging. This ensures that the workstation listener monitors requests from the correct location.

See Chapter 4, "Setting Application Server Domain Parameters," Workstation Listener Options, page 46.

Setting the PSDBGSRV Listener Port

In the PeopleCode Debugger section of PSADMIN make sure that the value that is assigned to the PSDBGSRV listener port is not already in use by another application or listener on the application server. The default value is 9500. If the default is not acceptable, assign a suitable value to the parameter. If it is acceptable, no changes are required.

For example,

```
Values for config section - PeopleCode Debugger
    PSDBGSRV Listener Port=9500

Do you want to change any values (y/n)? [n]:
```

Consider the following when debugging PeopleCode:

- If multiple application server domains are running on a single, physical machine, each domain needs to use different debugging port numbers.

  Otherwise, there is contention for the PSDBGSRV listener port value. This is the same principle that requires each application server domain on a server to have unique workstation listener port numbers.
- When you are not debugging, turn off (set to 0) the Enable Debugging parameter.

The debugging mode results in an unavoidable amount of overhead, which can degrade performance.

- Regarding performance, do not perform debugging on a production domain.

  Debugging should be performed on a designated testing domain only.

Enabling Multiple PSAPPSRV Server Processes

The minimum requirements for PeopleCode debugging are:

- Two PSAPPSRV server processes are configured to boot in the domain.
- The Service Timeout value in the PSAPPSRV configuration section must be set to 0.

For the debugger to work, it has to run in parallel with the application that it's debugging. Suppose that the domain has only one PSAPPSRV server process running. In this case, the PSAPPSRV can process the requests of only one component at a time, and therefore debugging is not possible. Debugging involves two items, the debugger (PSDBGSRV) and the PSAPPSRV server process that is running the application PeopleCode.

Provided that you have two PSAPPSRV server processes configured; one PSAPPSRV handles the debugger program, while the other handles the application that you're stepping through with the debugger. In this case, the two programs run in parallel, which enables interactive debugging.

The configuration templates that PeopleSoft delivers all have at least two PSAPPSRV processes. However, if you are using a custom template, make sure that you configure the domain to start two PSAPPSRV processes prior to debugging. To do this, in PSADMIN set the Min Instances parameter in the PSAPPSRV section to 2.

You must set the Service Timeout parameter for PSAPPSRV to 0. Disabling service timeouts prevents the application server processes from timing out if you stop at a particular point in the program while debugging.

The following example shows a sample PSAPPSRV section properly configured for debugging PeopleCode:

```
Values for config section - PSAPPSRV
    Min Instances=2
    Max Instances=2
    Service Timeout=0
    Recycle Count=0
    Allowed Consec Service Failures=0
    Max Fetch Size=5000


Do you want to change any values (y/n)? [n]:
```

When configuring the PeopleCode debugger:

- PeopleSoft recommends using the Developer configuration template because this template, by default, provides two PSAPPSRV server processes and has service timeout set to zero.
- PeopleSoft recommends using a simple configuration where you are assured that the server that PeopleSoft Application Designer connects to is the same server that the application you are debugging is running on.

Note. If you do not set the settings for PSAPPSRV correctly (at least two PSAPPSRV processes with the Service Timeout value set to 0), PSADMIN automatically sets these values to comply with the minimum requirements when you enable PeopleCode Debugging (as discussed in the next section).

Requesting a PSDBGSRV Server Process

After you specify the settings by using PSADMIN, the system prompts you with a series of options, such as setting up messaging server processes, enabling BEA JOLT, and so on.

When you're prompted to enable the PSDBGSRV, enter y. Y appears in the Developer template by default.

Using the PeopleCode Debugger

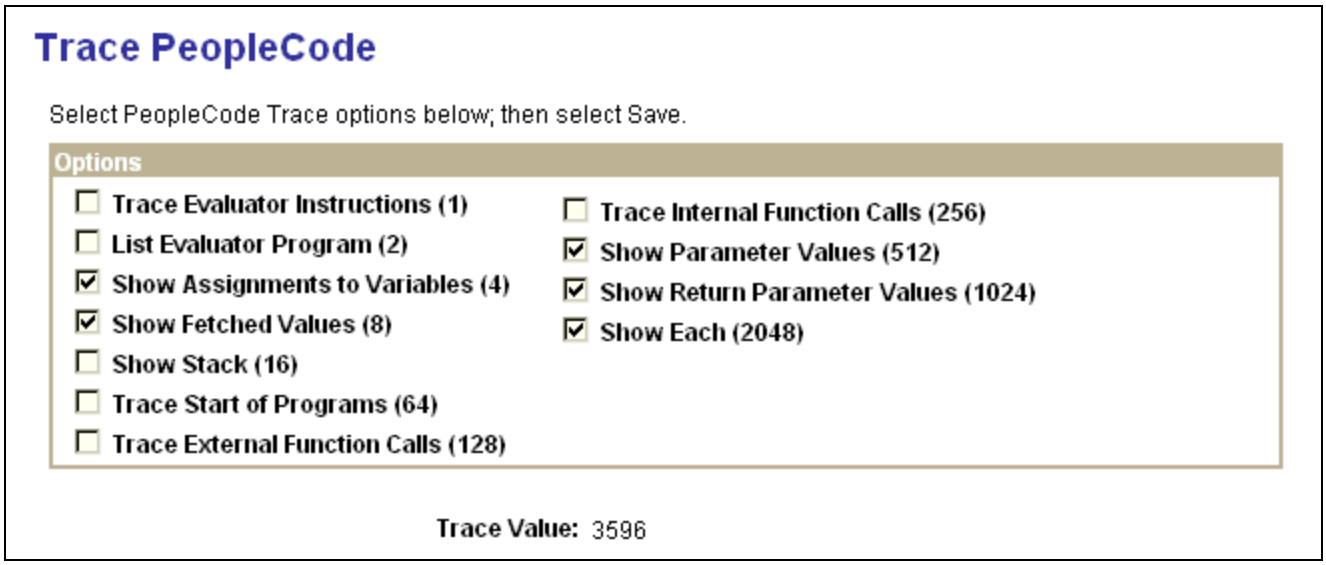
After the system is configured properly, using the PeopleCode debugger is just a matter of signing on to the PeopleSoft system and entering the PeopleCode Debugger mode in PeopleSoft Application Designer.

Note. You must use a unique user ID when you're performing PeopleCode debugging, as opposed to using a shared user ID, such as those that PeopleSoft delivers, for example QEDMO, PS, or VP1. Shared IDs are likely to be used by others that are connecting to the same test database, which can affect debugging.

Configuring PeopleCode Trace

Select PeopleTools, Utilities, Debug, Trace PeopleCode to access the Trace PeopleCode page.

You use this page to change the PeopleCode tracing options while online. This page does not affect trace options that are set in PeopleSoft Configuration Manager. Use Trace PeopleCode to create a file displaying information about PeopleCode programs processed from the time that you start the trace.



Trace PeopleCode page

Trace Evaluator Instructions	Select to show a line-by-line trace of the program
List Evaluator Program	Select to show the code of the PeopleCode program.
Show Assignments to Variables	Select to show variable assignments.
Show Fetched Values	Select to show values that are from PeopleCode Fetch call.
Show Stack	Select to display the PeopleCode evaluator's stack after each PeopleCode (internal) instruction.
Trace Start of Programs	Select to show the starting and ending points of the program.
Trace External Function Calls	Select to show calls to application written functions.

Trace Internal Function Calls	Select to show the calls to PeopleTools built-in function calls.
Show Parameter Values	Select to show function parameter values.
Show Return Parameter Values	Select to show function return parameter values.
Show Each	Select to trace each statement in the program.

Note. The Trace PeopleCode Utility decreases system performance because of the overhead that occurs during the monitoring and recording of all PeopleCode actions.

The check boxes on this page correspond to the options on the Trace tab in Configuration Manager. However, the selections that appear on this page do not necessarily reflect those that are made in Configuration Manager. While the Configuration Manager settings are stored in the Windows registry and used at each signon, the settings in the Utilities page only apply to the current online session, and, once set, they override the Configuration Manager's settings.

The benefit of using this page to control PeopleCode tracing is that you can turn it on and off without having to restart PeopleTools, and without resetting the Configuration Manager settings. Keep in mind, though, your selections are not enabled until you save the page.

To enable/disable PeopleCode tracing while on line

1. Select PeopleTools, Utilities, Debug, Trace PeopleCode.

   The Trace PeopleCode page appears.
2. Select/deselect the desired Options.
3. Save the page.

   If you selected any of the check boxes, the system starts writing to the trace file.

See Also

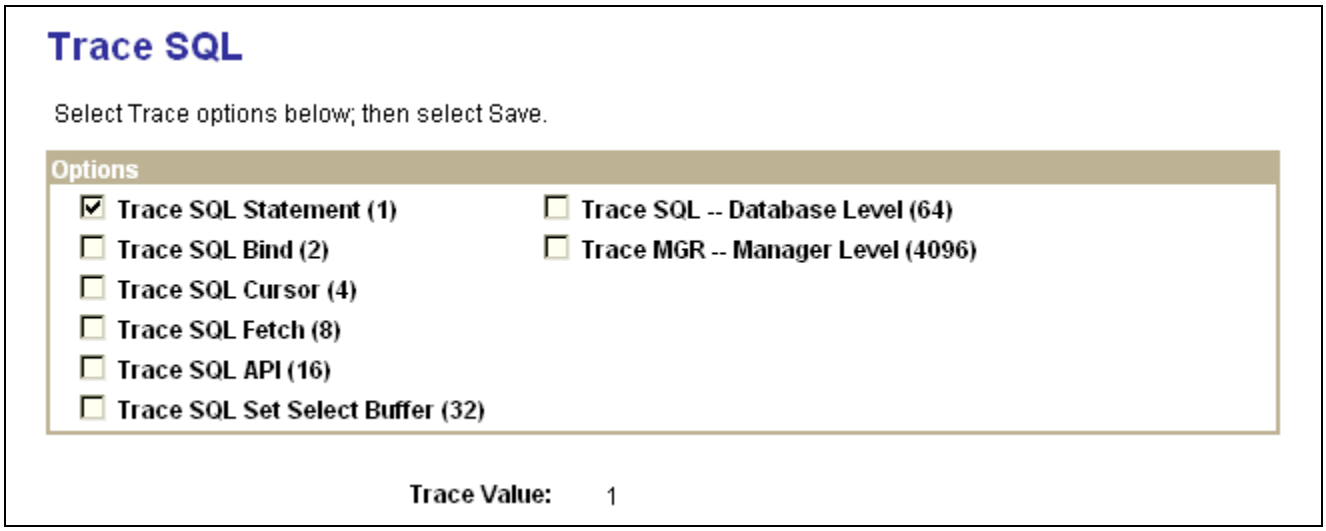
Chapter 9, "Using PeopleSoft Configuration Manager," Specifying Trace Settings, page 191

Enterprise PeopleTools 8.46 PeopleBook: PeopleCode Language Reference, "PeopleCode Built-in Functions," SetTracePC

Configuring SQL Trace

Select PeopleTools, Utilities, Debug, Trace SQL to access the Trace SQL page.

You use this page to change the SQL tracing options while you're online. Your Configuration Manager settings are not affected:



Trace SQL page

Trace SQL Statement	Select to show the SQL statement.
Trace SQL Bind	Select to show bind values for SQL statements that have parameter markers.
Trace SQL Cursor	Select to show connect, disconnect, commit and rollback calls.
Trace SQL Fetch	Select to show fetch call for Select Statement.
Trace SQL API	Select to show other API calls (Execute, Describe, and so on.)
Trace SQL Set Select Buffer	Select to show Binds for Select columns.
Trace SQL -- Database Level	Select to specify low-level tracing at the database API (ODBC, ct-lib, and so on.)
Trace SQL -- Manager Level	Select to show calls for Cache calls.

The check boxes on the Trace SQL page correspond to options on the Trace tab in the Configuration Manager. However, the selections that appear on this page do not necessarily reflect those that are made in the Configuration Manager. The displayed page selections are not enabled until you save the page.

To enable or disable SQL tracing while online:

1. Select or deselect the desired trace options.
2. Save the page.

   If you select any of the check boxes, the system starts writing to the trace file.

See Also

Chapter 9, “Using PeopleSoft Configuration Manager,” Specifying Trace Settings, page 191

Enterprise PeopleTools 8.46 PeopleBook: PeopleCode Language Reference, “PeopleCode Built-in Functions,” SetTraceSQL

CHAPTER 12

Setting Up Jolt Internet Relay

This chapter provides an overview of Jolt Internet Relay and discusses how to:

- Configure JRLY.
- Configure JRAD.
- Run Jolt Relay.

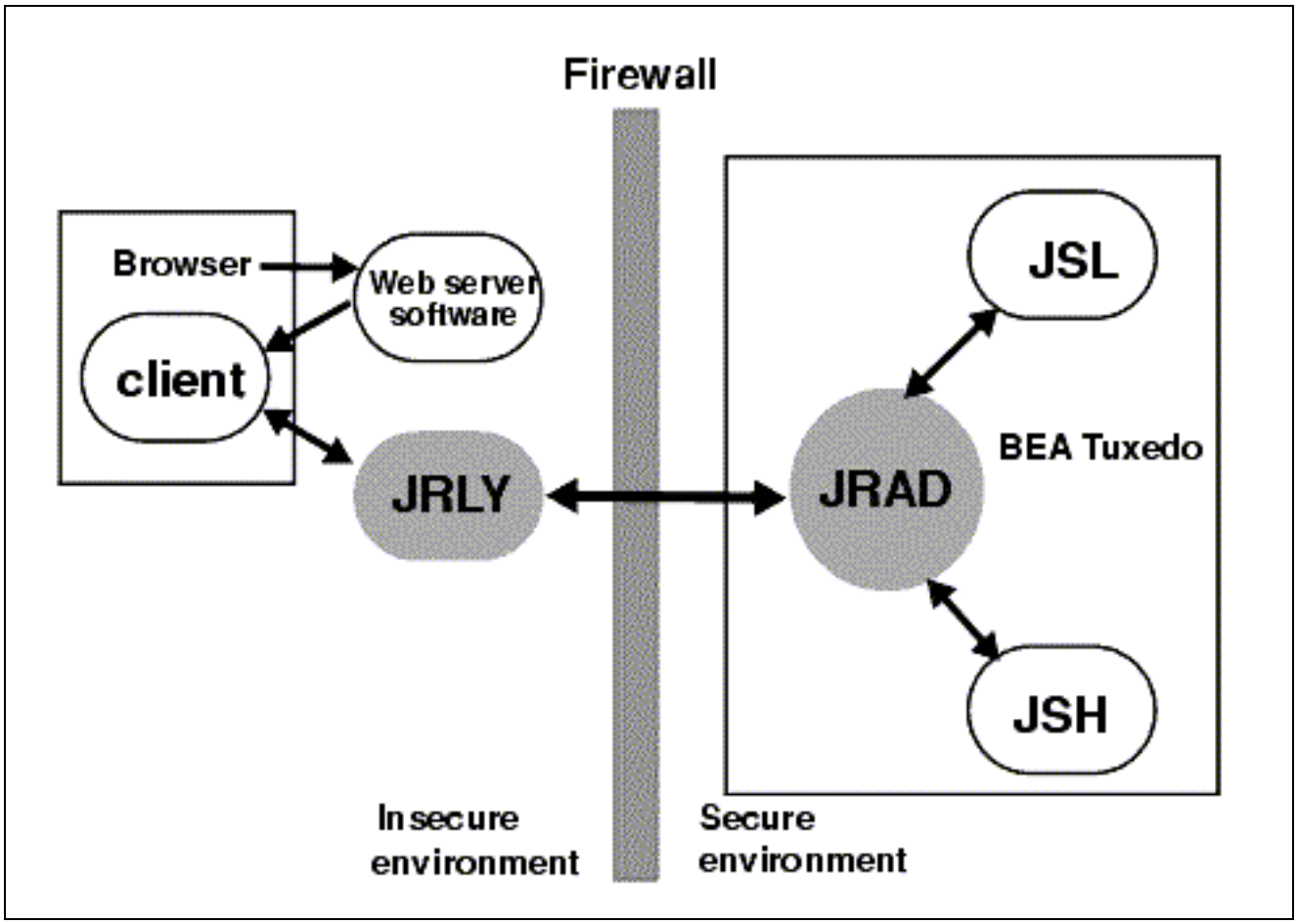
Understanding Jolt Internet Relay

This section discusses:

- Jolt Internet Relay architecture.
- A Jolt Internet Relay example.
- Implementation considerations.

Jolt Internet Relay Architecture

BEA Jolt Internet Relay provides an environment in which the PeopleSoft web server and application server can be further decoupled. This provides greater security at sites where security is of utmost importance. Jolt Internet Relay routes messages from a Jolt client to a Jolt Server Listener (JSL) or Jolt Server Handler (JSH), and eliminates the need for the JSL, JSH, and Tuxedo application to run on the same machine as the web server. Communication takes place between the JRLY and JRAD elements rather than between the Jolt client and JSL/JSH processes. Traditionally an application server domain opens between 2 and 6 ports for such communications. The use of Jolt relay restricts this to one port per domain on the application server machine. This enables an administrator to open just one port on the application server machine. The following diagram illustrates this feature:



Jolt Internet Relay Architecture

Jolt Internet Relay consists of two elements: Jolt Relay (JRLY) and Jolt Relay Adapter (JRAD). It's important to understand the difference between these two elements.

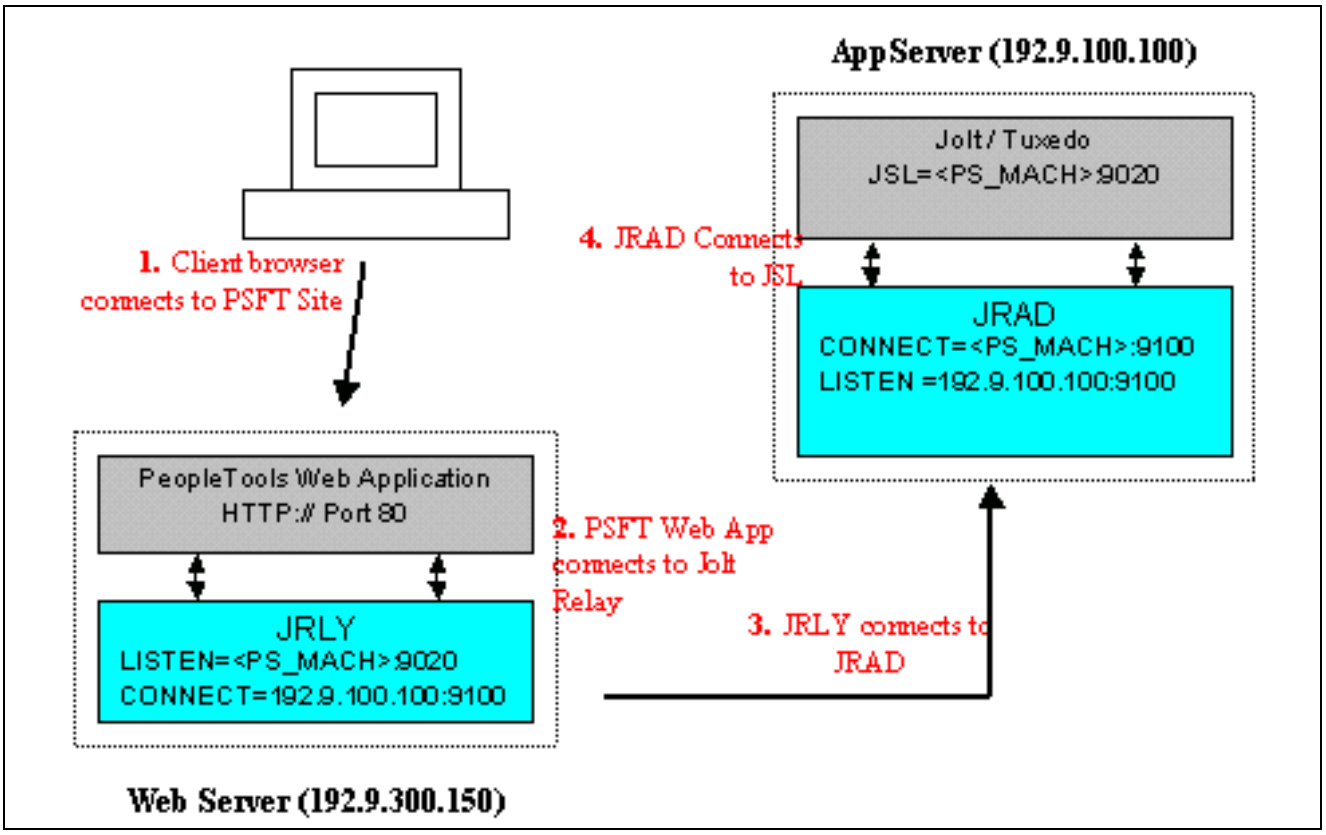
JRLY consists of a standalone program and configuration file; the program runs on the same machine as the web server. JRLY receives Jolt messages from a PeopleSoft web application and routes those messages to JRAD on the application server. It receives the Jolt message through one port, the LISTEN port, and connects to the JRAD by using another port, the CONNECT port. JRLY is sometimes referred to as a front-end relay.

JRAD runs on the same machine as the application server. It's configured on the application server domain as part of the PeopleSoft PSADMIN domain configuration procedure. JRAD listens for JRLY messages on its LISTEN port and transfers the message to the JSL or JSH. JRAD is sometimes referred to as a back-end relay.

A Jolt Internet Relay Example

The following example illustrates the relationship between the components, and most importantly, their respective port numbers. When you configure the JRLY system, it's very important to make sure that you specify the correct port numbers through which each component receives messages and to which port number they send messages. Any inconsistency results in a failed connection.

Note. In general, using JRLY with PIA is not recommended for performance reasons. For use with PIA, you must specify that the page servlet connect to the JRLY LISTEN port on the web server as opposed to specifying the JSL on the application server.



Jolt Relay in PeopleSoft

In the example, assume that the web server and the application server reside on separate machines. The following list describes what takes place within each numbered step:

1. The online PeopleSoft user connects to a URL to sign in to the PeopleTools web application.
2. Upon username authentication, the web application connects to port 9020 on the web server machine. Port 9020 reflects the JRLY Listener. The JRLY listener passes the request details to the JRLY connect process.
3. The JRLY connect process uses the machine IP address and port number to connect to the JRAD process on the application server machine.
4. JRAD passes the request on to the JSL, which initiates the transaction.

The return message to the web application follows the same path in reverse.

Note. A firewall might separate (and probably does in most cases) the web server and the application server.

Keep the following points in mind:

- The JRLY listener must match the port number to that in the Web Application's configuration.properties file.
- The JRLY connect must match the JRAD listener.
- The JRAD connect is set automatically by PeopleSoft to connect to the JSL.

Implementation Considerations

Keep the following points in mind as you configure the Jolt Internet Relay components:

- The jrly binary and its corresponding jrly.config file must exist in the same directory. To start multiple Jolt Relays on a machine, copy the jrly binary and jrly.config into each subdirectory, modify the parameters in the jrly.config file, and start Jolt Relay. On Windows, you can define multiple Jolt Relay services on a machine.
- You can start the JRLY process before or after you start JRAD. The JRLY attempts to connect to JRAD on the client request. If the JRLY is unable to connect to the JRAD, the client is denied access and disconnected. The connection will be retried upon the first use of PeopleTools..
- If you're installing Jolt Internet Relay on UNIX and anticipate a large number of concurrent connected clients, increase the file descriptors limit before running the JRLY executable.
- At runtime, if you get the following message:

```
[FRi JUNl 6 20:25:11 1997] JRLY:accept():accept failed,⇒
 err no: 23, strerror: File table overflow
```

  PeopleSoft recommends that you increase the MaxUSERS kernel parameter and regenerate the kernel.
- If you're unable to connect, make sure that you check the following items:
  - Port numbers do not match.

    Print out the jrly.cfg file and the psappsrv.cfg file and compare the port numbers that you specified.
  - Make sure that the application server is running.
  - Make sure that JRLY is running.
- Jolt Internet Relay can be installed on an intermediate machine rather than the web server machine if necessary. This extra level of indirection can cause peformance degradation.
- Make sure that JRAD is running on the application server and that you configure JRAD using PSADMIN.

Configuring JRLY

Configuring JRLY is identical on UNIX and Windows.

To configure JRLY, navigate to*TUXDIR*\udataobj\jolt\relay and open jrly.config in a text editor.

Important! On UNIX, you can edit this configuration file by using VI or an equivalent editor. However, on Windows, you must edit the file using an editor that preserves the file's UNIX line feeds. WordPad is valid for this purpose, but Notepad is not.

Modify the parameters in the configuration file to reflect the site specifications, as follows:

Parameter	Description
LOGDIR	*LOGDIR* specifies the directory where JRLY creates access and error log files. This directory must exist; the JRLY program does not start if it can't find this directory. The path that you specify for LOGDIR should be an absolute path (starting from / on UNIX systems, starting from a drive letter on Windows systems). The JRLY accepts relative path names, but LOGDIR is relative to the directory from which the JRLY program is started, unless you specify it as an absolute.

Parameter	Description
ACCESS_LOG	*ACCESS_LOG* specifies the name of the file where JRLY records access information. This log file is created in *LOGDIR*. If the log file already exists, the most recent information is appended to it. This parameter can be any valid file name. Everything after the equals sign (=) to the end of the line is considered as part of the file name, but leading and trailing blanks are ignored. **Note.** If the JRLY program can't create this file or open it for appending, the program exits.
ERROR_LOG	*ERROR_LOG* specifies the name of the file where JRLY records error information. This file follows all the rules that apply to the *ACCESS_LOG* parameter. JRLY_error_log is created in /tmp.

LISTEN	*LISTEN* specifies the host and port on the current machine (that is, the machine where you're installing Jolt Relay). JRLY listens for client connections. The following formats are acceptable:
	
	`LISTEN=192.9.100.100:9000`
	`LISTEN=//192.9.100.100:9000`
	`LISTEN=sp-ibm02:9000`
	`LISTEN=//sp-ibm02:9000`
	
	Specify the port number in decimal; it must match the port number that is specified by the *psserver* parameter in the configuration.properties file for the PIA web application.
	
	Note. If a machine has multiple network interfaces, you should use the IP address notation, because specifying the hostname could be ambiguous (the result is OS dependent). If the JRLY program can't establish a network listening end-point at the host and port specified, it prints an error and exits.
	
	The hostname that's specified for this parameter must be the name of the host on which the program is running.
	
	Note. You can create multiple configuration files to run multiple instances of JRLY. Each configuration file must specify a different port number for this parameter.

Parameter	Description

CONNECT	*CONNECT* specifies the location of the JRAD machine and process port on the application server machine to which the JRLY program connects. A JRLY program communicates only with a single JRAD. The address you specify for this parameter must match the JRAD listener address that's on the application server machine (check the PSAPPSRV.CFG file in *PS_HOME*/appserv/*domain*.) The JRAD doesn't have to be running when you start the JRLY program. JRLY attempts to connect to the JRAD when it first starts, and if the JRAD is not available, JRLY tries again whenever a new client connects to it. You can use any of the following formats for this parameter:
	`CONNECT=192.9.100.100:9100`
	`CONNECT=//207.135.44.91:9105`
	`CONNECT=sp-hp06:9105`
	`CONNECT=//sp-hp06:9105`
	Note. PeopleSoft has found that machine address formats are operating system and environment dependent. If one fails to connect to the application server, try another format.

SOCKETTIMEOUT	*SOCKETTIMEOUT* specifies the duration (in seconds) for which the Jolt Internet Relay Windows service blocks the establishment of new socket connections to allow network activity (new connections, data to be read, closed connections) to complete. It's valid only on Windows machines. *SOCKETTIMEOUT* also affects the Service Control Manager (SCM). When the SCM requests that the service stop, the SCM needs to wait at least the number of seconds specified by this parameter.

Configuring JRAD

The JRLY connect port connects to the JRAD listener port that is specified on the application server machine. JRAD then routes the message to Jolt, either using the JSL for initial connection from a web client, or using the JSH for all subsequent connections from a web client. The return message follows the same path in reverse.

To configure JRAD:

1. Launch the PSADMIN utility.
2. Navigate to the PeopleSoft Domain Administration menu and select *Configure this domain*.
3. In the Quick Configure menu, select the number for the Jolt Relay option, to enable Jolt Internet Relay.
4. Select the JRAD Port option, and enter the appropriate port number for the JRAD Port.

Note. The JRAD (listener) port number must match the JRLY connect port that you previously configured.

See Also

Chapter 2, "Using the PSADMIN Utility," Using the Quick-Configure Menu, page 10

Running Jolt Relay

This section discusses how to:

- Use the JRLY administration program.
- Run Jolt Relay on Windows.
- Run Jolt Relay on UNIX.

Using the JRLY Administration Program

You use the jrly command located in *TUXDIR*\udataobj\jolt\relay to administer Jolt Relay on all platforms. You can use the following jrly command options at any time:

- jrly -version

  Display the current version of the JRLY binary.

- jrly -help

  Display a summary of command-line options with brief descriptions.

Running Jolt Relay on Windows

On Windows, you set up Jolt Relay to run as a service. On other platforms you must run Jolt Relay directly.

See Chapter 12, "Setting Up Jolt Internet Relay," Running Jolt Relay on UNIX, page 270.

If you want to install multiple Jolt Relay services, you must specify a string to be used as a *display suffix* that uniquely identifies each additional service you install. You subsequently use the suffix to identify each service in commands. An additional service with the suffix MyJoltRelay, for example, is called *BEA Jolt Relay_MyJoltRelay*, but you refer to it using only the suffix. You can omit the suffix when installing only one of these services, which becomes the default Jolt Relay service, called *BEA Jolt Relay*.

Note. All administrative commands in the following table except -start and -stop require that you have write access to the Windows registry. The -start and -stop commands require that you have Windows service control access. These requirements are based on Windows user restrictions.

Command	Description
jrly -install [*display_suffix*]	Install JRLY as a Windows service.
jrly -remove [*display_suffix* | -all]	Remove one instance, all instances, or the default JRLY Windows service.

Command	Description
jrly -set [-d *display_suffix*] -f *config_file*	Update the registry with the full path of a new configuration file for the specified JRLY service. **Note.** You can run multiple Jolt Relay services by specifying a different display suffix along with the name of a different configuration file for each installed service. Each configuration file must contain a unique value for the LISTEN parameter that specifies a different port. This is essential to avoid port clashes when running the services concurrently. You must run this command before the service starts.
jrly -manual [*display_suffix*]	Set the start/stop to manual. This command sets the specified JRLY service to be manually controlled, using either the command-line options or the Service Control Manager (SCM).
jrly -auto [*display_suffix*]	Set the start/stop to automatic. This command sets all the operations for a specified JRLY service to be automatically started when the OS boots and stopped when the OS shuts down.
jrly -start [*display_suffix*]	Start the specified JRLY service.
jrly -stop [*display_suffix*]	Stop the specified JRLY service.

Running Jolt Relay on UNIX

This section discusses how to start and stop Jolt Relay directly from a command line on UNIX.

To start Jolt Relay on UNIX:

1. Change directories to the Jolt Relay directory within your BEA Tuxedo installation:

   ```
   cd $TUXDIR/udataobj/jolt/relay
   ```

2. Run the following command:

   ```
   jrly -f jrly_config &
   ```

   Where *jrly_config* is the name of a Jolt Relay configuration file.

   You can run multiple instances of Jolt Relay by using a different port for each instance. You run JRLY once for each instance, and specify a different configuration file each time. Each configuration file must contain a value for the LISTEN parameter that specifies a different port.

   The & causes JRLY to run in the background.

To shut down Jolt Relay on UNIX, use the UNIX kill -9 command.

CHAPTER 13

Replicating an Installed Environment

This appendix provides an overview of environment replication and discusses how to:

- Replicate a web server environment.
- Replicate an application server environment.
- Replicate the PeopleSoft Process Scheduler environment.
- Reconfigure replicated Environment Management components.

Understanding Environment Replication

Environment Replication involves taking a working, well-tested environment, and copying the Tools binary and configuration files to a new location to create a new environment by making minor modifications to the new configurations.

To further define the term “environment”, there are three separate components that can have multiple environment configurations: Web Server, Application Server, and Process Scheduler Server.

A web server environment consists of one or more site directories under the WEBSERV directory within a single *PS_HOME* location. Each directory contains configurations that point to a single Application Server through a JSL port designation. Although a site may point to more than one redundant application server machine names for failover or load balancing, each machine name is given a unique JSL port number. Another *PS_HOME* directory location on a Web Server machine would be considered another environment.

A single Application Server environment consists of one or more domain directories under the APPSERV directory within a single *PS_HOME* location. Each domain contains configuration settings that point to a single database. Multiple domains can be configured to point to the same database for failover or load balancing. Each domain has its own server processes and must be configured to have unique WSL and JSL port numbers.

A single Process Schedule Environment consists of one or more connect database directories under the APPSERV\PRCS directory within a single *PS_HOME* location. Each database that needs to schedule processes must have its own Process Scheduler server configured separately. The configuration files are kept in their own directory location under the PRCS directory.

Note. For the purposes of this document, the term “environment” refers to all the server processes under a single *PS_HOME* directory location on each of the servers. The Web Server environment is separate from the Application Server environment, which is separate from the Process Scheduler Server environment.

Important! When you configure and run an Environment Management agent, hub, or viewer, files associated with that component are created and updated with information that refers to the absolute directory structure of your PeopleSoft system. When you replicate your installed environment, the information in those files is no longer valid for the new location. You must reconfigure the Environment Management components after replicating them.

See Chapter 13, "Replicating an Installed Environment," Reconfiguring Replicated Environment Management Components, page 275.

Problems Associated with Environment Replication

In the field, there are many methods an installer, on-site consultant, or customer may perform in order to replicate a single server environment. Usually, this is done by procedures memorized by a consultant, without any formalized steps to follow. Also there are no formal final verification steps to ensure a proper environment setup.

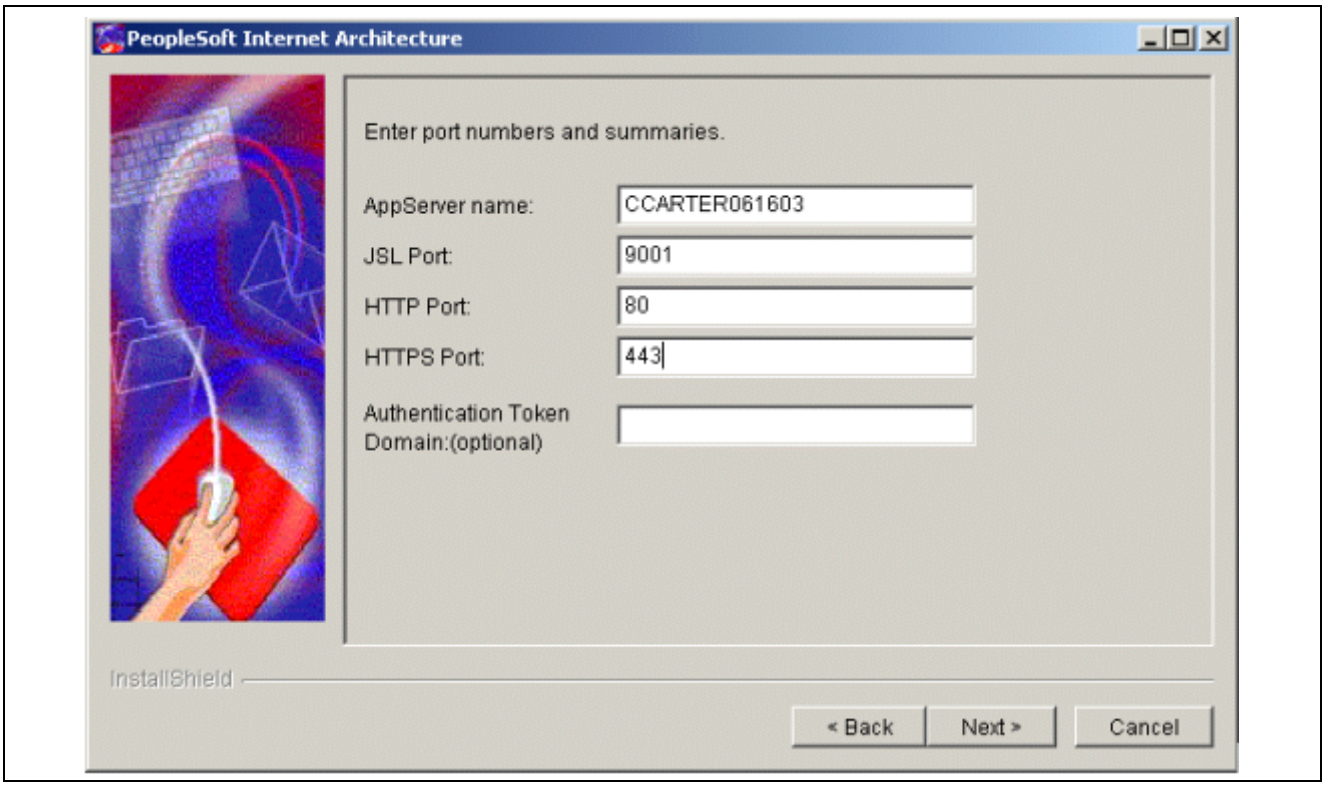
Discussions with field consultants who have performed environment replication many times in the past revealed that environment replication is much more prevalent within a single machine, rather than replicating to new machines. Since the need exists for creating new environments to keep binaries separate for patch acceptance and testing, new *PS_HOME* directory locations are created for this purpose. This is usually performed on the same machine for the sake of simplicity. Hence, there is no need to install and maintain 3rd party software on separate machines.

Replicating a Web Server Environment

Use the following steps to replicate the web server environment:

1. Copy the PeopleSoft Web Server *PS_HOME* directory structure to a new location, whether it is a new machine, or same machine with a new high level *PS_HOME* directory name.
2. Run *PS_HOME*\setup\mpinternet\setup.exe or equivalent from the copied/cloned location.
3. Select Redeploy PeopleSoft Application.
4. Choose a site name.
5. Enter the appserver name and port information.

The following is a sample screen that shows the port numbers and summaries.



Sample PeopleSoft Internet Architecture Installation screen

Note. Be sure to enter a different JSL port value when replicating on the same machine.

Replicating an Application Server Environment

For Application Server environment replication, the first step again is to copy the *PS_HOME* directory structure on the application server to the new location, either to a new machine or to a new directory on the same machine. This procedure copies the binaries, as well as the existing domains that are linked to a database.

After the delivery of the new *PS_HOME*, the existing domains need to be reconfigured in order for the domains to be booted and run. This procedure generates a new unique Tuxedo shared memory ID value that is different than the one in the original domain.

If you keep the original domain names and configuration settings (Database name, User ID, ports, etc), you can reconfigure the domain by simply running PSADMIN through the command line, using the *-c configure* command for each domain.

If you keep the original domain names but you need to change any of the configuration settings, then you should use the PSADMIN interactive tool. In this case, you need to do the following:

1. Select each domain to configure.
2. Select *Y* to change the configuration values.
3. Input the new values as needed.

Alternatively, you can edit the PSAPPSRV.CFG file directly in the domain directory. You can then use the command line interface to PSADMIN to reconfigure the domain.

If you want to rename the domain names to be more intuitive for its task (for example, Prod, dev, test, demo, etc), then you can utilize the PSADMIN import option.

Steps to Replicate PeopleSoft Application Server Using Import Option

Use the following steps to replicate an application server environment:

1. In PSADMIN, select *1* for Application Server.
2. Select option *4* to import domain configuration.
3. Select option *1* to import from file.
4. Enter a domain name that you want to create and press ENTER.
5. Select *1* to boot up the server.

The command line to rename a domain would be:

```
PSADMIN -c import PS_HOME -d domain_name -n new_name
```

The values are the current *PS_HOME* directory location, the domain name to rename, and the new name to create. A new domain name is created using the configuration values of the original name.

The import option also accepts a path to a configuration file (PSAPPSRV.CFG) for importing new configuration parameters. The configuration file can be manually modified for the users specifications, and then imported to create new domains.

Replicating the PeopleSoft Process Scheduler Environment

Process Scheduler Server configuration file, psprcs.cfg, is stored under *PS_HOME*\appserv*database*psprcs.cfg. Only one database can be accessed by a single Process Scheduler Server, although multiple servers can access the same database.

The procedure for replicating Process Scheduler environments is the same as it is for Application Servers. Changes to the configuration file require that the environment be reconfigured in order to recreate the Tuxedo binary configuration file. The process is the same as noted above, using the PSADMIN command line process to create, configure, and import Process Scheduler configurations.

The following configuration settings from psprcs.cfg point to the current *PS_HOME* location and can be moved to new locations without modifications: CBLBIN, CRWRPTPATH, TOOLBIN, TOOLBINSVR, DataMover dirs, and nVision dirs. The binaries will substitute the correct directory location for *PS_HOME*.

The configuration settings pointing to hard-coded locations are DBBIN, WINWORD, and SQRBIN. The DBBIN, Database connection binaries, and WINWORD settings, would presumably reside in the same common directory across environments. The SQRBIN setting generally points to a common file server location, which was created with the initial CD install. As long as the drive mappings remain intact, this setting should remain unmodified as well.

Reconfiguring Replicated Environment Management Components

This section discusses how to:

- Reconfigure an environment management agent.
- Reconfigure the environment management hub.
- Reconfigure the environment management viewer.

Reconfiguring an Environment Management Agent

The following files are required by an environment management agent:

- *New_PS_HOME*\PSEMAgent\StartAgent.bat (StartAgent.sh on UNIX)
- *New_PS_HOME*\PSEMAgent\StopAgent.bat (StopAgent.sh on UNIX)
- *New_PS_HOME*\PSEMAgent\envmetadata\config\configuration.properties
- *New_PS_HOME*\PSEMAgent\envmetadata\config\Logconfig.properties
- *New_PS_HOME*\PSEMAgent\envmetadata\data\matchers.xml
- *New_PS_HOME*\PSEMAgent\envmetadata\logs\emf.log

Note. *New_PS_HOME* is the directory where PeopleTools is located in your replicated environment.

To reconfigure the agent:

1. Delete all subdirectories below *New_PS_HOME*\PSEMAgent\envmetadata\data\

   Note. Don’t delete the matchers.xml file in this location.

2. Delete the following directories:
   - *New_PS_HOME*\PSEMAgent\envmetadata\PersistentStorage
   - *New_PS_HOME*\PSEMAgent\envmetadata\scratchpad
   - *New_PS_HOME*\PSEMAgent\envmetadata\transactions
3. Modify StartAgent.bat (or StartAgent.sh on UNIX).

   Ensure that references to the drive and path of the PSEMAgent directory are correct for the New *PS_HOME* location.
4. Modify StopAgent.bat (or StopAgent.sh on UNIX).

   Ensure that references to the drive and path of the PSEMAgent directory are correct for the New *PS_HOME* location.
5. Verify the settings in the agent configuration file:

   New_PS_HOME\PSEMAgent\envmetadata\config\configuration.properties

   Properties that specify path locations must be valid for the replicated agent in its new location, but the same hub is still addressed by all agents. Ensure that the following settings are correct for the replicated agent:
   - hubURL

- windowsdrivestocrawl
- unixdrivestocrawl

Reconfiguring the Environment Management Hub

Although replication produces multiple Environment Management hub directory structures, and starting PIA makes each of those hubs active, you specify only one hub as the location of the Environment Management log files. You must reconfigure the newly replicated hub only if you've modified the agents to specify it as the logging hub.

The following files are required by the environment management hub:

- *New_PS_HOME*\webserv\peoplesoft\applications\peoplesoft\PSEMHUB\envmetadata\config\configuration.properties
- *New_PS_HOME*\webserv\peoplesoft\applications\peoplesoft\PSEMHUB\envmetadata\config\Logconfig.properties
- *New_PS_HOME*\webserv\peoplesoft\applications\peoplesoft\PSEMHUB\envmetadata\data\data.txt
- *New_PS_HOME*\webserv\peoplesoft\applications\peoplesoft\PSEMHUB\envmetadata\logs\log.txt

Note. *New_PS_HOME* is the directory where PeopleTools is located in your replicated environment.

To reconfigure the hub:

1. Delete all subdirectories below *New_PS_HOME*\webserv\peoplesoft\applications\peoplesoft\PSEMHUB\envmetadata\data\

   Note. Don't delete the data.txt file in this location.

2. Delete the following directories:
   - *New_PS_HOME*\webserv\peoplesoft\applications\peoplesoft\PSEMHUB\envmetadata\PersistentStorage
   - *New_PS_HOME*\webserv\peoplesoft\applications\peoplesoft\PSEMHUB\envmetadata\scratchpad
   - *New_PS_HOME*\webserv\peoplesoft\applications\peoplesoft\PSEMHUB\envmetadata\transactions

Reconfiguring the Environment Management Viewer

The following files are required by an environment management viewer:

- *New_PS_HOME*\PSEMViewer\GetEnvInfo.bat
- *New_PS_HOME*\PSEMViewer\envmetadata\config\configuration.properties
- *New_PS_HOME*\PSEMViewer\envmetadata\config\Logconfig.properties
- *New_PS_HOME*\PSEMViewer\envmetadata\data\app_server.xsl
- *New_PS_HOME*\PSEMViewer\envmetadata\data\environmentinfo.xsl
- *New_PS_HOME*\PSEMViewer\envmetadata\data\environmentupdates.xsl
- *New_PS_HOME*\PSEMViewer\envmetadata\data\file_server.xsl
- *New_PS_HOME*\PSEMViewer\envmetadata\data\host.xsl
- *New_PS_HOME*\PSEMViewer\envmetadata\data\NEW_PS_LOGO.gif

- *New_PS_HOME*\PSEMViewer\envmetadata\data\prcs_scheduler.xsl
- *New_PS_HOME*\PSEMViewer\envmetadata\data\viewer.css
- *New_PS_HOME*\PSEMViewer\envmetadata\data\viewer.html
- *New_PS_HOME*\PSEMViewer\envmetadata\data\web_server.xsl

Note. *New_PS_HOME* is the directory where PeopleTools is located in your replicated environment.

To reconfigure the hub:

1. Delete all subdirectories below *New_PS_HOME*\PSEMViewer\envmetadata\data\

   Note. Don’t delete the files in this location, just the subdirectories.

2. Delete the following directories:
   - *New_PS_HOME*\PSEMViewer\envmetadata\PersistentStorage
   - *New_PS_HOME*\PSEMViewer\envmetadata\scratchpad

APPENDIX A

BEA WebLogic 8.1 Managed Server Architecture

In the current release, the PIA configuration of BEA WebLogic Server has been expanded to take advantage of WebLogic's managed server architecture. This appendix provides overviews of web applications in PIA, WebLogic domain types, WebLogic domain directory structure and files, and PIA install and reinstall options, and discusses:

- Administering a WebLogic server life cycle.
- Tuning performance and monitoring resources.
- Changing configuration settings.
- Applying an example single-server configuration.
- Applying an example multi-server configuration.

See Also

Chapter 6, "Working with BEA WebLogic," page 97

Web Applications in PIA

PIA is packaged as a J2EE Enterprise Archive and is comprised of five J2EE web applications, commonly referred to as webapps. The five webapps are as follows.

PORTAL	PeopleSoft Portal
PSIGW	PeopleSoft Integration Broker
PSOL	PeopleSoft On-line Library
PSEMHUB	PeopleSoft Environment Management Framework
PSINTERLINKS	PeopleSoft Business Interlinks

In addition to the PeopleSoft webapps, three more webapps are added when you install PIA on a WebLogic server machine. These three webapps are not added as part of the PeopleSoft Enterprise Archive, but instead are defined as individual webapps provided by BEA. These three webapps are as follows.

HttpProxyServlet	Reverse Proxy Server – Proxy to a single content server per URL. Each URL can provide unique content.
HttpClusterServlet	Reverse Proxy Server – Proxy to multiple WebLogic servers. All content servers provide access to the same content for load balancing.
Console	BEA's administrative console for WebLogic Server.

WebLogic Domain Types

This section provides an overview of Weblogic domain types and discusses:

- Single-server domain.
- Multi-server domain.
- Distributed managed server.
- Common default settings.
- Single-server and multi-server/distributed server analogy.
- Domain topology.

Understanding WebLogic Domain Types

During PIA setup, you can choose between two different WebLogic domain configurations: a single-server domain and a multi-server domain. In addition, a multi-server domain can be expanded across multiple machines using the *distributed managed server* option. Each of these domain configurations has a specific purpose but is fully customizable beyond that purpose.

Single-Server Domain

The single-server domain configuration consists of a single WebLogic server instance, with the WebLogic administration application and all of the web components of PeopleSoft Internet Architecture deployed to use it. This configuration is very similar in design to the WebLogic domain provided in previous PeopleTools releases.

This configuration is intended for single-user or very small scale, non-critical production environments. It can be used as a starting point for you to familiarize yourself with BEA WebLogic Server 8.1

In this configuration, the resources used to administer WebLogic Server and your PeopleSoft application are not isolated from one another, therefore leaving the applications vulnerable to possible resource starvation. The low resource requirements of this configuration make it ideal for small scale and non-production usage.

This configuration creates the following server:

PIA A server for WebLogic domain administration with PeopleSoft J2EE applications deployed.

Single-Server Webapp Deployment

Some of the webapps deployed in a single-server domain configuration must be accessed using a modified URL:

http://*server*:*port*/*webapp_name*/...

The single-server domain configuration deploys webapps as follows:

Application	Deployed to Server	Webapp Name in URL
PORTAL	PIA	(not needed)
PSIGW	PIA	PSIGW

Application	Deployed to Server	Webapp Name in URL
PSOL	PIA	PSOL
PSEMHUB	PIA	PSEMHUB
PSINTERLINKS	PIA	PSINTERLINKS
Console	PIA	console
HttpProxyServlet	Defined but not deployed.	(not needed)
HttpClusterServlet	Defined but not deployed.	(not needed)

Single-Server Domain Specific Settings

To configure the single-server domain specific settings, launch the WebLogic Server Console.

In the console, navigate to Servers, PIA, Configuration, General to configure the PIA server. The default web application for the PIA server is PORTAL. The single-server domain specific default settings for the PIA server are as follows:

Setting	Default Value
IP address.	* (all local IPs).
HTTP Listen port.	80 (set during PIA setup).
SSL.	Enabled with demonstration self-signed digital certificates.
HTTPS Listen port.	443 (set during PIA setup).

Note. To configure SSL, you must also define SSL certificates.

See Chapter 6, "Working with BEA WebLogic," Implementing WebLogic SSL Keys and Certificates, page 115.

Multi-Server Domain

The multi-server domain configuration consists of seven server definitions, a WebLogic cluster, and the web components of PIA split across multiple servers. This configuration takes advantage of WebLogic's administration server and managed server architecture.

This configuration is intended for production environments.

A production application warrants process and resource pool isolation for greater stability and optionally tighter security controls, which this configuration provides. In this configuration, the resources used for WebLogic domain administration and monitoring are isolated from similar resources used to support the PIA application. A server process named *WebLogicAdmin* performs nothing but WebLogic administration, which includes domain administration and monitoring. Continuing that separation, the individual web applications of PIA might be, and usually are, isolated from each other. The PIA applications are targeted and deployed across a portion of the six remaining server definitions, all of which are classified as *managed servers,* which are delivered in the multi-server configuration.

This configuration creates the following servers:

WebLogicAdmin	Administration server for WebLogic domain administration.
PIA	Server for the PeopleSoft Portal, integration gateway, and PeopleSoft Business Interlinks applications.
PIA1	Server for the PeopleSoft Portal, integration gateway, and PeopleSoft Business Interlinks applications.
PIA2	Server for the PeopleSoft Portal, integration gateway, and PeopleSoft Business Interlinks applications.
PSOL	Server for the PeopleSoft Online Library (PeopleBooks) application.
PSEMHUB	Server for the PeopleSoft Environment Management Framework application.
RPS	Server for WebLogic reverse proxy server applications.

The multi-server domain configuration deploys webapps as follows:

Multi-Server Webapp Deployment

Some of the webapps deployed in a multi-server domain configuration must be accessed using a modified URL:

http://*server*:*port*/*webapp_name*/...

The multi-server domain configuration deploys webapps as follows:

Application	Deployed to Server, Cluster (members)	Webapp Name in URL
PORTAL	PIA, PeopleSoftCluster (PIA1, PIA2)	(not needed)
PSIGW	PIA, PeopleSoftCluster (PIA1, PIA2)	PSIGW
PSOL	PSOL	PSOL
PSEMHUB	PSEMHUB	PSEMHUB
PSINTERLINKS	PIA, PeopleSoftCluster (PIA1, PIA2)	PSINTERLINKS
Console	WebLogicAdmin	console
HttpProxyServlet	RPS	(not needed)
HttpClusterServlet	Defined but not deployed.	(not needed)

Multi-Server Domain Specific Default Settings

To configure the multi-server domain specific settings, launch the WebLogic Server Console.

In the console, navigate to Servers, WebLogicAdmin, Configuration, General to configure the WebLogicAdmin server. The WebLogicAdmin server has no default web application. The domain specific default settings for the WebLogicAdmin server are as follows:

WebLogicAdmin Setting	Default Value
IP address	* (all local IPs)

WebLogicAdmin Setting	Default Value
HTTP Listen port	9999
SSL	Disabled

In the console, navigate to Servers, PIA, Configuration, General to configure the PIA server. The default web application for the PIA server is PORTAL. The domain specific default settings for the PIA server are as follows:

PIA Setting	Default Value
IP address	* (all local IPs)
HTTP Listen port	80 (set during PIA setup)
SSL	Enabled with demonstration self-signed digital certificates.
HTTPS Listen port	443 (set during PIA setup)

In the console, navigate to Servers, PIA1, Configuration, General to configure the PIA1 server. The default web application for the PIA1 server is PORTAL. The domain specific default settings for the PIA1 server are as follows:

PIA1 Setting	Default Value
IP address	Locally determined hostname.
HTTP Listen port	80 (set during PIA setup)
SSL	Enabled with demonstration self-signed digital certificates.
HTTPS Listen port	443 (set during PIA setup)

In the console, navigate to Servers, PIA2, Configuration, General to configure the PIA2 server. The default web application for the PIA2 server is PORTAL. The domain specific default settings for the PIA2 server are as follows:

PIA2 Setting	Default Value
IP address	127.0.0.1
HTTP Listen port	80 (set during PIA setup)
SSL	Enabled with demonstration self-signed digital certificates.
HTTPS Listen port	443 (set during PIA setup)

In the console, navigate to Servers, PSOL, Configuration, General to configure the PSOL server. The default web application for the PSOL server is PSOL. The domain specific default settings for the PSOL server are as follows:

PSOL Setting	Default Value
IP address	* (all local IPs)
HTTP Listen port	6001
SSL	Disabled

In the console, navigate to Servers, PSEMHUB, Configuration, General to configure the PSEMHUB server. The default web application for the PSEMHUB server is PSEMHUB. The domain specific default settings for the PSEMHUB server are as follows:

PSEMHUB Setting	Default Value
IP address	* (all local IPs)
HTTP Listen port	8001
SSL	Disabled

In the console, navigate to Servers, RPS, Configuration, General to configure the RPS server. The default web application for the RPS server is HttpProxyServlet. The domain specific default settings for the RPS server are as follows:

RPS Setting	Default Value
IP address	* (all local IPs)
HTTP Listen port	8080 (set during PIA setup)
SSL	Enabled with demonstration self-signed digital certificates.
HTTPS Listen port	8443 (set during PIA setup)

Note. To configure SSL, you must also define SSL certificates.

See Chapter 6, "Working with BEA WebLogic," Implementing WebLogic SSL Keys and Certificates, page 115.

Distributed Managed Server

The *distributed managed server* configuration, although listed alongside the single-server and multi-server domain types, is not a true domain type. It's an optional extension for an existing multi-server configuration that's used to extend a WebLogic domain configuration across multiple machines in a heterogeneous network. For example, on one machine you perform a PIA install and create a multi-server domain. On a second machine you again perform a PIA install but select to create a distributed managed server. You stop, start, and administer the distributed managed server on the second machine just as if it was a managed server local to the primary server's machine.

As with the multi-server domain type, this configuration takes advantage of WebLogic's managed server architecture.

Note. Only one managed server can be run per distributed managed server domain directory. If you intend to run multiple distributed managed servers on a single machine, perform the PIA install and create unique distributed managed server domain directories, one for each distributed managed server that you intend to run on that machine.

This configuration is intended for production environments that encompass multiple machines.

A distributed managed server configuration provides the same benefits as a multi-server configuration with the added benefit of hardware isolation. This option requires a multi-server installation to be performed to some other location, which will contain the configuration for this distributed managed server.

Distributed Managed Server Specific Defaults

The server configuration settings for a distributed managed server are maintained via that domain's administration server and are stored locally on that administration server. Configuration settings are replicated to a managed server during its startup, but are only maintained as a read-only backup copy for that individual managed server in the event that the administration server isn't available the next time this particular managed server needs to be started.

Common Default Settings

Single-server and multi-server domain configurations have many settings in common.

Domain Defaults

Many of these common settings can be configured in the WebLogic Server Console, but some are configured in other environments. Default values are listed when available.

Setting	Default Value	Where To Configure
SSL functionality	Enabled with demonstration self-signed digital certificates.	Console: Servers, *server name*, Configuration, Keystores & SSL. Command line: pskeymanager.
Server logs	*Weblogic domain*\logs*server name*_*.log	Console: Servers, *server name*, Logging, Server.
HTTP access log	Disabled	Console: Servers, *server name*, Logging, HTTP.
HTTP keep-Alive	30 seconds	Console: Servers, *server name*, Protocols, HTTP.
HTTPS keep-Alive	60 seconds	Console: Servers, *server name*, Protocols, HTTP.
Low JVM memory monitoring	On	Console: Servers, *server name*, Configuration, Tuning.
Stuck thread seconds interval	600	Console: Servers, *server name*, Configuration, Tuning.

Setting	Default Value	Where To Configure
Domain administration port	Disabled	Console: Servers, *server name*, Configuration, General.
PORTAL webapp HTTP session monitoring	On (applies only to servers running PORTAL)	Console: Deployments, Applications, peoplesoft, PORTAL, Monitoring.
System administrator user ID	system (set during PIA setup)	Console: Security, Realms, myrealm, Users.
System administrator password	password (set during PIA setup)	Console: Security, Realms, myrealm, Users.
System operator user ID	operator	Console: Security, Realms, myrealm, Users.
System operator password	password	Console: Security, Realms, myrealm, Users.
System monitor user ID	monitor	Console: Security, Realms, myrealm, Users.
System monitor password	password	Console: Security, Realms, myrealm, Users.

Script and Environment Defaults

Modify these settings by editing a setEnv script or applying command line parameter overrides to WebLogic control scripts.

The following settings specify the names and structure of various directories on the web server machine.

Setting	Default Value	Description/Override
PS_HOME	(none)	PeopleSoft home directory (set during PIA setup).
BEA_HOME	(none)	BEA home directory (set during PIA setup).
WL_HOME	(none)	WebLogic 8.1 home directory (set during PIA setup).
DOMAIN_NAME	peoplesoft	Name of this WebLogic domain (set during PIA setup).
JAVA_HOME	(Depends on the OS platform.)	Location of Java. Set during PIA setup or with a call to WebLogic's CommEnv script.

Note. You configure Java VM options including JVM memory size using the JAVA_OPTIONS_*OSplatform* parameter, during PIA setup.

The following are miscellaneous settings.

Setting	Default Value	Description/Override
HOSTNAME	*Local hostname*	Set during PIA setup.
PRODUCTION_MODE	TRUE	Enable WebLogic production mode (set during PIA setup).
DISCOVERY_MODE	FALSE	Disable auto detection of unregistered applications. Script: setEnv
WLS_USER	Operator	Use to stop WebLogic with stop scripts and run it as a Windows NT service.
WLS_PW	Password	Use to stop WebLogic with stop scripts and run it as a Windows NT service.
ADMINSERVER_PROTOCOL	HTTP	Protocol used for managed server to connect to administration server (not used in single-server domain).
ADMINSERVER_HOSTNAME	Single-server: *local hostname*. Multi-server: *local hostname*. Distributed server: (none — set manually).	Administration server's hostname that managed servers attempt to connect to by default when started. Set during PIA setup (except distributed server).
ADMINSERVER_PORT	Single-server: *HTTP port of PIA server*. Multi-server: 9999. Distributed server: (none — set manually).	Administration server's Listen port that managed servers attempt to connect to by default when started. Set during PIA setup (except distributed server).
ADMINSERVER_SERVERNAME	Single-server: PIA. Multi-server: WebLogicAdmin. Distributed server: WebLogicAdmin.	WebLogic server instance name of this domain's administration server, used for stopping and starting the server.
WL_VERSION	*Detected major WebLogic version*.	
WL_SERVICEPACK	*Detected minor WebLogic version*.	
WL_PATCH	*Detected WebLogic patch version*.	
BACKGROUND_PROCESS	TRUE	Run WebLogic server as a background process. On UNIX you can force foreground execution using the start script's –foreground option.

The following are debugging output settings.

Setting	Default Value	Description/Override
SET CAPTURE_STDOUT_STDERR	FALSE	(Windows only) Capture standard output and standard error of a WebLogic server running as a foreground process. You can also set this with the start script's –capture option.
ENABLE_JDPA_DEBUG	FALSE	(PeopleSoft development only) Enable JDPA debug support. You can also set this with the start script's –debug option.
ENABLE_VERBOSE_GC	FALSE	Enable verbose output of Java's garbage collector. You can also set this with the start script's –verbose:gc option.
ENABLE_VERBOSE_SSL	FALSE	Enable SSL debug support. Produces verbose SSL output. You can also set this with the start script's –verbose:ssl option.
ENABLE_VERBOSE_WL	FALSE	Enable verbose output for the core WebLogic server (not verbose output of PIA). You can also set this with the start script's –verbose:wl option.

The following are HTTP forward proxy support settings.

Setting	Default Value	Description/Override
ENABLE_HTTP_PROXY	FALSE	Enable the use of the forward http proxy.
HTTP_PROXY_HTTPHOST	(none)	IP address or hostname of the forward HTTP proxy server for HTTP requests.
HTTP_PROXY_HTTPPORT	(none)	HTTP Port number of the forward HTTP proxy server for HTTP requests.
HTTP_PROXY_HTTPSHOST	(none)	IP address or hostname of the forward HTTP proxy server for HTTPS requests.
HTTP_PROXY_HTTPSPORT	(none)	HTTP Port number of the forward HTTP proxy server for HTTPS requests.
HTTP_PROXY_NONPROXY_ HOSTS	*localhost*, *local hostname*, and *domainname*.	Host names and domain names of content servers that will not be proxied.

Single-Server and Multi-Server/Distributed Server Analogy

In a production environment it is highly recommended to use a multi-server or distributed server configuration. The multi-server selection provides an initial WebLogic domain configuration that consists of multiple server definitions: one administration server and multiple managed servers, each with a specific purpose. The distributed server option augments a multi-server configuration by expanding the domain configuration across multiple machines.

An analogy that can describe the differences between the single-server configuration and a multi-server or distributed server configuration uses the example of two common beverage distribution models: a single lemonade stand, and a large chain of coffee shops.

Single-Server Model — The Lemonade Stand

With a lemonade stand you have yourself, one table, one collection of resources and ingredients, and a single recipe. If any of these are ruined, your lemonade stand is out of business. Similarly, in a single-server configuration you have one server process, one machine, one collection of resources and program files, and a single domain configuration file. If any of those is ruined, your web server is out of business. Throughput can also be an issue for both a lemonade stand and a single-server configuration. For example, if the weather is warm and you receive a rush of thirsty customers, your throughput will reach a maximum at either the rate you accept new orders, ring up cashier transactions, or prepare drinks. If you accept orders and ring up transactions in half the time that you prepare drinks, adding a second bartender would double your throughput, but in this configuration you can't do that because it's only you. Likewise, the single-server configuration is constrained to provide server configuration and all web server based portions of PIA on a single process. The multi-server model enables you to overcome these limitations.

Multi-Server Model — The Coffee Shop Chain

Within the same analogy, a large chain of coffee shops has multiple employees, multiple cashier stations, multiple coffee prep stations, and recipes archived and known to multiple employees. Similarly, in a multi-server configuration you can have multiple machines, multiple collections of resources and program files, multiple web server processes, and a replicated domain configuration file. In this model, if any of those resources is ruined, work simply shifts to the next instance of that resource. In addition, throughput can be maintained. For example, if the coffee house received a rush of parched customers, additional cashiers and beverage engineers could be added to maintain throughput. Likewise, in the multi-server configuration, an increase in PeopleSoft Portal usage can easily be accommodated by configuring an additional WebLogic server instance to also serve the PeopleSoft Portal application. In this way, the multi-server model supports extendibility and resource independence.

The fundamental benefits of a multi-server configuration are:

- Dedicated service providers.

  Web servers can be dedicated to providing PeopleSoft Portal and are insolated from other portions of PIA such as PeopleSoft Integration Gateway or PeopleBooks.

- Redundant service providers.

  Multiple web servers can be used to serve different aspects of PIA, providing load balancing and failover support.

- Distributed resources.

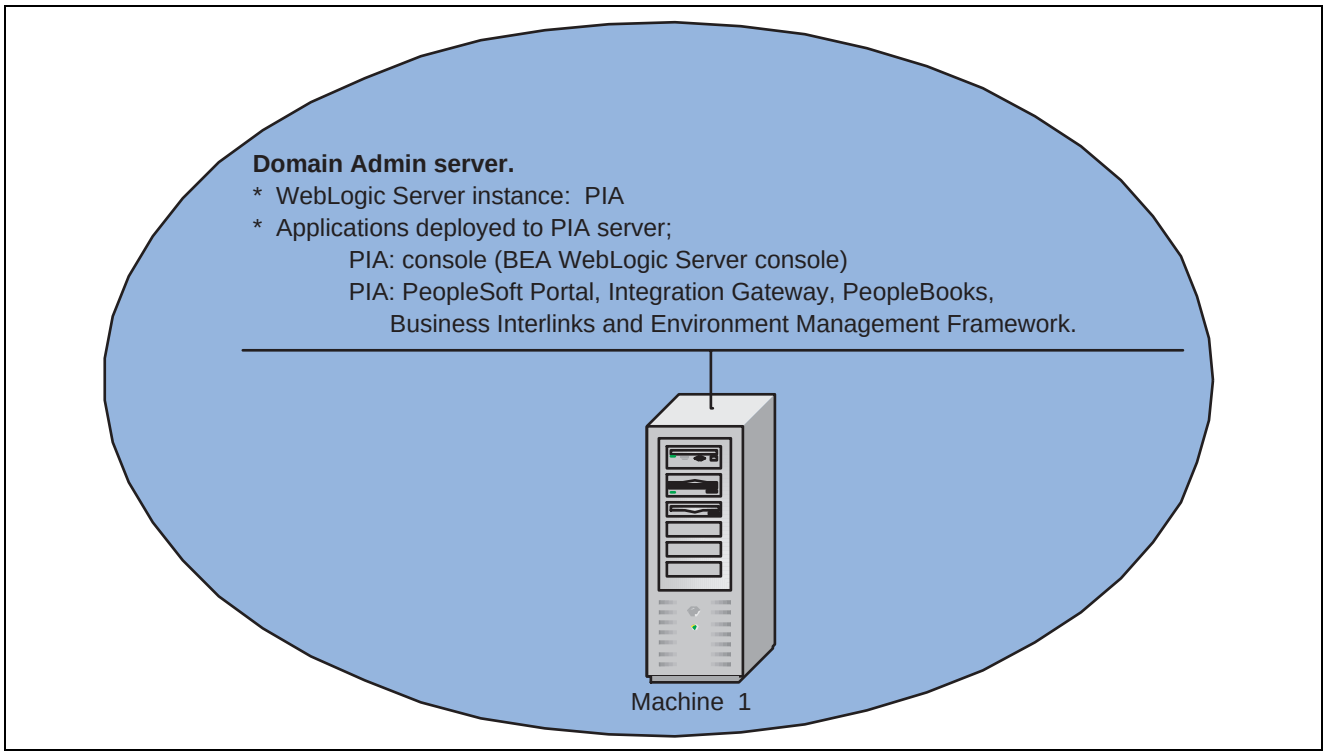
  Multiple web server machines can be used, each capable of serving different or redundant aspects of PIA.

- Centralized and replicated configuration.

  Master domain configuration is centralized and distributed server information is replicated locally.

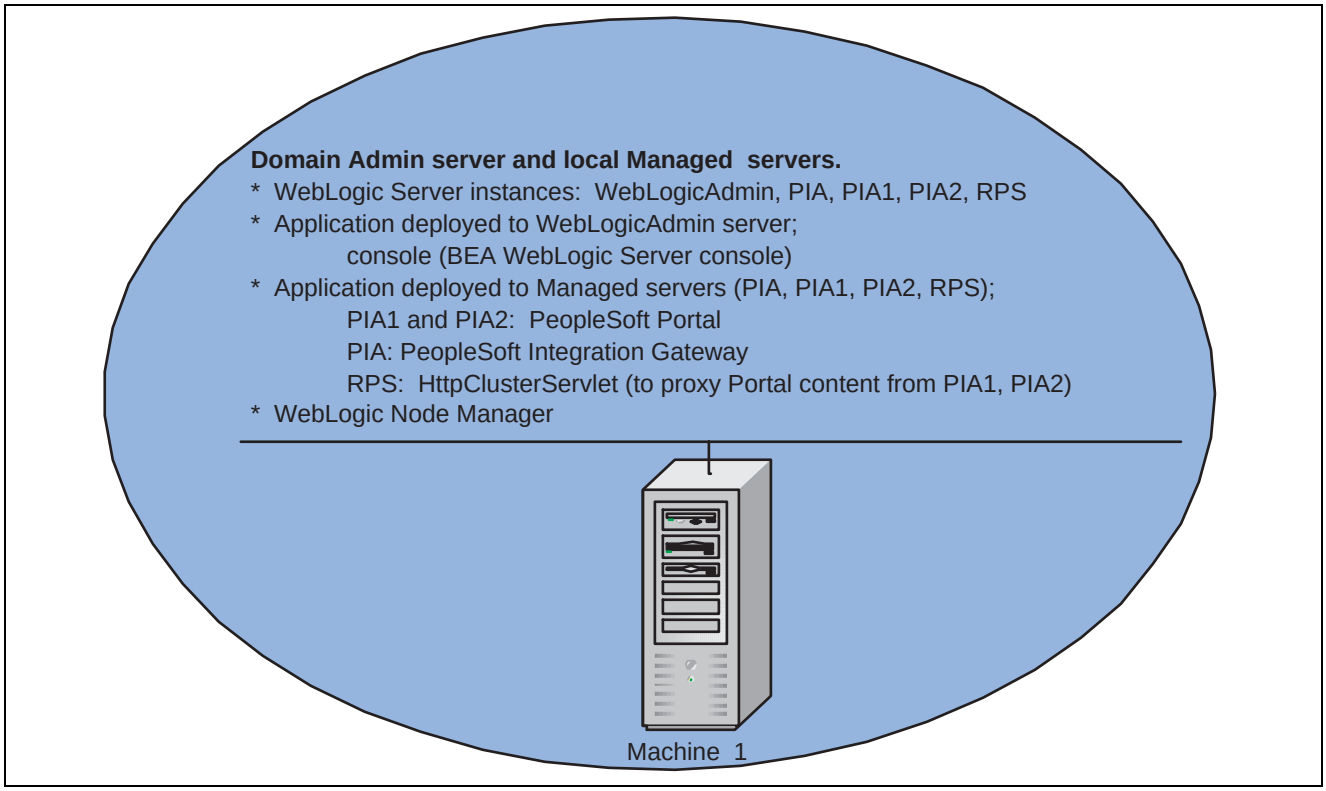
Domain Topology

A WebLogic domain can be as small and simple as the Single-server configuration or as broad and complex as the Multi-server configuration with distributed managed servers. In no way exhausting the possibilities, following are three sample layouts depicting each domain configuration.



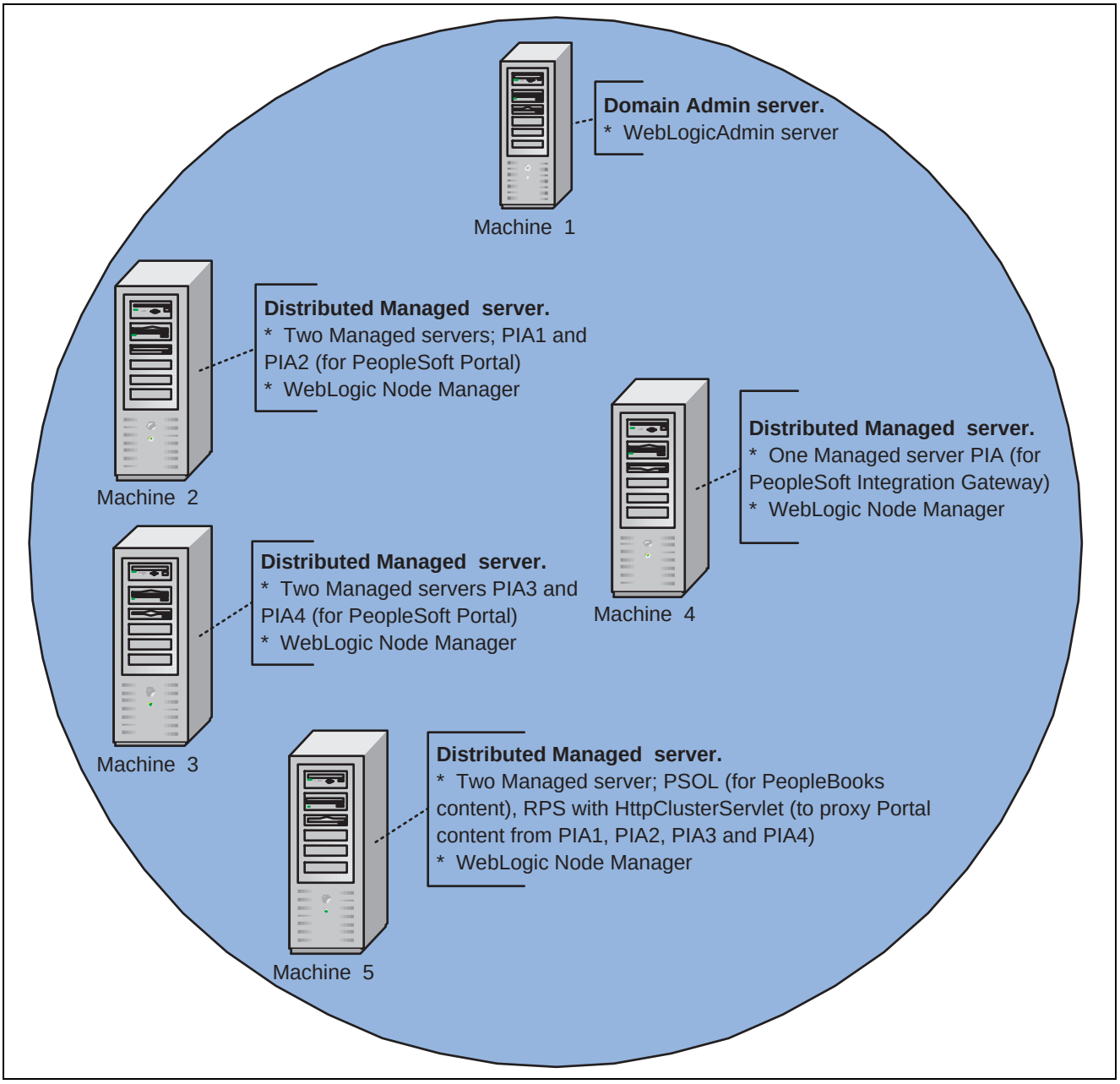
Single server — one machine

In a single-server configuration, the WebLogic domain's administration console and the J2EE components of PIA are all provided on a single instance of WebLogic Server.



Multi-server — one machine

In a multi-server configuration, multiple instances of WebLogic server are used, each contributing a specific function. The WebLogic console is provided on the domain's administration server, WebLogicAdmin, and the J2EE components of PIA are provided on individual or shared WebLogic managed servers.



Multi-server with distributed managed servers — multiple machines

In a Multi-server configuration with distributed managed servers, multiple instances of WebLogic server are used, each providing a specific function. The WebLogic console is provided on the domain's administration server, WebLogicAdmin, and the J2EE components of PIA are provided on individual or shared WebLogic managed servers. The only differentiating factor from the initial Multi serve configuration is that in this configuration two or more actual machines are used.

WebLogic Domain Directory Structure and Files

This section discusses:

- WebLogic domain directory structure.
- WebLogic domain file listing by type.
- J2EE application files.

WebLogic Domain Directory Structure

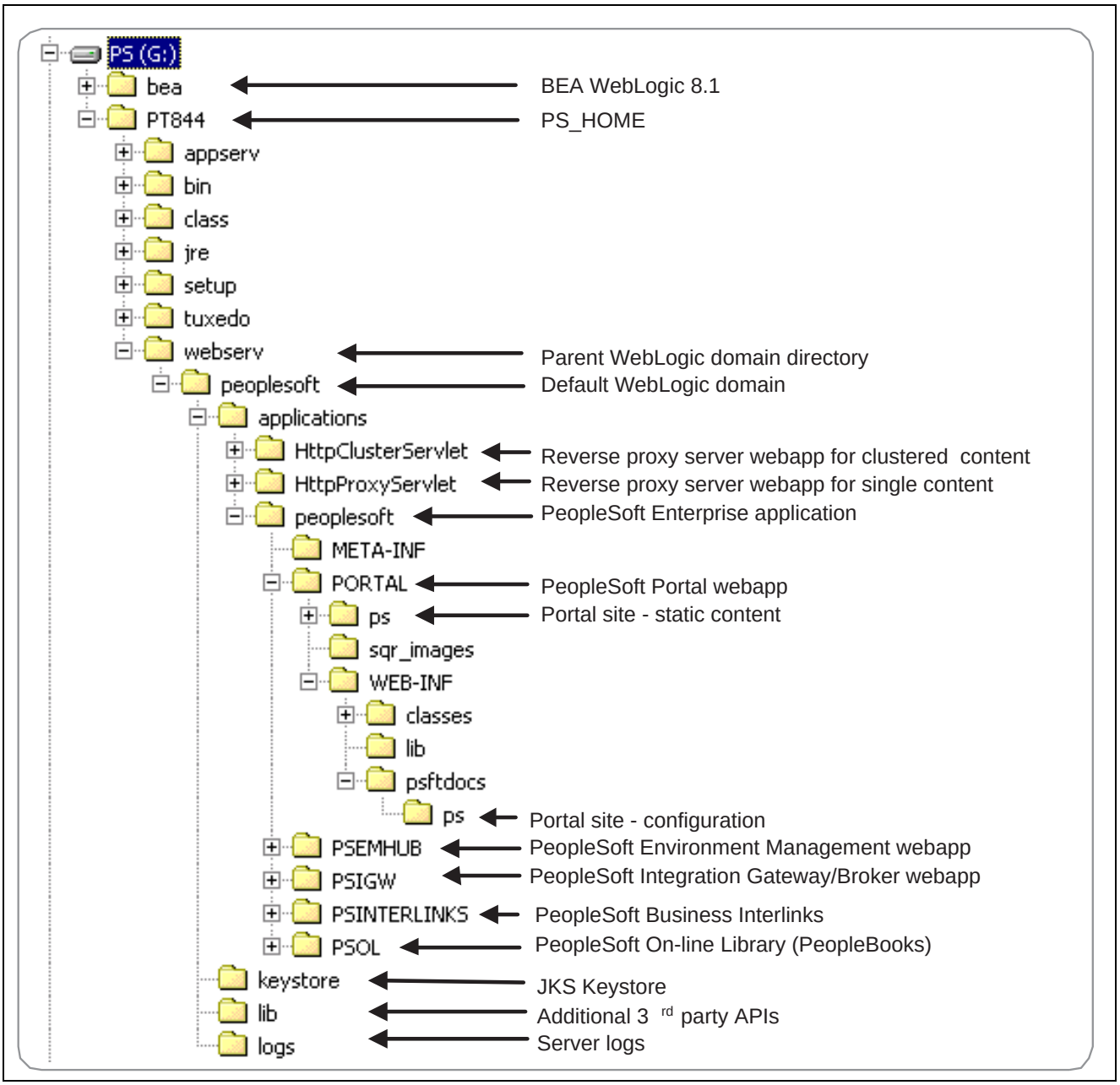
File and directory layout of PIA on WebLogic 8.1 does differ from that on prior versions of WebLogic. At a high level, the first difference is that the WebLogic domains are installed within your *PS_HOME* directory structure, rather than into WebLogic's directory structure as with prior versions of WebLogic Server.

The default home directories for PIA on different versions of WebLogic are as follows:

WebLogic Version	PIA Home Directory
WebLogic 5.1	*weblogic_home*\myserver\
WebLogic 6.1	*bea_home*\wlserver6.1\config\peopelsoft\
WebLogic 8.1	*ps_home*\webserv\peoplesoft\

This change was performed because managing the WebLogic configuration and PIA configuration similarly to the PeopleSoft application server and PeopleSoft Process Scheduler provided a clearer and more extendable architecture.

File and directory layout of PIA on WebLogic 8.1 differs from that of prior versions of PIA. The following illustration clarifies the directory structure layout of a PIA install on WebLogic Server 8.1.



WebLogic domain directory structure

WebLogic Domain File Listing by Type

Following are listings of all WebLogic domain files installed by the PIA setup, organized by file type. Where necessary, each table includes columns that indicate whether a given file is used in a single-server, multi-server, or distributed server configuration.

This listing does not include java classes or PIA configuration files. On UNIX an equivalent Bourne shell script is provided where a Windows script is listed.

The following table lists WebLogic server administration scripts.

Script	Single-Server	Multi-Server	Distributed Server	Description
setEnv.cmd	X	X	X	Use this script to set required environment variables for the WebLogic server, for example: CLASSPATH, PATH, UNIX Library Path, and JVM options.
startPIA.cmd	X			Use this script to start the WebLogic domain's administration server (the PIA server) in a single-server configuration. On Windows this starts WebLogic as a foreground process. On UNIX this starts WebLogic as a background process. Run the script with –help for usage.
startWebLogicAdmin. cmd		X		Use this script to start the WebLogic domain's administration server (the WebLogicAdmin server) in a multi-server configuration. On Windows this starts WebLogic as a foreground process. On UNIX this starts WebLogic as a background process. Run the script with –help for usage.
startManagedWebLogic.cmd		X	X	Use this script to start a WebLogic managed server. All WebLogic servers in a WebLogic domain other than the administration server are WebLogic managed servers. Run the script with –help for usage.
stopPIA.cmd	X			Use this script to stop the WebLogic PIA server. Run the script with –help for usage.
stopWebLogic.cmd		X	X	Use this script to stop WebLogic servers. Run the script with –help for usage.
InstallNTservicePIA.cmd	X			(Windows only) Use this script to install the WebLogic PIA server as a Windows service. The service name is *WebLogic_domain*-PIA. Run the script with –help for usage.
InstallNTservice.cmd		X	X	(Windows only) Use this script to install a WebLogic server as a Windows service. The service name is *WebLogic_domain-server_name*. Run the script with –help for usage.

Script	Single-Server	Multi-Server	Distributed Server	Description
uninstallNTServicePIA.cmd	X			(Windows only) Use this script to uninstall the WebLogic PIA server Windows service. Run the script with –help for usage.
uninstallNTService.cmd		X	X	(Windows only)Use this script to uninstall a WebLogic server Windows service. Run the script with –help for usage.
pskeymanager.cmd	X	X	X	Use this script to manage the JKS keystore used by WebLogic Server, which is in *WebLogic_domain*\keystore\pskey. SSL certificates for WebLogic Server are stored in this keystore. PeopleSoft Integration Gateway can also share this keystore. Run the script with –help for usage.
startWebLogicBuilder.cmd	X	X	X	Use this script to start WebLogic Builder, which is used to change local application deployment descriptors.
createThreadDump.cmd	X	X	X	Use this script to create a JVM Thread dump. Run the script with –help for usage.

The following table lists WebLogic server configuration files.

File	Single-Server	Multi-Server	Distributed Server	Description
config.xml	X	X		This file stores the WebLogic domain configuration, including information about server names, ports, IP addresses, webapps, and SSL. Edit these settings using the WebLogic administration console: http://*webserver*:*port*/console.
msi-config.xml		X	X	This is a version of config.xml that's copied for use with a distributed managed server configuration. It's automatically replicated from the original config.xml after a managed server successfully starts.
boot.properties	X	X	X	THis file contains the WebLogic system ID and password used for administering the WebLogic domain.
fileRealm.properties	X	X	X	This file is used by WebLogic's internal LDAP server for system administration.

File	Single-Server	Multi-Server	Distributed Server	Description
DefaultAuthenticatorInit.ldift	X	X	X	This file is used by WebLogic's internal LDAP server for system administration.
DefaultRoleMapperInit.ldift	X	X	X	This file is used by WebLogic's internal LDAP server for system administration.
SerializedSystemIni.dat	X	X	X	This file is used by WebLogic's internal LDAP server for system administration.

The following table lists PeopleSoft J2EE application scripts, which are all used with PeopleSoft Integration Broker, and can be used with every WebLogic server configuration.

Script	Description
BatchProjectExecutor.bat	Use this script for PeopleSoft Integration Broker batch EIP testing.
HashKeyGenerator.bat	Use this script to generate a hash key used for Integration Gateway playback.
MessageExport.bat	Use this PeopleSoft Integration Broker script for extracting transaction data from request and response data.
PSCipher.bat	Use this script for encrypting PeopleSoft Integration Broker passwords.
StartSendMaster.bat	This is a PeopleSoft Integration Broker test utility.

The following table lists miscellaneous files, which can be used with every WebLogic server configuration.

File	Description
Businterlink.txt	This file is used by PeopleSoft's Business Interlinks servlet for loading PeopleSoft libraries when needed.
piaInstallLog.xml	This is the PIA install log.

See Also

Enterprise PeopleTools 8.46 PeopleBook: PeopleSoft Integration Broker

J2EE Application Files

In addition to WebLogic domain configuration files, application descriptors are installed with the PeopleSoft J2EE enterprise application. The following table lists these descriptor files. The path shown for each file is relative to *PS_HOME*\webserv*WebLogic_domain*\applications\.

File	Description
peoplesoft\META-INF\MANIFEST.MF	Use this script to set required environment variables for the WebLogic server, for example: CLASSPATH, PATH, UNIX Library Path, and JVM options.
peoplesoft\META-INF\application.xml	This file contains a list of the webapps that comprise the PeopleSoft J2EE enterprise application.
peoplesoft\PORTAL\WEB-INF\web.xml	This file is the web application descriptor for the PORTAL webapp. It lists all of the servlets deployed as part of that application.
peoplesoft\PORTAL\WEB-INF\weblogic.xml	This file is the PORTAL web application extension descriptor. It specifies, among other things, the HTTP session cookie name, optional cookie domain, and context path of this application.
peoplesoft\PSIGW\WEB-INF\web.xml	This file is the web application descriptor for the PeopleSoft Integration Gateway (PSIGW) webapp. It lists all of the servlets deployed as part of that application.
peoplesoft\PSIGW\WEB-INF\weblogic.xml	This file is the PSIGW web application extension descriptor. It specifies the context path of this application.
peoplesoft\PSEMHUB\WEB-INF\web.xml	This file is the web application descriptor for the PeopleSoft Environment Framework (PSEMHUB) webapp. It lists all of the servlets deployed as part of that application.
peoplesoft\PSEMHUB\WEB-INF\weblogic.xml	This file is the PSEMHUB web application extension descriptor. It specifies the context path of this application.
peoplesoft\PSOL\WEB-INF\web.xml	This file is the web application descriptor for the PeopleSoft Online Library (PSOL) webapp (PeopleBooks). It lists all of the servlets deployed as part of that application.
peoplesoft\PSOL\WEB-INF\weblogic.xml	This file is the PSOL web application extension descriptor. It specifies the context path of this application.
peoplesoft\PSINTERLINKS\WEB-INF\web.xml	This file is the web application descriptor for the PeopleSoft Business Interlinks (PSINTERLINKS) webapp. It lists all of the servlets deployed as part of that application.
peoplesoft\PSINTERLINKS\WEB-INF\weblogic.xml	This file is the PSINTERLINKS web application extension descriptor. It specifies the context path of this application.
HttpProxtServlet\WEB-INF\web.xml	This file is the web application descriptor for the BEA WebLogic Server Reverse Proxy Server (RPS) webapp that's used to proxy content from a single WebLogic server. It lists all of the servlets deployed as part of that application.
HttpProxyServlet\WEB-INF\weblogic.xml	This file is the single-server RPS web application extension descriptor. It specifies the context path of this application.

File	Description
HttpClusterServlet\WEB-INF\web.xml	This file is the web application descriptor for the BEA WebLogic Server Reverse Proxy Server (RPS) webapp that's used to proxy content from a cluster of WebLogic servers. It lists all of the servlets deployed as part of that application.
HttpClusterServlet\WEB-INF\weblogic.xml	This file is the multi-server RPS web application extension descriptor. It specifies the context path of this application.

PIA Install and Reinstall Options

The PeopleSoft Internet Architecture (PIA) installer enables you to create a new WebLogic server domain or update a valid existing WebLogic domain. A valid domain is a domain built by the PIA installer in the *PS_HOME* directory that you specify.

Depending on which option you select, you're prompted for additional information relevant to that selection. When creating a new domain, you're prompted to select from three configuration types: Single-server, multi-server and distributed managed server. If you select to update an existing domain, you're prompted to indicate which domain you would like to update and what type of update you would like to perform, which are as follows:

- Install additional PeopleSoft site.

  This option is relevant only to the PeopleSoft PORTAL web application, and doesn't modify or revert any other configuration settings. Select this option to install only the necessary files for defining an additional PeopleSoft site onto an existing WebLogic configuration. The new site will be accessed using its name in the URL. A site named "CRM" would be accessed using a URL similar to http://mywebserver_machine/CRM. To reset or recreate an existing PeopleSoft site, simply enter that site's name as the site to create. On your web server, a PeopleSoft site is comprised of the following directories within the PORTAL web application:

  weblogic_domain\applications\peoplesoft\PORTAL*site**

  weblogic_domain\applications\peoplesoft\PORTAL\WEB-INF\psftdocs*site**

  See *Enterprise PeopleTools 8.46 PeopleBook: Internet Technology.*

- Redeploy PeopleSoft Internet Architecture.

  This selection affects all of the PIA web applications installed to the local WebLogic domain. Select this option to redeploy all of the class files and jar files that comprise web components of PIA. WebLogic Server configuration files, scripts and any existing PeopleSoft (PORTAL) sites are not overwritten, unless you specify an existing PeopleSoft site during this setup.

- Re-create WebLogic domain and redeploy PeopleSoft Internet Architecture.

  This option affects WebLogic Server configuration and all of the PIA web applications installed to the local WebLogic domain. Select this option to completely remove an existing WebLogic domain and create the newly specified PeopleSoft site.

  Warning! The entire WebLogic and PIA configuration in the specified WebLogic domain is deleted when you select this option.

- Deploy additional PeopleSoft application extensions.

This option is solely for use with PeopleSoft applications. PeopleSoft *application extensions* are provided with certain PeopleSoft applications, and this option enables you to deploy those extensions. Consult the installation documentation for your PeopleSoft application to see if this option is appropriate. PeopleTools does not use application extensions.

Administering a WebLogic Server Life Cycle

This section provides an overview of the WebLogic server life cycle and discusses how to:

- Start and stop single-server processes.
- Start and stop multi-server processes.
- Start and stop a distributed managed server.

See Also

Chapter 6, "Working with BEA WebLogic," Starting BEA WebLogic, page 99

Chapter 6, "Working with BEA WebLogic," Stopping BEA WebLogic, page 101

Understanding the WebLogic Server Life Cycle

You control a WebLogic server's life cycle primarily using a collection of scripts provided in that server's WebLogic domain directory. Each instance of a WebLogic server runs in an isolated Java Runtime Environment (JRE), regardless of whether you're testing with a single-server configuration or implementing a multi-server configuration for production. All scripts must be launched from the WebLogic domain directory; and provide usage syntax if run with –help.

Starting and Stopping Single-Server Processes

In a single-server configuration, there's only one server to administer: PIA. You can control the life cycle of the PIA server using scripts or in the WebLogic console.

Scripts

For all platforms:

startPIA	Use this script to start the WebLogic server locally.
stopPIA	Use this script to connect to a locally running WebLogic server and issue a shutdown command through WebLogic APIs.

For Windows only:

installNTservicePIA	Use this script to register the PIA WebLogic server as a Windows service that runs as a background process. the service is named as *WebLogicDomainName*-PIA, for example: peoplesoft-PIA.
uninstallNTservicePIA	Use this script to deregister the PIA Windows service.

WebLogic Console

A WebLogic server can also be shut down from its administration console. To shut down a WebLogic server in the WebLogic Server Console, sign in to the console at http://*webserver*/console and perform either of the following.

- In the navigation tree on the left, expand your domain, then Servers, then right click the PIA server and select Start/Stop this server.
- In the console, navigate to Servers, PIA, Control.

On the Start/Stop page, you can elect to perform either of the following two shutdown procedures:

Graceful shutdown of this server	This option enables transactions in progress to complete before shutting down the server. To terminate all HTTP sessions immediately, you can first select Ignore Sessions During Shutdown.
Force shutdown of this server	Immediately terminate all HTTP sessions and transactions in progress, and shut down the server.

Starting and Stopping Multi-Server Processes

In a Multi-server configuration, as the title implies there are multiple servers to administer. Controlling the life cycle of these servers can be done via scripts, the WebLogic console and the WebLogic Node Manager.

Scripts

For all platforms:

startWebLogicAdmin	Use this script to start the WebLogicAdmin server.
startManagedWebLogic	Use this script to start a WebLogic managed server. All of the servers defined in a multi-server domain, except the WebLogicAdmin server, are controlled as managed servers. For example, to start PIA1 as a managed server run startManagedWebLogic PIA1.
stopWebLogic	Use this script to connect to a locally running WebLogic server and issue a shutdown command using WebLogic APIs. A remote distributed managed server can be shut down using a local administration server.

For Windows only:

installNTservice.cmd	Use this script to register a WebLogic server as a Windows service that runs as a background process. The service is named as *WebLogicDomainName-ServerName*. For example, to define the PIA1 managed server as a Windows service, run installNTservice PIA1. To define the WebLogicAdmin server as a Windows service, simply run installNTservice.
UninstallNTservice.cmd	Use this script to deregister a WebLogic server that's defined as a Windows service.

Following are some important managed server considerations for scripts:

- For all platforms:

When starting a WebLogic managed server it will attempt to connect to its administration server. A managed server's administration server is specified either as a command line parameter when starting the managed server, or using the three administration server environment variables in setEnv, specifically ADMINSERVER_PROTOCOL, ADMINSERVER_PORT, and ADMINSERVER_HOSTNAME. The first time a managed server starts, it *must* connect to its administration server. If on subsequent startups the administration server is not available, the managed server starts up in Managed Server Independence (MSI) mode by using its locally replicated msi-config.xml. A managed server running in MSI mode can't be administered from a console, so this situation should only be encountered when it is imperative that the managed server be started even though the administration server is not running. Once the administration server is back online, running managed servers that were not previously known by the administration server to be running may be rediscovered using WebLogic's command line utility *java weblogic.Admin DISCOVERMANAGEDSERVER*, or you can just restart the managed server.

To use WebLogic's java command line utility classes run setEnv to set up your environment, then run *java weblogic.Admin* for usage.

- For Windows only:

  When running a WebLogic managed server as a Windows service, the managed server's administration server *must* be running. When installing a managed server as a Windows service, the managed server service can be configured to be dependent on its local administration server. To configure a managed server service to be dependent on its local admin server service use the *–depends* option of installNTservice.cmd when defining the Windows service for the managed server. In addition, when the administration server is also a Windows service, you must define it using the following command:

  installNTservice.cmd –delay *interval*

  Where *interval*is a period in milliseconds, for example *6000*. This allows the administration server sufficient time to start before the managed server starts.

WebLogic Server Console

A WebLogic server can also be shut down from its administration console. To shut down a WebLogic server in the WebLogic Server Console, sign in to the console of the WebLogicAdmin server (http://*webserver*:*port*/console) and perform either of the following:

- In the navigation applet on the left, expand your domain, then Servers, then right click the PIA server and select Start/Stop this server.
- In the console, navigate to Servers, PIA, Control.

On the Start/Stop page, you can elect to perform either of the following two shutdown procedures:

Graceful shutdown of this server	This option enables transactions in progress to complete before shutting down the server. To terminate all HTTP sessions immediately, you can first select Ignore Sessions During Shutdown.
Force shutdown of this server	Immediately terminate all HTTP sessions and transactions in progress, and shut down the server.

WebLogic Node Manager

The WebLogic Node Manager provides the ability to start a WebLogic managed server from the WebLogic Server Console. In addition, the console provides a way to automatically restart a failed server. As with all WebLogic servers, the WebLogic Node Manager runs isolated in its own JRE, and on Windows it can also run as a Windows service. The WebLogic Node Manager binds to a unique IP address and port at startup and accepts lifecycle commands from a WebLogic administration server.

Multiple WebLogic domains running on a singe machine can have its managed servers administered by a shared WebLogic Node Manager, as long as each WebLogic domain uses the same version of WebLogic.

The following table lists the WebLogic Node Manager files that are provided with WebLogic server, not the PIA install. These files are located in *BEA_HOME*\weblogic81\server\bin\, not your PeopleSoft created WebLogic domain directory.

File	Description
startNodeManager.cmd	Use this script to start the WebLogic Node Manager as a foreground process.
installNodeMgrSvc.cmd	Use this script to define the WebLogic Node Manager as a Windows service that runs as a background process. The service is called BEA WebLogic Platform 8.1 NodeManager.
uninstallNodeMgrSvc.cmd	Use this script to uninstall the WebLogic Node Manager as a Windows service.
nodemanager.properties	This is the WebLogic Node Manager configuration file.

Note. *BEA_HOME*\weblogic81\common\nodemanager\NodeManagerLogs\ is the default logs directory for WebLogic Node Manager.

To start a local WebLogic Managed server from the WebLogic Node Manager:

1. Start the WebLogic domain's WebLogic administration server.
2. Start the Node Manager of that domain's local BEA_HOME.

   If you're attempting to start a distributed managed server, you must start the Node manager on that distributed machine, otherwise the administration server, Node Manager and managed server are all on one machine.
3. Sign on to the WebLogic Server Console by entering the following URL in a browser:

   http://*webserver*:9999/console

   Where *webserver* is the domain of the WebLogic server.
4. Traverse the following in the navigation tree on the left:
   a. Expand your WebLogic domain (for example, peoplesoft).
   b. Expand Servers.
   c. Right click the server you would like to start and click 'Start/Stop this server...'.
5. On the console home page, within "Domain Configuration" \ "Network Configuration" select Servers, click the server you want to start, then select the Control tab.
6. Click the "Start this server..." link to start that server.

   If that text is not a hyperlink, either the WebLogic Node Manager is not running or the administration server was not able to communicate to the Node Manager.

A WebLogic Managed server started from the WebLogic Node Manager doesn't use Java options defined in setEnv, such as Java heap size. If you opt to use the WebLogic Node Manager, you should confirm or adjust the Java options in the WebLogic console of the administration server.

To adjust the Java options:

1. Expand your WebLogic domain (for example, peoplesoft).
2. Expand Servers.
3. Select the managed server that you would like to modify.
4. Select the first level Configuration tab
5. Select the second level Remote Start tab.
6. Update the Arguments field.
7. Click the Apply button.

Starting and Stopping a Distributed Managed Server

In a multi-server configuration, a distributed managed server is simply a managed server that isn't started from the same physical location as its domain's administration server. You can control the life cycle of these servers using scripts, the WebLogic Server Console and the WebLogic Node Manager.

See Appendix A, "BEA WebLogic 8.1 Managed Server Architecture," Starting and Stopping Multi-Server Processes, page 300.

In addition to the multi-server considerations, edit the following file:

BEA_HOME\weblogic81\common\nodemanager\nodemanager.hosts

Edit this file to include a list of IP addresses or host names of WebLogic administration servers from which the connection to the node manager will be accepted. In a typical configuration, there's only one such server

Tuning Performance and Monitoring Resources

Monitoring the performance of a WebLogic instance is primarily performed at the WebLogic console. This section discusses how to:

- Manage JVM heap size and execute thread usage.
- Monitor HTTP session count for PeopleSoft portal.

See Also

Chapter 6, "Working with BEA WebLogic," Using WebLogic Server 8.1 Console to Monitor PeopleSoft Sessions, page 102

Managing JVM Heap Size and Execute Thread Usage

Two of the main resource to monitor and tune with regards to WebLogic Server are the JVM memory heap size and the Execute thread count pool size. Monitoring and tuning both of these resources and more are describe below.

Monitoring JVM Heap and Execute Thread Usage

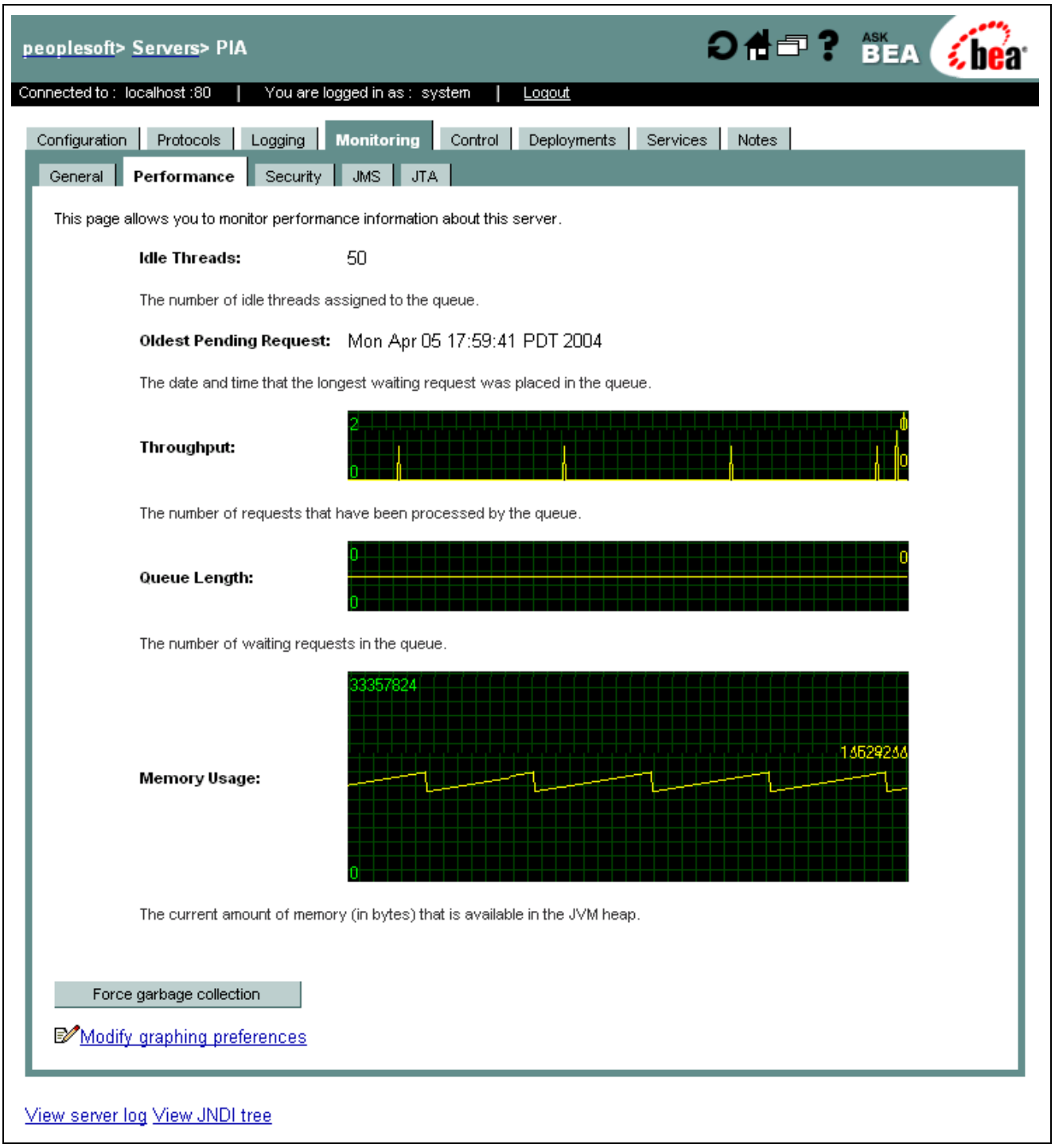
The JVM heap size is the amount of memory that a particular JRE (Java Runtime Environment) gives to the JVM (Java Virtual Machine) that it creates. The java.exe command on Windows, java on UNIX and beasvc.exe when running WebLogic as a Windows service is the JRE and the JVM exists within the JRE's memory space. The two primary sources for monitoring the amount of memory that is in use within a JVM are the WebLogic console and the WebLogic logs.

To monitor the amount of the JVM heap size available and in use:

1. Sign on to the WebLogic Server Console by entering the following URL in a browser:

   http://*webserver*:9999/console

   Where *webserver* is the domain of the WebLogic server.
2. Traverse the following in the navigation tree on the left:
   a. Expand your WebLogic domain (for example, peoplesoft).
   b. Expand Servers.
3. Select the server you intend to monitor (for example, PIA).
4. Select the first level Monitoring tab.
5. Select the second level Performance tab.



WebLogic Server Console performance monitoring

Idle Threads Displays the number of execute threads available for work. Having an excessive number of idle threads, such as 200, will waste a small amount of JVM resources.

Note. This value is static, so you must refresh the page to update it. Use the refresh icon at the top of the page.

Oldest Pending Request	Displays the date and time stamp of the oldest request in the queue. **Note.** This value is static, so you must refresh the page to update it. Use the refresh icon at the top of the page.
Throughput	Displays the number of requests over the displayed time frame that have been processed by an execute thread for the sampling time period. The top number on the left is the highest throughput, the bottom number on the left is the lowest throughput and the number on the right is the throughput for the current sample period.
Queue Length	Displays the number of requests over the displayed time frame that are queued up and waiting to be processed for the sampling time period. The top number on the left is the highest backlog, the bottom number on the left is the lowest backlog and the number on the right is the backlog for the current sample period.
Memory Usage	Displays the amount of JVM memory used in bytes over the displayed time frame. The top number on the left is the maximum memory available to the JVM (set using Java's –Xmx setting) and the number on the right is the memory in use for the current sample period.

Changing the JVM Heap Size

If you need to adjust any of the Java options, most commonly the JVM heap size, you must manually edit that WebLogic domain's local setEnv script. The two parameters, -Xms and -Xmx, control the JVM memory minimum and maximum heap size respectively.

Following are examples of the JVM heap size as specified in setEnv using the JAVA_OPTIONS_*OSplatform* environment variable. You only need to set the variables that correspond to the OS where the WebLogic server is running.

- JAVA_OPTIONS_WIN32="-server -Xms32m -Xmx200m -XX:MaxPermSize=128m"
- JAVA_OPTIONS_AIX="-Xms32m -Xmx200m"
- JAVA_OPTIONS_HPUX="-server -Xms32m -Xmx200m -XX:MaxPermSize=128m"
- JAVA_OPTIONS_LINUX="-Xms128m -Xmx256m"
- JAVA_OPTIONS_TRU64="-Xms32m -Xmx200m"
- JAVA_OPTIONS_SOLARIS="-server -Xms32m -Xmx200m -XX:MaxPermSize=128m"

Note. If you do adjust any of the Java options, most commonly the JVM heap size, you must restart WebLogic for these changes to take effect.

If you're running WebLogic Server as a Windows service you must rerun the installNTservice script to propagate this change into the Windows registry.

The WebLogic Node Manager does not use the Java options set in setEnv, but instead uses Java options set from the WebLogic console.

To modify the Java options for a WebLogic instance started via the WebLogic Node Manager:

1. Sign on to the WebLogic Server Console by entering the following URL in a browser:

   http://*webserver*:9999/console

Where *webserver* is the domain of the WebLogic server.

2. Traverse the following in the navigation tree on the left:
    a. Expand your WebLogic domain (for example, peoplesoft).
    b. Expand Servers.
3. Select the server you intend to modify (for example, PIA).
4. Select the first level Configuration tab.
5. Select the second level Remote Start tab.
6. Update the Arguments field.
7. Click Apply.

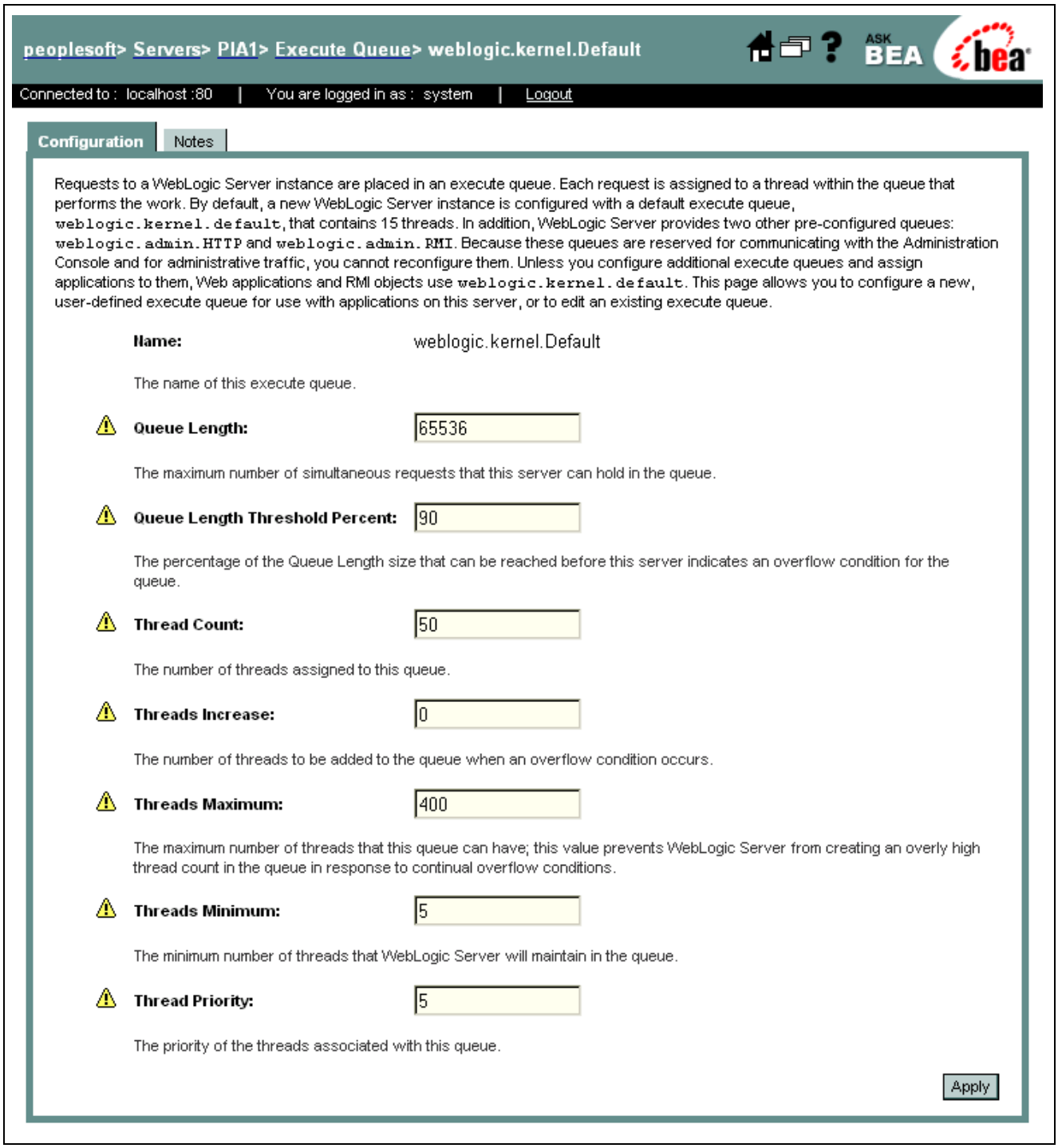
Changing the Execute Thread Pool Size

To change the size of the Execute thread pool used by PeopleSoft applications for a particular WebLogic instance:

1. Sign on to the WebLogic Server Console by entering the following URL in a browser:

    http://*webserver*:9999/console

    Where *webserver* is the domain of the WebLogic server.
2. Traverse the following in the navigation tree on the left:
    a. Expand your WebLogic domain (for example, peoplesoft).
    b. Expand Servers.
3. Right-click the server you intend to modify (for example, PIA).
4. Select View Execute Queues.
5. Click the queue that your server is using, most likely weblogic.kernel.Default.
6. Adjust the queue size settings to fit your needs.
7. Click Apply.



WebLogic Server Console execute queue configuration

See Also

Chapter 6, “Working with BEA WebLogic,” Adjusting the JVM Heap Size, page 125

Monitoring HTTP Session Count for PeopleSoft Portal

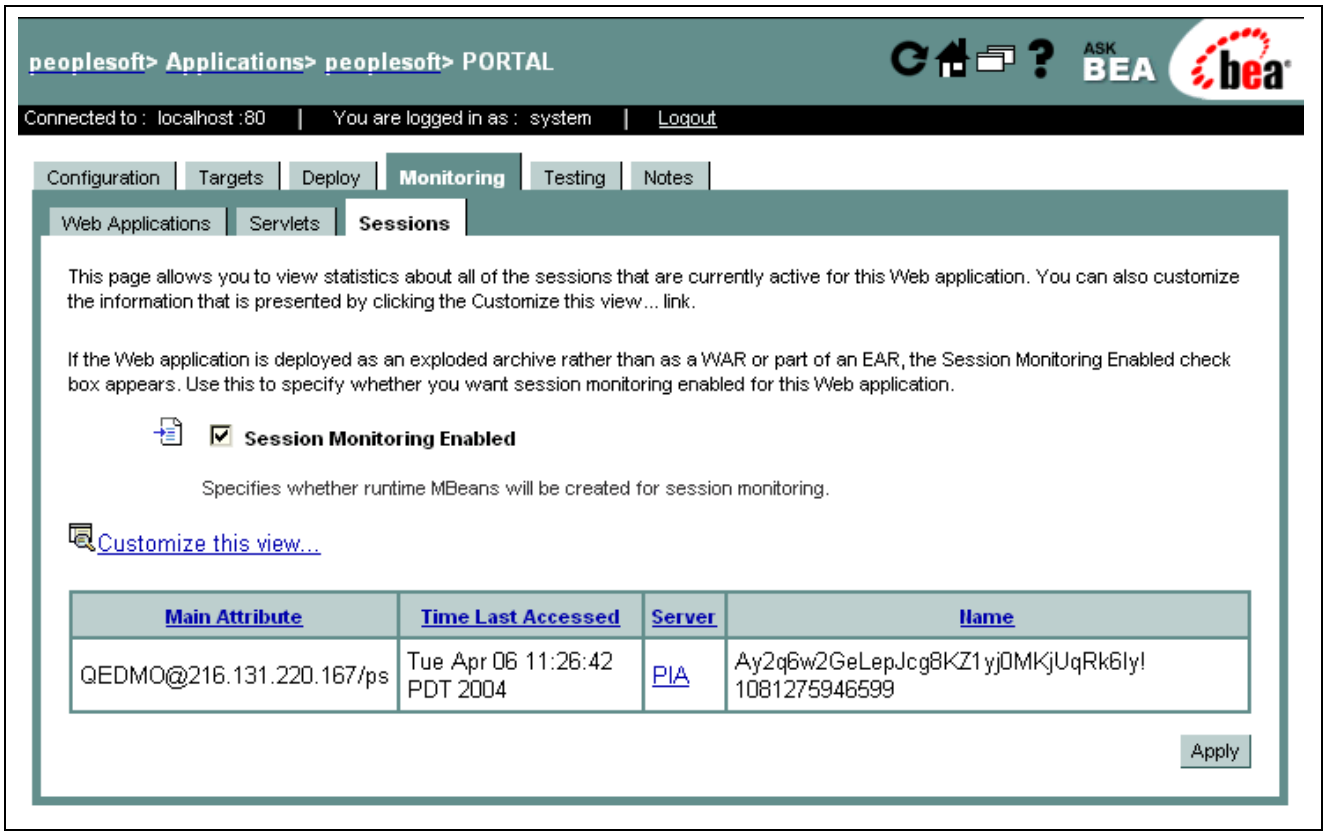
In addition to memory and thread usage, it's also possible to monitor the number of established HTTP sessions used in conjunction with the PeopleSoft PORTAL application. This number, although not necessarily directly related to current performance, is an indicator of the following performance factors:

- JVM memory used to store HTTP session data.
- Current number of logged on clients.
- Peak number of logged on clients.
- Idle time of logged on clients.

To monitor HTTP sessions:

1. Sign on to the WebLogic Server Console.
2. Traverse the following in the navigation tree on the left:
    a. Expand peoplesoft.
    b. Expand Deployments.
    c. Expand Applications.
    d. Expand peoplesoft.
3. Select PORTAL.
4. Select the Monitoring tab.
5. Select the Sessions tab.
6. Ensure that the Session Monitoring Enabled check box is selected, which enables it for the current WebLogic instance.

    If the check box was previously cleared, restart WebLogic Server to apply the new setting.
7. Click the Refresh button at the top of the console page.



WebLogic Server Console PORTAL HTTP session monitoring

With HTTP session monitoring enabled, the information displayed will be users' PeopleSoft ID, their workstation name or IP address, the PeopleSoft site they are logged onto and the time of their last request.

Note. Established HTTP sessions remain on the web server until users sign out of PeopleSoft or until the sessions time out. Simply closing the browser does *not* sign a PeopleSoft user out. You can control HTTP session time out settings from a site's web profile in the PeopleSoft database.

See Also

Enterprise PeopleTools 8.46 PeopleBook: Internet Technology, "Configuring the Portal Environment," Configuring Web Profiles

Changing Configuration Settings

This section provides an overview of the WebLogic server configuration files, and discusses how to:

- Change the WebLogicAdmin server's listen ports.
- Change application and server deployment targets.

Understanding the WebLogic Server Configuration Files

WebLogic server configuration settings are stored in a collection of files, primarily the setEnv script, config.xml, and the web.xml and weblogic.xml for each webapp.

- SetEnv contains statically and dynamically defined environment variables. It's called from all of the WebLogic administration scripts to assist in building the Java command line. You modify this file using a text editor.
- Config.xml contains server runtime settings, such as the HTTP port. You modify this file using the WebLogic Server Console.
- Web.xml and weblogic.xml, which are located in the WEB-INF directory of each webapp, are web application descriptors and contain settings relevant to their application.

Changing the WebLogicAdmin Server's Listen Ports

In the multi-server configuration, several parameters are set based on the environment detected and on delivered defaults. One such parameter is the HTTP port of the WebLogicAdmin server. By default the WebLogicAdmin server's HTTP listen port is 9999.

To change this value:

1. Start the WebLogicAdmin server via the startWebLogicAdmin script
2. Sign on to the WebLogic Server Console by entering the following URL in a browser:

   http://*webserver*:9999/console

   Where *webserver* is the domain of the WebLogic server.
3. Navigate to Servers, WebLogicAdmin, Configuration, General.
4. Modify the value of the Listen Port field.
5. Click Apply.
6. Restart the WebLogic server.

Note. If you can't initially start the server due to a port conflict, you can manually edit the value of the ListenPort parameter in that domain's config.xml file. Creating a backup of config.xml is recommended before manually changing this file.

After changing the ListenPort value in your domain's config.xml, either directly or using the console, you must also update your setEnv script. Update the ADMINSERVER_PORT environment variable to reflect the new HTTP port. This setting is used by the stopWebLogic and startManagedWebLogic scripts as the default administration server HTTP port.

Changing Application and Server Deployment Targets

With WebLogic, J2EE applications are targeted to any combination of WebLogic servers and WebLogic clusters. A WebLogic cluster is a logical grouping of servers, generally all providing the same application, though that's not a requirement. To change the servers or clusters that an application is targeted and deployed to, sign on to the WebLogic Server Console and update the application's target assignments. You can view and modify application and server target assignments on the Deployments, Applications tabs, and on the Deployments tab for each server.

Following is an example of how to change the target assignments of the PeopleSoft Integration Gateway (PSIGW) web application so it's the only application targeted to the PIA server, and is the sole application on that instance.

To change the target assignments of the PeopleSoft Integration Gateway web application:

1. Sign on to the WebLogic Server Console.

2. Traverse the following in the navigation tree on the left:
    a. Expand peoplesoft.
    b. Expand Deployments.
    c. Expand Applications.
    d. Expand peoplesoft.
3. Select PSIGW.
4. Select the Targets tab.
5. In the Clusters section, clear the peoplesoftCluster check box.
6. Click Apply.
7. In the navigation tree, select PORTAL.
8. Select the Targets tab.
9. In the Independent Servers grid, clear the check box for targeting the PORTAL webapp to this server.
10. Click Apply.
11. Repeat steps 7 to 10 for the PSEMHUB, PSINTERLINKS and PSOL web applications.

To deploy an application to a cluster, target the server to the cluster and target the application to the cluster.

Applying an Example Single-Server Configuration

This section provides overviews of the basic machine configuration, PIA setup prompts, configuration file settings, and unprompted configuration file settings, and discusses how to start the weblogic server.

Basic Machine Configuration

This example single-server setup is performed on a machine with the following configuration:

- Operating system: Windows
- Hostname: DBROWN040403
- PeopleSoft home directory: C:\PT845
- BEA WebLogic Server home directory: C:\BEA

PIA Setup Prompts

Following are the PIA setup prompts and corresponding values appropriate to building the example single-server domain.

Note. The Prompt # column doesn't reflect actual step numbers during setup. The prompt numbers simply provide a reference for subsequent discussion of the configuration file settings.

Prompt #	Prompted Field	Value
1	PeopleSoft CD install location (PS_HOME).	*C:\PT845*
2	BEA WebLogic Server home directory.	*C:\BEA*
3	WebLogic domain admin ID.	*system*
4	WebLogic domain admin password.	*password*
5	WebLogic domain name.	*peoplesoft*
6	WebLogic domain type.	*Single Server*
7	Portal website name.	*ps*
8	Application Server hostname.	*DBROWN040403*
9	Application Server JSL port.	*9845*
10	HTTP Listen Port.	*80*
11	HTTPS Listen Port.	*443*
12	Authentication Token domain.	*.peoplesoft.com*
13	Web profile name.	*DEV*
14	Web profile user ID.	*PTWEBSERVER*
15	Web profile password.	*PTWEBSERVER*
16	Report Repository location.	*C:\psreports*

Configuration File Settings

As a result of the specified setup response values, several configuration files are updated. Following are the corresponding configuration file updates by prompt number.

Prompt #	Configuration File	Setting
1	setEnv.cmd	SET PS_HOME=C:/PT845
1	config.xml	`<ServerStart Arguments="-Xms64m -Xmx128m -Dps_home=C:/PT845"⇒ ClassPath="c:/bea/weblogic81\server\lib\weblogic_sp.jar;⇒ C:/PT845\webserv\peoplesoft\lib\ps_patch.jar;⇒ c:/bea/weblogic81\server\lib\weblogic.jar"⇒ JavaHome="c:\bea\jdk142"⇒ RootDirectory="C:\PT845\webserv\peoplesoft"...`
2	setEnv.cmd	SET BEA_HOME=c:\bea SET WL_HOME=c:\bea\weblogic81

Prompt #	Configuration File	Setting
2	config.xml	<ServerStart BeaHome="c:\bea"⇒ ClassPath="c:/bea/weblogic81\server\lib\weblogic_sp.jar;⇒ C:/PT845\webserv\peoplesoft\lib\ps_patch.jar;⇒ c:/bea/weblogic81\server\lib\weblogic.jar"...
3	config.xml	<ServerStart Username="system"...
3	boot.properties	(encrypted)
4	boot.properties	(encrypted)
5	setEnv.cmd	SET DOMAIN_NAME=peoplesoft
5	config.xml	<Domain Name="peoplesoft"...
6	n/a (Domain configuration is based on this selection.)	
10	setEnv.cmd	SET ADMINSERVER_PORT=80
10	config.xml	<Server Name="PIA" ListenPort="80"...
11	config.xml	<Server Name="PIA" SSLListenPort="443"...
12	..\applications\peoplesoft \PORTAL\WEB-INF \weblogic.xml	<session-param> <param-name>CookieDomain</param-name> <param-value>.peoplesoft.com</param-value> </session-param>

Note. The prompts numbered 7, 8, 9, 13, 14, 15, and 16 don't directly affect the WebLogic configuration.

Unprompted Configuration File Settings

In addition to the configuration file changes based on the setup response values, additional automatic configuration changes are performed based on the domain type you selected.

Configuration File	Setting
setEnv.cmd	SET ADMINSERVER_HOSTNAME=DBROWN040403 SET ADMINSERVER_SERVERNAME=PIA
config.xml	<Server Name="PIA"… <SSL PrivateKeyAlias=" DBROWN040403"…

Starting the Weblogic Server

To start the WebLogic server created in this example, use the scripts created in the generated WebLogic domain directory of *PS_HOME*\webserv\peoplesoft\. To start this WebLogic server as a foreground process, run startPIA.cmd. A successful startup produces the following messages, which indicate that the server has started, that it has bound to all local IP addresses, and that it's listening on ports 80 and 443 for HTTP and HTTPS traffic respectively.

```
<Feb 27, 2004 11:23:41 AM PST> <Notice> <WebLogicServer> <BEA-000355>⇒
 <Thread "ListenThread.Default" listening on port 80, ip address *.*>
<Feb 27, 2004 11:23:41 AM PST> <Notice> <WebLogicServer> <BEA-000355>⇒
 <Thread "SSLListenThread.Default" listening on port 443, ip address *.*>
<Feb 27, 2004 11:23:41 AM PST> <Notice> <WebLogicServer> <BEA-000330>⇒
 <Started WebLogic Admin Server "PIA"⇒
 for domain "peoplesoft" running in Production Mode>
<Feb 27, 2004 11:23:41 AM PST> <Notice> <WebLogicServer> <BEA-000360>⇒
 <Server started in RUNNING mode>
```

Applying an Example Multi-Server Configuration

This section provides overviews of the basic machine configurations, PIA setup prompts for multi-server configuration, configuration file settings for multi-server configuration, unprompted multi-server configuration file settings, PIA setup prompts for distributed managed server configuration, configuration file settings for distributed managed server configuration, and unprompted distributed server configuration file settings, and discusses how to start the weblogic servers.

Basic Machine Configurations

An extended Multi-server configuration requires performing two PIA installs, once to create the Multi-Server configuration and a second time to create the Distributed Managed Server.

This Multi-Server installation was performed on a machine with the following configuration.

- Operating system: Windows
- Hostname: DBROWN040403
- PeopleSoft home directory: C:\PT845
- BEA WebLogic Server home directory: C:\BEA

This Distributed-Server installation was performed on a machine with the following configuration.

- Operating system: RedHat Linux
- Hostname: PT-LINUX01
- PeopleSoft home directory: /apps/pt845
- BEA WebLogic Server home directory: /apps/bea

PIA Setup Prompts for Multi-Server Configuration

Following are the PIA setup prompts and corresponding values appropriate to building the example extended multi-server domain.

Note. The Prompt # column doesn't reflect actual step numbers during setup. The prompt numbers simply provide a reference for subsequent discussion of the configuration file settings.

Prompt #	Prompted Field	Value
1	PeopleSoft CD install location (PS_HOME).	*C:\PT845*
2	BEA WebLogic Server home directory.	*C:\BEA*
3	WebLogic domain admin ID.	*system*
4	WebLogic domain admin password.	*password*
5	WebLogic domain name.	*peoplesoft*
6	WebLogic domain type.	*Multi Server*
7	Portal website name.	*ps*
8	Application Server hostname.	*DBROWN040403*
9	Application Server JSL port.	*9845*
10	HTTP Listen Port.	*80*
11	HTTPS Listen Port.	*443*
12	Authentication Token domain.	*.peoplesoft.com*
13	Web profile name.	*DEV*
14	Web profile user ID.	*PTWEBSERVER*
15	Web profile password.	*PTWEBSERVER*
16	Report Repository location.	*C:\psreports*

Configuration File Settings for Multi-Server Configuration

As a result of the specified setup response values, several configuration files are updated. Following are the corresponding configuration file updates by prompt number.

Prompt #	Configuration File	Setting
1	setEnv.cmd	SET PS_HOME=C:/PT845
1	config.xml **Note.** This setting occurs 6 times, once for each managed server.	`<ServerStart Arguments="-Xms64m -Xmx128 -Dps_home=c:/pt845"⇒ ClassPath="c:/bea/weblogic81/server/lib/weblogic_sp.jar;⇒ c:/pt845/webserv/peoplesoft/lib/ps_patch.jar;⇒ c:/bea/weblogic81/server/lib/weblogic.jar"⇒ JavaHome="c:/bea/jdk142"⇒ RootDirectory="c:/PT845/webserv/peoplesoft"...`
2	setEnv.cmd	SET BEA_HOME=c:\bea SET WL_HOME=c:\bea\weblogic81

Prompt #	Configuration File	Setting
2	config.xml **Note.** This setting occurs 6 times, once for each managed server.	<ServerStart BeaHome="c:/bea"⇒ ClassPath="c:/bea/weblogic81/server/lib/weblogic_sp.jar;⇒ c:/pt845/webserv/peoplesoft/lib/ps_patch.jar;⇒ c:/bea/weblogic81/server/lib/weblogic.jar"...
3	config.xml **Note.** This setting occurs 6 times, once for each managed server.	<ServerStart Username="system"...
3	boot.properties	(encrypted)
4	boot.properties	(encrypted)
5	setEnv.cmd	SET DOMAIN_NAME=peoplesoft
5	config.xml	<Domain Name="peoplesoft"...
6	n/a (Domain configuration is based on this selection.)	
10	setEnv.cmd	SET ADMINSERVER_PORT=80
10	config.xml **Note.** This setting occurs 3 times, once each for the PIA, PIA1, and PIA2 managed servers.	<Server Name="PIA" ListenPort="80"... <Server Name="PIA1" ListenPort="80"... <Server Name="PIA2" ListenPort="80"...
11	config.xml **Note.** This setting occurs 3 times, once each for the PIA, PIA1, and PIA2 managed servers.	<Server Name="PIA" SSLListenPort="443"... <Server Name="PIA1" SSLListenPort="443"... <Server Name="PIA2" SSLListenPort="443"...
12	..\applications\peoplesoft\PORTAL\WEB-INF\weblogic.xml	<session-param> <param-name>CookieDomain</param-name> <param-value>.peoplesoft.com</param-value> </session-param>

Note. The prompts numbered 7, 8, 9, 13, 14, 15, and 16 don't directly affect the WebLogic configuration.

Unprompted Multi-Server Configuration File Settings

In addition to the configuration file changes based on the setup response values, additional automatic configuration changes are performed based on the domain type you selected.

Configuration File	Setting
setEnv.cmd	SET ADMINSERVER_PORT=9999 SET ADMINSERVER_HOSTNAME=DBROWN040403 SET ADMINSERVER_SERVERNAME= WebLogicAdmin
config.xml	<Server Name="PIA"… <SSL PrivateKeyAlias=" DBROWN040403"…

PIA Setup Prompts for Distributed Managed Server Configuration

Following are the PIA setup prompts and corresponding values appropriate to building the example extended multi-server domain.

Note. The Prompt # column doesn't reflect actual step numbers during setup. The prompt numbers simply provide a reference for subsequent discussion of the configuration file settings.

Prompt #	Prompted Field	Value
1	PeopleSoft CD install location (PS_HOME).	*/apps/pt845*
2	BEA WebLogic Server home directory.	*/apps/bea*
3	WebLogic domain admin ID.	*system*
4	WebLogic domain admin password.	*password*
5	WebLogic domain name.	*peoplesoft*
6	WebLogic domain type.	*Distributed Managed Server*
7	Portal website name.	*ps*
8	Application Server hostname.	*PT-LINUX01*
9	Application Server JSL port.	*9845*
10	HTTP Listen Port.	*80*
11	HTTPS Listen Port.	*443*
12	Authentication Token domain.	*.peoplesoft.com*
13	Web profile name.	*DEV*
14	Web profile user ID.	*PTWEBSERVER*
15	Web profile password.	*PTWEBSERVER*
16	Report Repository location.	*/apps/pt845/psreports*

Configuration File Settings for Distributed Server Configuration

As a result of the specified setup response values, several configuration files are updated. Following are the corresponding configuration file updates by prompt number.

Prompt #	Configuration File	Setting
1	setEnv.cmd	SET PS_HOME=/apps/pt845
2	setEnv.cmd	SET BEA_HOME=/apps/bea SET WL_HOME=/apps/bea/weblogic81
3	config.xml **Note.** This setting occurs 6 times, once for each managed server.	<ServerStart Username="system"...
3	boot.properties	(encrypted)
4	boot.properties	(encrypted)
5	setEnv.cmd	SET DOMAIN_NAME=peoplesoft
6	n/a (Domain configuration is based on this selection.)	
10	setEnv.cmd	SET ADMINSERVER_PORT=80
10	config.xml **Note.** This value is set in the config.xml file of the administration server (where the multi-server configuration was set up).	<Server ListenPort="80"...
11	config.xml **Note.** This value is set in the config.xml file of the administration server (where the multi-server configuration was set up).	<Server SSLListenPort="443"...

12	../applications/peoplesoft/PORTAL/WEB-INF /weblogic.xml	`<session-param>`
		`  <param-name>CookieDomain⇒`
		`</param-name>`
		`  <param-value>.peoplesoft.com⇒`
		`</param-value>`
		`</session-param>`

Note. The prompts numbered 7, 8, 9, 13, 14, 15, and 16 don't directly affect the WebLogic configuration.

Unprompted Distributed Server Configuration File Settings

In addition to the configuration file changes based on the setup response values, additional automatic configuration changes are performed based on the domain type you selected.

Configuration File	Setting
setEnv.cmd	SET ADMINSERVER_PORT= SET ADMINSERVER_HOSTNAME= SET ADMINSERVER_SERVERNAME= WebLogicAdmin
config.xml	<Server Name="PIA"… <SSL PrivateKeyAlias="PT_ LNX01"...

Starting the WebLogic Servers

To start the WebLogic Servers created in this example, use the scripts created in the generated WebLogic domain directory of *PS_HOME*\webserv\peoplesoft.

Starting the Domain Administration Server

To start the domain administration server, WebLogicAdmin, run the following command:

c:\pt845\webserv\peoplesoft\startWebLogicAdmin.cmd

A successful startup of this server produces the following messages, which indicate that the server has started, that it has bound to all local IP addresses, and that it's listening on port 9999 for HTTP traffic.

```
<Feb 27, 2004 11:23:41 AM PST> <Notice> <WebLogicServer> <BEA-000355>⇒
 <Thread "ListenThread.Default" listening on port 9999, ip address *.*>
<Feb 27, 2004 11:23:41 AM PST> <Notice> <WebLogicServer>⇒
 <BEA-000330> <Started WebLogic Admin Server "WebLogicAdmin"⇒
 for domain "peoplesoft" running in Production Mode>
<Feb 27, 2004 11:23:41 AM PST> <Notice> <WebLogicServer>⇒
 <BEA-000360> <Server started in RUNNING mode>
```

Starting the Local Managed Server

To start the local managed server (PIA) on DBROWN040403, the same machine where the WebLogicAdmin server is running, run the following command:

c:\pt845\webserv\peoplesoft\startManagedWebLogic.cmd PIA

A successful startup of this server produces the following messages, which indicate that the server has started, that it has bound to all local IP addresses, and that it's listening on ports 80 and 443 for HTTP and HTTPS traffic respectively.

```
<Feb 27, 2004 2:17:04 PM PST> <Notice> <WebLogicServer> <BEA-000355>⇒
 <Thread "ListenThread.Default" listening on port 80, ip address *.*>
<Feb 27, 2004 2:17:04 PM PST> <Notice> <WebLogicServer> <BEA-000355>⇒
 <Thread "SSLListenThread.Default" listening on port 443, ip address *.*>
<Feb 27, 2004 2:17:04 PM PST> <Notice> <WebLogicServer>⇒
 <BEA-000330> <Started WebLogic Managed Server "PIA"⇒
 for domain "peoplesoft" running in Production Mode>
<Feb 27, 2004 2:17:04 PM PST> <Notice> <WebLogicServer>⇒
 <BEA-000360> <Server started in RUNNING mode>
```

Starting a Remote Distributed Managed Server

Prior to starting a remote distributed managed server, which in this example is a WebLogic Reverse Proxy Server (RPS), you must modify the HttpProxyServlet web application's web.xml file to specify the address of the back-end WebLogic instance from which content is proxied. The HttpProxyServlet webapp is the default webapp for the RPS server, and is used when proxing content from a single back-end WebLogic server.

To modify HttpProxyServlet's web.xml you can use BEA's graphical WebLogic Builder, or simply edit the HttpProxyServlet's web.xml. To use WebLogic Builder, launch the startWebLogicBuilder script from within your WebLogic domain directory, and either specify *applications/HttpProxyServlet* as a command line parameter, or once in WebLogic Builder select File/Open and open the HttpProxyServlet application. To edit the web.xml file manually, the location of the file is *weblogic_domain*\applications\HttpProxyServlet\WEB-INF\web.xml.

The parameter values to adjust are the WebLogicHost and WebLogicPort init-params for the HttpProxyServlet servlet. Following the relevant part of a properly configured web.xml.

```
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD⇒
 Web Application 2.3//EN" "http://java.sun.com/dtd/web-app_2_3.dtd">
 <web-app>
    <display-name>HttpProxyServlet</display-name>
    <description>HttpProxyServlet</description>
 <servlet>
    <servlet-name>HttpProxyServlet</servlet-name>
    <display-name>HttpProxyServlet</display-name>
 <servlet-class>weblogic.servlet.proxy.HttpProxyServlet</servlet-class>
    <init-param>
       <param-name>WebLogicHost</param-name>
       <param-value>DBROWN040403</param-value>
       <description>Hostname of WebLogic PIA/PORTAL server.</description>
    </init-param>
    <init-param>
       <param-name>WebLogicPort</param-name>
       <param-value>80</param-value>
       <description>HTTP listen port of WebLogic PIA/PORTAL server.</description>
    </init-param>
    . . .
 . . .
```

To start the remote managed server (RPS) on PT-LINUX01, the machine where the distributed managed server setup was performed, run the following command:

/apps/pt845/webserv/peoplesoft/startManagedWebLogic.cmd RPS http://dbrown040403:9999

A successful startup of this server produces the following messages, which indicate that the managed server started and was able to connect to the administration server, that it has bound to all local IP addresses, and that it's listening on ports 80 and 443 for HTTP and HTTPS traffic, respectively.

```
<Feb 27, 2004 2:17:04 PM PST> <Notice> <WebLogicServer> <BEA-000355>⇒
 <Thread "ListenThread.Default" listening on port 8080, ip address *.*>
<Feb 27, 2004 2:17:04 PM PST> <Notice> <WebLogicServer> <BEA-000355>⇒
 <Thread "SSLListenThread.Default" listening on port 8443, ip address *.*>
<Feb 27, 2004 2:17:04 PM PST> <Notice> <WebLogicServer>⇒
 <BEA-000330> <Started WebLogic Managed Server "RPS"⇒
 for domain "peoplesoft" running in Production Mode>
```

```
<Feb 27, 2004 2:17:04 PM PST> <Notice> <WebLogicServer>⇒
 <BEA-000360> <Server started in RUNNING mode>
```

Note. During the PIA setup of the distributed managed server, nothing was specified to indicate which server would be run on PT-LINUX01. The managed server that's run as a distributed managed server is the server name specified as a command line parameter to running startWeblogic along with the URL of its administration server.

APPENDIX B

PeopleSoft Timeout Settings

This appendix discusses:

- Web server timeouts.
- Application server timeouts.
- Process Scheduler timeouts.
- PIA timeouts.

Web Server Timeouts

You specify web server timeouts using the Web Profile Configuration component (WEB_PROFILE). To access these settings in PIA, select PeopleTools, Web Profile, Web Profile Configuration, then select the appropriate page.

The following table provides basic information about the web server timeout settings, which are more completely documented in the *PeopleTools 8.46 Internet Technology PeopleBook*.

Page Element	Page Name	Description	Default
Inactivity Warning	Security	Specify how long the portal should wait before warning users that their browser session is about to expire. They can continue with their current session by clicking the OK button in the message. If a user doesn't respond, the session ends and the expired connection page appears. Suppress this warning by setting this value to be greater than the sessionTimeout value.	1080 seconds (18 minutes)
Inactivity Logout	Security	Specify the inactivity timeout interval of the PeopleSoft application for which the user is currently authenticated. When the interval passes with no user activity, the user's browser displays the page specified by the Expire Page - Page field on the Web Profile Configuration - Look and Feel page. **Note.** Depending on the application implementation, authenticated users might also experience an HTTP session inactivity timeout.	1200 seconds (20 minutes)

Page Element	Page Name	Description	Default
Authenticated Users - HTTP Session Inactivity	Security	Specify the HTTP session inactivity timeout interval that applies to 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. If not set, the HTTP interval for an authenticated user is the same value as the inactivity logout.	1200 seconds (20 minutes)
Public Users - HTTP Session Inactivity	Security	Specify in seconds the inactivity timeout interval that applies to public 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. Unlike authenticated users, public users are not signed out of their PeopleSoft application when this interval expires. However, PIA releases their application states from memory. If users click a link, they regain access to the application at the search dialog. This setting prevents an overload of web server resources for inactive public users.	DEV, KIOSK profile: 1200 seconds (20 minutes). TEST, PROD profile: not set.
Disconnect Timeout	Security	Specify the amount of time to wait before disconnecting the BEA Jolt connection. A value of *0* seconds (the default) means no limit. This means that the client connection must be retained throughout the session. If the connection becomes invalid (due to one one of the other timeouts) the session will be expired. **Note.** If you specify *0* seconds, the Jolt client attempts to connect the Jolt Server Handler (JSH) in RETAINED mode. If any positive value is specified, the Jolt client attempts to connect the JSH in RECONNECT mode.	0 seconds

Page Element	Page Name	Description	Default
Send Timeout	Security	Specify the maximum time permitted between the sending of the Jolt Request by the client servlet and its full receipt on the application server. **Note.** You might need to increase this value where a large amount of data is being sent to the application server, or the network is slow.	50 seconds
Receive Timeout	Security	Specify how long the client servlet should wait after issuing a Jolt Request for a response from the application server. This value should be considerably larger than the Send Timeout. Make sure that this value is also greater than your application server online service timeouts, such as the Service Timeout setting for PSAPPSRV that appears in the PSAPPSRV.CFG configuration file on the application server. **Note.** Ideally this timeout should also be greater than the Tuxedo SANITY_SCAN setting (BLOCKTIME * SCANUNIT).	600 seconds (10 minutes)

See Also

Enterprise PeopleTools 8.46 PeopleBook: Internet Technology, "Configuring the Portal Environment," Configuring Portal Security

Session-Timeout

You specify the web server *session-timeout* setting in the web.xml file for the web server (BEA WebLogic or IBM WebSphere) on which you're running your PeopleSoft system.

This setting (in minutes) determines the interval that elapses before the web server terminates the HttpSession. This is similar to an abandoned session cleanup timeout.

If you specify a value less than Inactivity Logout in the current web profile, it doesn't terminate the user's online session. The HttpSession is removed, but the user's session remains valid because cookies are present in the user's browser. Adjusting this setting affects users by causing their states (stored in the HttpSession) to be lost. If this setting is too high, it affects resource utilization on the web server.

Ideally, the value of this setting should be the same as the Inactivity Logout setting. This prevents both state loss and dangling HttpSessions on the web server. The default value of this setting is 20 minutes.

Application Server Timeouts

All configurable settings for the application server require modification in PSADMIN:

Name	In This File	Description	Default
JOLT Listener/Client CleanupTimeout	psappsrv.cfg	Specify the inactivity interval permitted for the server-side JoltSession. Specifying too low a value can cause unnecessary reinstantiation of resources for clients who surpass this inactivity interval. However, specifying too high a value can keep unnecessary server-side resources allocated. **Note.** This setting doesn't affect the user experience, but it has an impact on server-side performance.	10 minutes
JOLT Listener/Init Timeout	psappsrv.cfg	Specify the amount of time that's allowed for the JSL process to start. **Note.** It's not necessary to adjust this setting.	5 minutes
Workstation Listener/Client Cleanup Timeout	psappsrv.cfg	Specify the inactivity interval permitted for the server-side Workstation Listener Session. Specifying too low a value can cause unnecessary reinstantiation of resources for clients who surpass this inactivity interval. However, specifying too high a value can keep unnecessary server-side resources allocated. **Note.** This value is required only for three-tier connections.	60 minutes
Workstation Listener/init Timeout	psappsrv.cfg	Specify the amount of time that's allowed for the WSL process to start. **Note.** It's not necessary to adjust this setting. This value is required only for three-tier connections.	5 minutes

Name	In This File	Description	Default
Spawn Threshold	psappsrv.cfg	Specify the rates at which PSAPPSRV processes spawn and decay. The spawn rate is determined by the last two numbers, and the decay rate is determined by the first two numbers. Using the default value as an example, for the spawn rate of *1,1* an extra PSAPPSRV process is spawned if there is at least 1 oustanding service request on the application server request queue for 1 second or more. This spawning will continue until the PSAPPSRV Max Instances value is reached. For the decay rate of *1,600* a server process is decayed if less than 1 service request is in the application server request queue for 600 seconds (ten minutes) or more. **Note.** This parameter applies only if, for PSAPPSRV, the value of *Max Instances* is greater than that of *Min Instances*.	1,600:1,1
Service Timeout	psappsrv.cfg	Each server process has its own instance of this setting in its section of the psappsrv.cfg file. Specify the maximum interval for services to run in a given process. If a service has not completed within the specified interval, BEA Tuxedo terminates the server processing and restarts the server process. For each server process, specify the longest time that any service is expected to take. **Note.** A value of 0 produces an indefinite timeout for any service.	PSAPPSRV: 300 seconds (5 minutes) PSOPTENG: 270 seconds (must be less than the PSAPPSRV timeout) PSSAMSRV: 300 seconds PSQCKSRV: 300 seconds PSQRYSRV: 1200 seconds (20 minutes) PSUQSRV: 300 seconds PSMCFLOG: 300 seconds PSRENSRV: 300 secs PSBRKHND_dflt: 1200 seconds PSSUBHND_dflt: 1200 seconds PSPUBHND_dflt: 1200 seconds

Name	In This File	Description	Default
Restart Period (PSBRKDSP_dflt, PSSUBDSP_dflt, PSPUBDSP_dflt)	psappsrv.cfg	Specify how long each dispatcher should wait before redispatching a message if the associated handler has not started processing it.	120 seconds
TM_RESTARTSRV TIMEOUT	psappsrv.ubx (which is the template for psappsrv.env)	Specify the time period that a domain server process (for example, PSAPPSRV, PSWATCHSRV, PSSAMSRV) is permitted to remain in REStarting mode before it is killed by Tuxedo. This setting resolves processes hanging during restart. **Note.** To modify this setting, you must change the value in the .UBX template file, then recreate your domain.	60 seconds (one minute)

Process Scheduler Timeouts

All configurable settings for PeopleSoft Process Scheduler require modification through domain configuration within PSADMIN:

Name	In This File	Description	Default
Process Scheduler /Reconnection Interval	psprcs.cfg	Specify the interval between attempts to reconnect to the database when the connection is lost.	300 seconds (5 minutes)
Process Scheduler /Authentication Timeout	psprcs.cfg	Specify how long PeopleSoft Security has to authenticate a process that's released by PeopleSoft Process Scheduler The timer starts when Process Scheduler initiates the request.	5 minutes
PSOPTENG/Service Timeout	psprcs.cfg	Specify the maximum interval for services to run in the process. If a service has not completed within the specified interval, BEA Tuxedo terminates the server processing and restarts the server process. Specify the longest time that any service is expected to take. **Note.** A value of 0 produces an indefinite timeout for any service.	10 seconds
RemoteCall/RCCBL Timeout	psprcs.cfg	Specify the maximum interval for a remote call from an Application Engine program to run before it's terminated. This is similar to a general BEA Tuxedo service timeout.	300 seconds (5 minutes)

PIA Timeouts

A number of additional timeouts may be set through PIA. These settings reflect changes at the database level that may pertain to different groups of users.

Note. The timeout settings in PIA are optional and are not required to run PIA. However, an understanding of how these settings can contribute to a user's session duration is important in the context of other timeout values that appear in this appendix.

Name	Navigation Path	Description	Default
Authentication Token expiration time	PeopleTools, Security, Security Objects, Single Signon	Specify the interval during which the system can trust a single signon token (PS_TOKEN) from the same or another content provider. **Note.** As long as users remain signed in, the expiration of PS_TOKEN does not affect them. This setting is relevant only for the GetCertificate request during single signon.	720 minutes (12 hours)
Permission List - Time-out Minutes	PeopleTools, Security, Permissions & Roles, Permission Lists	Specify an interval during which a given permission list applies. The interval starts for a user to which the permission list is assigned when that user signs in. When the timeout period elapses, the user's online session is terminated. If a user belongs to multiple permission lists, the largest timeout value from among those permission lists is applied to the user's session during signon. The permission list timeout is effective only if its value is less than the web server session-timeout. This means that all of the permission list timeouts for a given user must be less than the web server session-timeout to be effective. However, the Inactivity Warning timeout still applies. **Note.** A value of 0 produces an indefinite timeout.	0 minutes

See Also

Appendix B, "PeopleSoft Timeout Settings," Web Server Timeouts, page 323

Glossary of PeopleSoft Terms

absence entitlement This element defines rules for granting paid time off for valid absences, such as sick time, vacation, and maternity leave. An absence entitlement element defines the entitlement amount, frequency, and entitlement period.

absence take This element defines the conditions that must be met before a payee is entitled to take paid time off.

academic career In PeopleSoft Enterprise Campus Solutions, all course work that a student undertakes at an academic institution and that is grouped in a single student record. For example, a university that has an undergraduate school, a graduate school, and various professional schools might define several academic careers—an undergraduate career, a graduate career, and separate careers for each professional school (law school, medical school, dental school, and so on).

academic institution In PeopleSoft Enterprise Campus Solutions, an entity (such as a university or college) that is independent of other similar entities and that has its own set of rules and business processes.

academic organization In PeopleSoft Enterprise Campus Solutions, an entity that is part of the administrative structure within an academic institution. At the lowest level, an academic organization might be an academic department. At the highest level, an academic organization can represent a division.

academic plan In PeopleSoft Enterprise Campus Solutions, an area of study—such as a major, minor, or specialization—that exists within an academic program or academic career.

academic program In PeopleSoft Enterprise Campus Solutions, the entity to which a student applies and is admitted and from which the student graduates.

accounting class In PeopleSoft Enterprise Performance Management, the accounting class defines how a resource is treated for generally accepted accounting practices. The Inventory class indicates whether a resource becomes part of a balance sheet account, such as inventory or fixed assets, while the Non-inventory class indicates that the resource is treated as an expense of the period during which it occurs.

accounting date The accounting date indicates when a transaction is recognized, as opposed to the date the transaction actually occurred. The accounting date and transaction date can be the same. The accounting date determines the period in the general ledger to which the transaction is to be posted. You can only select an accounting date that falls within an open period in the ledger to which you are posting. The accounting date for an item is normally the invoice date.

accounting split The accounting split method indicates how expenses are allocated or divided among one or more sets of accounting ChartFields.

accumulator You use an accumulator to store cumulative values of defined items as they are processed. You can accumulate a single value over time or multiple values over time. For example, an accumulator could consist of all voluntary deductions, or all company deductions, enabling you to accumulate amounts. It allows total flexibility for time periods and values accumulated.

action reason The reason an employee's job or employment information is updated. The action reason is entered in two parts: a personnel action, such as a promotion, termination, or change from one pay group to another—and a reason for that action. Action reasons are used by PeopleSoft Human Resources, PeopleSoft Benefits Administration,

PeopleSoft Stock Administration, and the COBRA Administration feature of the Base Benefits business process.

action template In PeopleSoft Receivables, outlines a set of escalating actions that the system or user performs based on the period of time that a customer or item has been in an action plan for a specific condition.

activity In PeopleSoft Enterprise Learning Management, an instance of a catalog item (sometimes called a class) that is available for enrollment. The activity defines such things as the costs that are associated with the offering, enrollment limits and deadlines, and waitlisting capacities.

In PeopleSoft Enterprise Performance Management, the work of an organization and the aggregation of actions that are used for activity-based costing.

In PeopleSoft Project Costing, the unit of work that provides a further breakdown of projects—usually into specific tasks.

In PeopleSoft Workflow, a specific transaction that you might need to perform in a business process. Because it consists of the steps that are used to perform a transaction, it is also known as a step map.

address usage In PeopleSoft Enterprise Campus Solutions, a grouping of address types defining the order in which the address types are used. For example, you might define an address usage code to process addresses in the following order: billing address, dormitory address, home address, and then work address.

adjustment calendar In PeopleSoft Enterprise Campus Solutions, the adjustment calendar controls how a particular charge is adjusted on a student's account when the student drops classes or withdraws from a term. The charge adjustment is based on how much time has elapsed from a predetermined date, and it is determined as a percentage of the original charge amount.

administrative function In PeopleSoft Enterprise Campus Solutions, a particular functional area that processes checklists, communication, and comments. The administrative function identifies which variable data is added to a person's checklist or communication record when a specific checklist code, communication category, or comment is assigned to the student. This key data enables you to trace that checklist, communication, or comment back to a specific processing event in a functional area.

admit type In PeopleSoft Enterprise Campus Solutions, a designation used to distinguish first-year applications from transfer applications.

agreement In PeopleSoft eSettlements, provides a way to group and specify processing options, such as payment terms, pay from a bank, and notifications by a buyer and supplier location combination.

allocation rule In PeopleSoft Enterprise Incentive Management, an expression within compensation plans that enables the system to assign transactions to nodes and participants. During transaction allocation, the allocation engine traverses the compensation structure from the current node to the root node, checking each node for plans that contain allocation rules.

alternate account A feature in PeopleSoft General Ledger that enables you to create a statutory chart of accounts and enter statutory account transactions at the detail transaction level, as required for recording and reporting by some national governments.

analysis database In PeopleSoft Enterprise Campus Solutions, database tables that store large amounts of student information that may not appear in standard report formats. The analysis database tables contain keys for all objects in a report that an application program can use to reference other student-record objects that are not contained in the printed report. For instance, the analysis database contains data on courses that are considered for satisfying a requirement but that are rejected. It also contains information on

courses captured by global limits. An analysis database is used in PeopleSoft Enterprise Academic Advisement.

Application Messaging PeopleSoft Application Messaging enables applications within the PeopleSoft Enterprise product family to communicate synchronously or asynchronously with other PeopleSoft and third-party applications. An application message defines the records and fields to be published or subscribed to.

AR specialist Abbreviation for *receivables specialist.* In PeopleSoft Receivables, an individual in who tracks and resolves deductions and disputed items.

arbitration plan In PeopleSoft Enterprise Pricer, defines how price rules are to be applied to the base price when the transaction is priced.

assessment rule In PeopleSoft Receivables, a user-defined rule that the system uses to evaluate the condition of a customer's account or of individual items to determine whether to generate a follow-up action.

asset class An asset group used for reporting purposes. It can be used in conjunction with the asset category to refine asset classification.

attribute/value pair In PeopleSoft Directory Interface, relates the data that makes up an entry in the directory information tree.

audience In PeopleSoft Enterprise Campus Solutions, a segment of the database that relates to an initiative, or a membership organization that is based on constituent attributes rather than a dues-paying structure. Examples of audiences include the Class of '65 and Undergraduate Arts & Sciences.

authentication server A server that is set up to verify users of the system.

base time period In PeopleSoft Business Planning, the lowest level time period in a calendar.

benchmark job In PeopleSoft Workforce Analytics, a benchmark job is a job code for which there is corresponding salary survey data from published, third-party sources.

billing career In PeopleSoft Enterprise Campus Solutions, the one career under which other careers are grouped for billing purposes if a student is active simultaneously in multiple careers.

bio bit or bio brief In PeopleSoft Enterprise Campus Solutions, a report that summarizes information stored in the system about a particular constituent. You can generate standard or specialized reports.

book In PeopleSoft Asset Management, used for storing financial and tax information, such as costs, depreciation attributes, and retirement information on assets.

branch A tree node that rolls up to nodes above it in the hierarchy, as defined in PeopleSoft Tree Manager.

budgetary account only An account used by the system only and not by users; this type of account does not accept transactions. You can only budget with this account. Formerly called "system-maintained account."

budget check In commitment control, the processing of source transactions against control budget ledgers, to see if they pass, fail, or pass with a warning.

budget control In commitment control, budget control ensures that commitments and expenditures don't exceed budgets. It enables you to track transactions against corresponding budgets and terminate a document's cycle if the defined budget conditions are not met. For example, you can prevent a purchase order from being dispatched to a vendor if there are insufficient funds in the related budget to support it.

budget period The interval of time (such as 12 months or 4 quarters) into which a period is divided for budgetary and reporting purposes. The ChartField allows maximum flexibility to define operational accounting time periods without restriction to only one calendar.

business activity The name of a subset of a detailed business process. This might be a specific transaction, task, or action that you perform in a business process.

business event In PeopleSoft Receivables, defines the processing characteristics for the Receivable Update process for a draft activity.

In PeopleSoft Sales Incentive Management, an original business transaction or activity that may justify the creation of a PeopleSoft Enterprise Incentive Management event (a sale, for example).

business process A standard set of 17 business processes are defined and maintained by the PeopleSoft product families and are supported by Business Process Engineering group at PeopleSoft. An example of a business process is Order Fulfillment, which is a business process that manages sales orders and contracts, inventory, billing, and so forth.

See also *detailed business process.*

business task The name of the specific function depicted in one of the business processes.

business unit A corporation or a subset of a corporation that is independent with regard to one or more operational or accounting functions.

buyer In PeopleSoft eSettlements, an organization (or business unit, as opposed to an individual) that transacts with suppliers (vendors) within the system. A buyer creates payments for purchases that are made in the system.

campus In PeopleSoft Enterprise Campus Solutions, an entity that is usually associated with a distinct physical administrative unit, that belongs to a single academic institution, that uses a unique course catalog, and that produces a common transcript for students within the same academic career.

catalog item In PeopleSoft Enterprise Learning Management, a specific topic that a learner can study and have tracked. For example, “Introduction to Microsoft Word.” A catalog item contains general information about the topic and includes a course code, description, categorization, keywords, and delivery methods. A catalog item can have one or more learning activities.

catalog map In PeopleSoft Catalog Management, translates values from the catalog source data to the format of the company’s catalog.

catalog partner In PeopleSoft Catalog Management, shares responsibility with the enterprise catalog manager for maintaining catalog content.

categorization Associates partner offerings with catalog offerings and groups them into enterprise catalog categories.

category In PeopleSoft Enterprise Campus Solutions, a broad grouping to which specific comments or communications (contexts) are assigned. Category codes are also linked to 3C access groups so that you can assign data-entry or view-only privileges across functions.

channel In PeopleSoft MultiChannel Framework, email, chat, voice (computer telephone integration [CTI]), or a generic event.

ChartField A field that stores a chart of accounts, resources, and so on, depending on the PeopleSoft application. ChartField values represent individual account numbers, department codes, and so forth.

ChartField balancing You can require specific ChartFields to match up (balance) on the debit and the credit side of a transaction.

ChartField combination edit The process of editing journal lines for valid ChartField combinations based on user-defined rules.

ChartKey One or more fields that uniquely identify each row in a table. Some tables contain only one field as the key, while others require a combination.

checkbook In PeopleSoft Promotions Management, enables you to view financial data (such as planned, incurred, and actual amounts) that is related to funds and trade promotions.

checklist code In PeopleSoft Enterprise Campus Solutions, a code that represents a list of planned or completed action items that can be assigned to a staff member, volunteer, or unit. Checklists enable you to view all action assignments on one page.

class In PeopleSoft Enterprise Campus Solutions, a specific offering of a course component within an academic term.

See also *course.*

Class ChartField A ChartField value that identifies a unique appropriation budget key when you combine it with a fund, department ID, and program code, as well as a budget period. Formerly called *sub-classification.*

clearance In PeopleSoft Enterprise Campus Solutions, the period of time during which a constituent in PeopleSoft Contributor Relations is approved for involvement in an initiative or an action. Clearances are used to prevent development officers from making multiple requests to a constituent during the same time period.

clone In PeopleCode, to make a unique copy. In contrast, to *copy* may mean making a new reference to an object, so if the underlying object is changed, both the copy and the original change.

cohort In PeopleSoft Enterprise Campus Solutions, the highest level of the three-level classification structure that you define for enrollment management. You can define a cohort level, link it to other levels, and set enrollment target numbers for it.

See also *population* and *division*.

collection To make a set of documents available for searching in Verity, you must first create at least one collection. A collection is set of directories and files that allow search application users to use the Verity search engine to quickly find and display source documents that match search criteria. A collection is a set of statistics and pointers to the source documents, stored in a proprietary format on a file server. Because a collection can only store information for a single location, PeopleSoft maintains a set of collections (one per language code) for each search index object.

collection rule In PeopleSoft Receivables, a user-defined rule that defines actions to take for a customer based on both the amount and the number of days past due for outstanding balances.

comm key See *communication key.*

communication key In PeopleSoft Enterprise Campus Solutions, a single code for entering a combination of communication category, communication context, communication method, communication direction, and standard letter code. Communication keys (also called *comm keys* or *speed keys*) can be created for background processes as well as for specific users.

compensation object In PeopleSoft Enterprise Incentive Management, a node within a compensation structure. Compensation objects are the building blocks that make up a compensation structure's hierarchical representation.

compensation structure In PeopleSoft Enterprise Incentive Management, a hierarchical relationship of compensation objects that represents the compensation-related relationship between the objects.

component interface A component interface is a set of application programming interfaces (APIs) that you can use to access and modify PeopleSoft database information using a program instead of the PeopleSoft client.

condition In PeopleSoft Receivables, occurs when there is a change of status for a customer's account, such as reaching a credit limit or exceeding a user-defined balance due.

configuration parameter catalog Used to configure an external system with PeopleSoft. For example, a configuration parameter catalog might set up configuration and communication parameters for an external server.

configuration plan In PeopleSoft Enterprise Incentive Management, configuration plans hold allocation information for common variables (not incentive rules) and are attached to a node without a participant. Configuration plans are not processed by transactions.

constituents In PeopleSoft Enterprise Campus Solutions, friends, alumni, organizations, foundations, or other entities affiliated with the institution, and about which the institution maintains information. The constituent types delivered with PeopleSoft Enterprise Contributor Relations Solutions are based on those defined by the Council for the Advancement and Support of Education (CASE).

content reference Content references are pointers to content registered in the portal registry. These are typically either URLs or iScripts. Content references fall into three categories: target content, templates, and template pagelets.

context In PeopleCode, determines which buffer fields can be contextually referenced and which is the current row of data on each scroll level when a PeopleCode program is running.

In PeopleSoft Enterprise Campus Solutions, a specific instance of a comment or communication. One or more contexts are assigned to a category, which you link to 3C access groups so that you can assign data-entry or view-only privileges across functions.

In PeopleSoft Enterprise Incentive Management, a mechanism that is used to determine the scope of a processing run. PeopleSoft Enterprise Incentive Management uses three types of context: plan, period, and run-level.

control table Stores information that controls the processing of an application. This type of processing might be consistent throughout an organization, or it might be used only by portions of the organization for more limited sharing of data.

cost profile A combination of a receipt cost method, a cost flow, and a deplete cost method. A profile is associated with a cost book and determines how items in that book are valued, as well as how the material movement of the item is valued for the book.

cost row A cost transaction and amount for a set of ChartFields.

course In PeopleSoft Enterprise Campus Solutions, a course that is offered by a school and that is typically described in a course catalog. A course has a standard syllabus and credit level; however, these may be modified at the class level. Courses can contain multiple components such as lecture, discussion, and lab.

See also *class*.

course share set In PeopleSoft Enterprise Campus Solutions, a tag that defines a set of requirement groups that can share courses. Course share sets are used in PeopleSoft Enterprise Academic Advisement.

current learning In PeopleSoft Enterprise Learning Management, a self-service repository for all of a learner's in-progress learning activities and programs.

data acquisition In PeopleSoft Enterprise Incentive Management, the process during which raw business transactions are acquired from external source systems and fed into the operational data store (ODS).

data cube In PeopleSoft Analytic Calculation Engine, a data cube is a container for one kind of data (such as Sales data) and works with in tandem with one or more dimensions. Dimensions and data cubes in PeopleSoft Analytic Calculation Engine are unrelated to dimensions and online analytical processing (OLAP) cubes in PeopleSoft Cube Manager.

data elements Data elements, at their simplest level, define a subset of data and the rules by which to group them.

For Workforce Analytics, data elements are rules that tell the system what measures to retrieve about your workforce groups.

dataset A data grouping that enables role-based filtering and distribution of data. You can limit the range and quantity of data that is displayed for a user by associating dataset rules with user roles. The result of dataset rules is a set of data that is appropriate for the user's roles.

delivery method In PeopleSoft Enterprise Learning Management, identifies the primary type of delivery method in which a particular learning activity is offered. Also provides default values for the learning activity, such as cost and language. This is primarily used to help learners search the catalog for the type of delivery from which they learn best. Because PeopleSoft Enterprise Learning Management is a blended learning system, it does not enforce the delivery method.

In PeopleSoft Supply Chain Management, identifies the method by which goods are shipped to their destinations (such as truck, air, rail, and so on). The delivery method is specified when creating shipment schedules.

delivery method type In PeopleSoft Enterprise Learning Management, identifies how learning activities can be delivered—for example, through online learning, classroom instruction, seminars, books, and so forth—in an organization. The type determines whether the delivery method includes scheduled components.

detailed business process A subset of the business process. For example, the detailed business process named Determine Cash Position is a subset of the business process called Cash Management.

dimension In PeopleSoft Analytic Calculation Engine, a dimension contains a list of one kind of data that can span various contexts, and it is a basic component of an analytic model. Within the analytic model, a dimension is attached to one or more data cubes. In PeopleSoft Cube Manager, a dimension is the most basic component of an OLAP cube and specifies the PeopleSoft metadata to be used to create the dimension's rollup structure. Dimensions and data cubes in PeopleSoft Analytic Calculation Engine are unrelated to dimensions and OLAP cubes in PeopleSoft Cube Manager.

directory information tree In PeopleSoft Directory Interface, the representation of a directory's hierarchical structure.

division In PeopleSoft Enterprise Campus Solutions, the lowest level of the three-level classification structure that you define in PeopleSoft Enterprise Recruiting and Admissions for enrollment management. You can define a division level, link it to other levels, and set enrollment target numbers for it.

See also *population* and *cohort*.

document sequencing A flexible method that sequentially numbers the financial transactions (for example, bills, purchase orders, invoices, and payments) in the system for statutory reporting and for tracking commercial transaction activity.

dynamic detail tree A tree that takes its detail values—dynamic details—directly from a table in the database, rather than from a range of values that are entered by the user.

edit table A table in the database that has its own record definition, such as the Department table. As fields are entered into a PeopleSoft application, they can be validated against an edit table to ensure data integrity throughout the system.

effective date A method of dating information in PeopleSoft applications. You can predate information to add historical data to your system, or postdate information in order to enter it before it actually goes into effect. By using effective dates, you don't delete values; you enter a new value with a current effective date.

EIM ledger Abbreviation for *Enterprise Incentive Management ledger.* In PeopleSoft Enterprise Incentive Management, an object to handle incremental result gathering within the scope of a participant. The ledger captures a result set with all of the appropriate traces to the data origin and to the processing steps of which it is a result.

elimination set In PeopleSoft General Ledger, a related group of intercompany accounts that is processed during consolidations.

entry event In PeopleSoft General Ledger, Receivables, Payables, Purchasing, and Billing, a business process that generates multiple debits and credits resulting from single transactions to produce standard, supplemental accounting entries.

equitization In PeopleSoft General Ledger, a business process that enables parent companies to calculate the net income of subsidiaries on a monthly basis and adjust that amount to increase the investment amount and equity income amount before performing consolidations.

equity item limit In PeopleSoft Enterprise Campus Solutions, the amounts of funds set by the institution to be awarded with discretionary or gift funds. The limit could be reduced by amounts equal to such things as expected family contribution (EFC) or parent contribution. Students are packaged by Equity Item Type Groups and Related Equity Item Types. This limit can be used to assure that similar student populations are packaged equally.

event A predefined point either in the Component Processor flow or in the program flow. As each point is encountered, the event activates each component, triggering any PeopleCode program that is associated with that component and that event. Examples of events are FieldChange, SavePreChange, and RowDelete.

In PeopleSoft Human Resources, also refers to an incident that affects benefits eligibility.

event propagation process In PeopleSoft Sales Incentive Management, a process that determines, through logic, the propagation of an original PeopleSoft Enterprise Incentive Management event and creates a derivative (duplicate) of the original event to be processed by other objects. Sales Incentive Management uses this mechanism to implement splits, roll-ups, and so on. Event propagation determines who receives the credit.

exception In PeopleSoft Receivables, an item that either is a deduction or is in dispute.

exclusive pricing In PeopleSoft Order Management, a type of arbitration plan that is associated with a price rule. Exclusive pricing is used to price sales order transactions.

fact In PeopleSoft applications, facts are numeric data values from fields from a source database as well as an analytic application. A fact can be anything you want to measure your business by, for example, revenue, actual, budget data, or sales numbers. A fact is stored on a fact table.

financial aid term In PeopleSoft Enterprise Campus Solutions, a combination of a period of time that the school determines as an instructional accounting period and an academic career. It is created and defined during the setup process. Only terms eligible for financial aid are set up for each financial aid career.

forecast item A logical entity with a unique set of descriptive demand and forecast data that is used as the basis to forecast demand. You create forecast items for a wide range of uses, but they ultimately represent things that you buy, sell, or use in your organization and for which you require a predictable usage.

fund In PeopleSoft Promotions Management, a budget that can be used to fund promotional activity. There are four funding methods: top down, fixed accrual, rolling accrual, and zero-based accrual.

gap In PeopleSoft Enterprise Campus Solutions, an artificial figure that sets aside an amount of unmet financial aid need that is not funded with Title IV funds. A gap can be used to prevent fully funding any student to conserve funds, or it can be used to preserve unmet financial aid need so that institutional funds can be awarded.

generic process type In PeopleSoft Process Scheduler, process types are identified by a generic process type. For example, the generic process type SQR includes all SQR process types, such as SQR process and SQR report.

gift table In PeopleSoft Enterprise Campus Solutions, a table or so-called *donor pyramid* describing the number and size of gifts that you expect will be needed to successfully complete the campaign in PeopleSoft Contributor Relations. The gift table enables you to estimate the number of donors and prospects that you need at each gift level to reach the campaign goal.

GL business unit Abbreviation for *general ledger business unit.* A unit in an organization that is an independent entity for accounting purposes. It maintains its own set of accounting books.

See also *business unit*.

GL entry template Abbreviation for *general ledger entry template.* In PeopleSoft Enterprise Campus Solutions, a template that defines how a particular item is sent to the general ledger. An item-type maps to the general ledger, and the GL entry template can involve multiple general ledger accounts. The entry to the general ledger is further controlled by high-level flags that control the summarization and the type of accounting—that is, accrual or cash.

GL Interface process Abbreviation for *General Ledger Interface process.* In PeopleSoft Enterprise Campus Solutions, a process that is used to send transactions from PeopleSoft Enterprise Student Financials to the general ledger. Item types are mapped to specific general ledger accounts, enabling transactions to move to the general ledger when the GL Interface process is run.

group In PeopleSoft Billing and Receivables, a posting entity that comprises one or more transactions (items, deposits, payments, transfers, matches, or write-offs).

In PeopleSoft Human Resources Management and Supply Chain Management, any set of records that are associated under a single name or variable to run calculations in PeopleSoft business processes. In PeopleSoft Time and Labor, for example, employees are placed in groups for time reporting purposes.

incentive object In PeopleSoft Enterprise Incentive Management, the incentive-related objects that define and support the PeopleSoft Enterprise Incentive Management calculation process and results, such as plan templates, plans, results data, user interaction objects, and so on.

incentive rule In PeopleSoft Sales Incentive Management, the commands that act on transactions and turn them into compensation. A rule is one part in the process of turning a transaction into compensation.

incur In PeopleSoft Promotions Management, to become liable for a promotional payment. In other words, you owe that amount to a customer for promotional activities.

initiative In PeopleSoft Enterprise Campus Solutions, the basis from which all advancement plans are executed. It is an organized effort targeting a specific constituency, and it can occur over a specified period of time with specific purposes and goals. An initiative can be a campaign, an event, an organized volunteer effort, a membership drive, or any other type of effort defined by the institution. Initiatives can be multipart, and they can be related to other initiatives. This enables you to track individual parts of an initiative, as well as entire initiatives.

inquiry access In PeopleSoft Enterprise Campus Solutions, a type of security access that permits the user only to view data.

See also *update access.*

institution In PeopleSoft Enterprise Campus Solutions, an entity (such as a university or college) that is independent of other similar entities and that has its own set of rules and business processes.

integration A relationship between two compatible integration points that enables communication to take place between systems. Integrations enable PeopleSoft applications to work seamlessly with other PeopleSoft applications or with third-party systems or software.

integration point An interface that a system uses to communicate with another PeopleSoft application or an external application.

integration set A logical grouping of integrations that applications use for the same business purpose. For example, the integration set ADVANCED_SHIPPING_ORDER contains all of the integrations that notify a customer that an order has shipped.

item In PeopleSoft Inventory, a tangible commodity that is stored in a business unit (shipped from a warehouse).

In PeopleSoft Demand Planning, Inventory Policy Planning, and Supply Planning, a noninventory item that is designated as being used for planning purposes only. It can represent a family or group of inventory items. It can have a planning bill of material (BOM) or planning routing, and it can exist as a component on a planning BOM. A planning item cannot be specified on a production or engineering BOM or routing, and it cannot be used as a component in a production. The quantity on hand will never be maintained.

In PeopleSoft Receivables, an individual receivable. An item can be an invoice, a credit memo, a debit memo, a write-off, or an adjustment.

item shuffle In PeopleSoft Enterprise Campus Solutions, a process that enables you to change a payment allocation without having to reverse the payment.

joint communication In PeopleSoft Enterprise Campus Solutions, one letter that is addressed jointly to two people. For example, a letter might be addressed to both Mr. Sudhir Awat and Ms. Samantha Mortelli. A relationship must be established between the two individuals in the database, and at least one of the individuals must have an ID in the database.

keyword In PeopleSoft Enterprise Campus Solutions, a term that you link to particular elements within PeopleSoft Student Financials, Financial Aid, and Contributor Relations. You can use keywords as search criteria that enable you to locate specific records in a search dialog box.

KPI An abbreviation for *key performance indicator.* A high-level measurement of how well an organization is doing in achieving critical success factors. This defines the data value or calculation upon which an assessment is determined.

LDIF file Abbreviation for *Lightweight Directory Access Protocol (LDAP) Data Interchange Format file.* Contains discrepancies between PeopleSoft data and directory data.

learner group In PeopleSoft Enterprise Learning Management, a group of learners who are linked to the same learning environment. Members of the learner group can share the same attributes, such as the same department or job code. Learner groups are used to control access to and enrollment in learning activities and programs. They are also used to perform group enrollments and mass enrollments in the back office.

learning components In PeopleSoft Enterprise Learning Management, the foundational building blocks of learning activities. PeopleSoft Enterprise Learning Management supports six basic types of learning components: web-based, session, webcast, test, survey, and assignment. One or more of these learning component types compose a single learning activity.

learning environment In PeopleSoft Enterprise Learning Management, identifies a set of categories and catalog items that can be made available to learner groups. Also defines the default values that are assigned to the learning activities and programs that are created within a particular learning environment. Learning environments provide a way to partition the catalog so that learners see only those items that are relevant to them.

learning history In PeopleSoft Enterprise Learning Management, a self-service repository for all of a learner's completed learning activities and programs.

ledger mapping You use ledger mapping to relate expense data from general ledger accounts to resource objects. Multiple ledger line items can be mapped to one or more resource IDs. You can also use ledger mapping to map dollar amounts (referred to as *rates*) to business units. You can map the amounts in two different ways: an actual amount that represents actual costs of the accounting period, or a budgeted amount that can be used to calculate the capacity rates as well as budgeted model results. In PeopleSoft Enterprise Warehouse, you can map general ledger accounts to the EW Ledger table.

library section In PeopleSoft Enterprise Incentive Management, a section that is defined in a plan (or template) and that is available for other plans to share. Changes to a library section are reflected in all plans that use it.

linked section In PeopleSoft Enterprise Incentive Management, a section that is defined in a plan template but appears in a plan. Changes to linked sections propagate to plans using that section.

linked variable In PeopleSoft Enterprise Incentive Management, a variable that is defined and maintained in a plan template and that also appears in a plan. Changes to linked variables propagate to plans using that variable.

LMS Abbreviation for *learning management system.* In PeopleSoft Enterprise Campus Solutions, LMS is a PeopleSoft Student Records feature that provides a common set of interoperability standards that enable the sharing of instructional content and data between learning and administrative environments.

load In PeopleSoft Inventory, identifies a group of goods that are shipped together. Load management is a feature of PeopleSoft Inventory that is used to track the weight, the volume, and the destination of a shipment.

local functionality In PeopleSoft HRMS, the set of information that is available for a specific country. You can access this information when you click the appropriate country flag in the global window, or when you access it by a local country menu.

location Locations enable you to indicate the different types of addresses—for a company, for example, one address to receive bills, another for shipping, a third for postal deliveries, and a separate street address. Each address has a different location number. The primary location—indicated by a *1*—is the address you use most often and may be different from the main address.

logistical task In PeopleSoft Services Procurement, an administrative task that is related to hiring a service provider. Logistical tasks are linked to the service type on the work order so that different types of services can have different logistical tasks. Logistical tasks include both preapproval tasks (such as assigning a new badge or ordering a new laptop) and postapproval tasks (such as scheduling orientation or setting up the service provider email). The logistical tasks can be mandatory or optional. Mandatory preapproval tasks must be completed before the work order is approved. Mandatory postapproval tasks, on the other hand, must be completed before a work order is released to a service provider.

market template In PeopleSoft Enterprise Incentive Management, additional functionality that is specific to a given market or industry and is built on top of a product category.

mass change In PeopleSoft Enterprise Campus Solutions, mass change is a SQL generator that can be used to create specialized functionality. Using mass change, you can set up a series of Insert, Update, or Delete SQL statements to perform business functions that are specific to the institution.

See also *3C engine*.

match group In PeopleSoft Receivables, a group of receivables items and matching offset items. The system creates match groups by using user-defined matching criteria for selected field values.

MCF server Abbreviation for *PeopleSoft MultiChannel Framework server.* Comprises the universal queue server and the MCF log server. Both processes are started when *MCF Servers* is selected in an application server domain configuration.

merchandising activity In PeopleSoft Promotions Management, a specific discount type that is associated with a trade promotion (such as off-invoice, billback or rebate, or lump-sum payment) that defines the performance that is required to receive the discount. In the industry, you may know this as an offer, a discount, a merchandising event, an event, or a tactic.

meta-SQL Meta-SQL constructs expand into platform-specific Structured Query Language (SQL) substrings. They are used in functions that pass SQL strings, such as in SQL objects, the SQLExec function, and PeopleSoft Application Engine programs.

metastring Metastrings are special expressions included in SQL string literals. The metastrings, prefixed with a percent (%) symbol, are included directly in the string literals. They expand at run time into an appropriate substring for the current database platform.

multibook In PeopleSoft General Ledger, multiple ledgers having multiple-base currencies that are defined for a business unit, with the option to post a single transaction to all base currencies (all ledgers) or to only one of those base currencies (ledgers).

multicurrency The ability to process transactions in a currency other than the business unit's base currency.

national allowance In PeopleSoft Promotions Management, a promotion at the corporate level that is funded by nondiscretionary dollars. In the industry, you may know this as a national promotion, a corporate promotion, or a corporate discount.

need In PeopleSoft Enterprise Campus Solutions, the difference between the cost of attendance (COA) and the expected family contribution (EFC). It is the gap between the cost of attending the school and the student's resources. The financial aid package

	is based on the amount of financial need. The process of determining a student's need is called *need analysis.*
node-oriented tree	A tree that is based on a detail structure, but the detail values are not used.
pagelet	Each block of content on the home page is called a pagelet. These pagelets display summary information within a small rectangular area on the page. The pagelet provide users with a snapshot of their most relevant PeopleSoft and non-PeopleSoft content.
participant	In PeopleSoft Enterprise Incentive Management, participants are recipients of the incentive compensation calculation process.
participant object	Each participant object may be related to one or more compensation objects. See also *compensation object.*
partner	A company that supplies products or services that are resold or purchased by the enterprise.
pay cycle	In PeopleSoft Payables, a set of rules that define the criteria by which it should select scheduled payments for payment creation.
payment shuffle	In PeopleSoft Enterprise Campus Solutions, a process allowing payments that have been previously posted to a student's account to be automatically reapplied when a higher priority payment is posted or the payment allocation definition is changed.
pending item	In PeopleSoft Receivables, an individual receivable (such as an invoice, a credit memo, or a write-off) that has been entered in or created by the system, but hasn't been posted.
PeopleCode	PeopleCode is a proprietary language, executed by the PeopleSoft component processor. PeopleCode generates results based on existing data or user actions. By using various tools provided with PeopleTools, external services are available to all PeopleSoft applications wherever PeopleCode can be executed.
PeopleCode event	See *event.*
PeopleSoft Pure Internet Architecture	The fundamental architecture on which PeopleSoft 8 applications are constructed, consisting of a relational database management system (RDBMS), an application server, a web server, and a browser.
performance measurement	In PeopleSoft Enterprise Incentive Management, a variable used to store data (similar to an aggregator, but without a predefined formula) within the scope of an incentive plan. Performance measures are associated with a plan calendar, territory, and participant. Performance measurements are used for quota calculation and reporting.
period context	In PeopleSoft Enterprise Incentive Management, because a participant typically uses the same compensation plan for multiple periods, the period context associates a plan context with a specific calendar period and fiscal year. The period context references the associated plan context, thus forming a chain. Each plan context has a corresponding set of period contexts.
person of interest	A person about whom the organization maintains information but who is not part of the workforce.
personal portfolio	In PeopleSoft Enterprise Campus Solutions, the user-accessible menu item that contains an individual's name, address, telephone number, and other personal information.
plan	In PeopleSoft Sales Incentive Management, a collection of allocation rules, variables, steps, sections, and incentive rules that instruct the PeopleSoft Enterprise Incentive Management engine in how to process transactions.

plan context In PeopleSoft Enterprise Incentive Management, correlates a participant with the compensation plan and node to which the participant is assigned, enabling the PeopleSoft Enterprise Incentive Management system to find anything that is associated with the node and that is required to perform compensation processing. Each participant, node, and plan combination represents a unique plan context—if three participants are on a compensation structure, each has a different plan context. Configuration plans are identified by plan contexts and are associated with the participants that refer to them.

plan template In PeopleSoft Enterprise Incentive Management, the base from which a plan is created. A plan template contains common sections and variables that are inherited by all plans that are created from the template. A template may contain steps and sections that are not visible in the plan definition.

planned learning In PeopleSoft Enterprise Learning Management, a self-service repository for all of a learner's planned learning activities and programs.

planning instance In PeopleSoft Supply Planning, a set of data (business units, items, supplies, and demands) constituting the inputs and outputs of a supply plan.

population In PeopleSoft Enterprise Campus Solutions, the middle level of the three-level classification structure that you define in PeopleSoft Enterprise Recruiting and Admissions for enrollment management. You can define a population level, link it to other levels, and set enrollment target numbers for it.

See also *division* and *cohort*.

portal registry In PeopleSoft applications, the portal registry is a tree-like structure in which content references are organized, classified, and registered. It is a central repository that defines both the structure and content of a portal through a hierarchical, tree-like structure of folders useful for organizing and securing content references.

price list In PeopleSoft Enterprise Pricer, enables you to select products and conditions for which the price list applies to a transaction. During a transaction, the system either determines the product price based on the predefined search hierarchy for the transaction or uses the product's lowest price on any associated, active price lists. This price is used as the basis for any further discounts and surcharges.

price rule In PeopleSoft Enterprise Pricer, defines the conditions that must be met for adjustments to be applied to the base price. Multiple rules can apply when conditions of each rule are met.

price rule condition In PeopleSoft Enterprise Pricer, selects the price-by fields, the values for the price-by fields, and the operator that determines how the price-by fields are related to the transaction.

price rule key In PeopleSoft Enterprise Pricer, defines the fields that are available to define price rule conditions (which are used to match a transaction) on the price rule.

primacy number In PeopleSoft Enterprise Campus Solutions, a number that the system uses to prioritize financial aid applications when students are enrolled in multiple academic careers and academic programs at the same time. The Consolidate Academic Statistics process uses the primacy number indicated for both the career and program at the institutional level to determine a student's primary career and program. The system also uses the number to determine the primary student attribute value that is used when you extract data to report on cohorts. The lowest number takes precedence.

primary name type In PeopleSoft Enterprise Campus Solutions, the name type that is used to link the name stored at the highest level within the system to the lower-level set of names that an individual provides.

process category In PeopleSoft Process Scheduler, processes that are grouped for server load balancing and prioritization.

process group In PeopleSoft Financials, a group of application processes (performed in a defined order) that users can initiate in real time, directly from a transaction entry page.

process definition Process definitions define each run request.

process instance A unique number that identifies each process request. This value is automatically incremented and assigned to each requested process when the process is submitted to run.

process job You can link process definitions into a job request and process each request serially or in parallel. You can also initiate subsequent processes based on the return code from each prior request.

process request A single run request, such as a Structured Query Report (SQR), a COBOL or Application Engine program, or a Crystal report that you run through PeopleSoft Process Scheduler.

process run control A PeopleTools variable used to retain PeopleSoft Process Scheduler values needed at runtime for all requests that reference a run control ID. Do not confuse these with application run controls, which may be defined with the same run control ID, but only contain information specific to a given application process request.

product A PeopleSoft or third-party product. PeopleSoft organizes its software products into product families and product lines. Interactive Services Repository contains information about every release of every product that PeopleSoft sells, as well as products from certified third-party companies. These products are displayed with the product name and release number.

product category In PeopleSoft Enterprise Incentive Management, indicates an application in the Enterprise Incentive Management suite of products. Each transaction in the PeopleSoft Enterprise Incentive Management system is associated with a product category.

product family A group of products that are related by common functionality. The family names that can be searched using Interactive Service Repository are PeopleSoft Enterprise, PeopleSoft EnterpriseOne, PeopleSoft World, and third-party, certified PeopleSoft partners.

product line The name of a PeopleSoft product line or the company name of a third-party certified partner. Integration Services Repository enables you to search for integration points by product line.

programs In PeopleSoft Enterprise Learning Management, a high-level grouping that guides the learner along a specific learning path through sections of catalog items. PeopleSoft Enterprise Learning Systems provides two types of programs—curricula and certifications.

progress log In PeopleSoft Services Procurement, tracks deliverable-based projects. This is similar to the time sheet in function and process. The service provider contact uses the progress log to record and submit progress on deliverables. The progress can be logged by the activity that is performed, by the percentage of work that is completed, or by the completion of milestone activities that are defined for the project.

project transaction In PeopleSoft Project Costing, an individual transaction line that represents a cost, time, budget, or other transaction row.

promotion In PeopleSoft Promotions Management, a trade promotion, which is typically funded from trade dollars and used by consumer products manufacturers to increase sales volume.

prospects

In PeopleSoft Enterprise Campus Solutions, students who are interested in applying to the institution.

In PeopleSoft Enterprise Contributor Relations, individuals and organizations that are most likely to make substantial financial commitments or other types of commitments to the institution.

publishing

In PeopleSoft Enterprise Incentive Management, a stage in processing that makes incentive-related results available to participants.

rating components

In PeopleSoft Enterprise Campus Solutions, variables used with the Equation Editor to retrieve specified populations.

record group

A set of logically and functionally related control tables and views. Record groups help enable TableSet sharing, which eliminates redundant data entry. Record groups ensure that TableSet sharing is applied consistently across all related tables and views.

record input VAT flag

Abbreviation for *record input value-added tax flag.* Within PeopleSoft Purchasing, Payables, and General Ledger, this flag indicates that you are recording input VAT on the transaction. This flag, in conjunction with the record output VAT flag, is used to determine the accounting entries created for a transaction and to determine how a transaction is reported on the VAT return. For all cases within Purchasing and Payables where VAT information is tracked on a transaction, this flag is set to Yes. This flag is not used in PeopleSoft Order Management, Billing, or Receivables, where it is assumed that you are always recording only output VAT, or in PeopleSoft Expenses, where it is assumed that you are always recording only input VAT.

record output VAT flag

Abbreviation for *record output value-added tax flag.*

See *record input VAT flag.*

recname

The name of a record that is used to determine the associated field to match a value or set of values.

recognition

In PeopleSoft Enterprise Campus Solutions, the recognition type indicates whether the PeopleSoft Enterprise Contributor Relations donor is the primary donor of a commitment or shares the credit for a donation. Primary donors receive hard credit that must total 100 percent. Donors that share the credit are given soft credit. Institutions can also define other share recognition-type values such as memo credit or vehicle credit.

reference data

In PeopleSoft Sales Incentive Management, system objects that represent the sales organization, such as territories, participants, products, customers, channels, and so on.

reference object

In PeopleSoft Enterprise Incentive Management, this dimension-type object further defines the business. Reference objects can have their own hierarchy (for example, product tree, customer tree, industry tree, and geography tree).

reference transaction

In commitment control, a reference transaction is a source transaction that is referenced by a higher-level (and usually later) source transaction, in order to automatically reverse all or part of the referenced transaction's budget-checked amount. This avoids duplicate postings during the sequential entry of the transaction at different commitment levels. For example, the amount of an encumbrance transaction (such as a purchase order) will, when checked and recorded against a budget, cause the system to concurrently reference and relieve all or part of the amount of a corresponding pre-encumbrance transaction, such as a purchase requisition.

regional sourcing

In PeopleSoft Purchasing, provides the infrastructure to maintain, display, and select an appropriate vendor and vendor pricing structure that is based on a regional sourcing model where the multiple ship to locations are grouped. Sourcing may occur at a level higher than the ship to location.

relationship object In PeopleSoft Enterprise Incentive Management, these objects further define a compensation structure to resolve transactions by establishing associations between compensation objects and business objects.

remote data source data Data that is extracted from a separate database and migrated into the local database.

REN server Abbreviation for *real-time event notification server* in PeopleSoft MultiChannel Framework.

requester In PeopleSoft eSettlements, an individual who requests goods or services and whose ID appears on the various procurement pages that reference purchase orders.

reversal indicator In PeopleSoft Enterprise Campus Solutions, an indicator that denotes when a particular payment has been reversed, usually because of insufficient funds.

role Describes how people fit into PeopleSoft Workflow. A role is a class of users who perform the same type of work, such as clerks or managers. Your business rules typically specify what user role needs to do an activity.

role user A PeopleSoft Workflow user. A person's role user ID serves much the same purpose as a user ID does in other parts of the system. PeopleSoft Workflow uses role user IDs to determine how to route worklist items to users (through an email address, for example) and to track the roles that users play in the workflow. Role users do not need PeopleSoft user IDs.

roll up In a tree, to roll up is to total sums based on the information hierarchy.

run control A run control is a type of online page that is used to begin a process, such as the batch processing of a payroll run. Run control pages generally start a program that manipulates data.

run control ID A unique ID to associate each user with his or her own run control table entries.

run-level context In PeopleSoft Enterprise Incentive Management, associates a particular run (and batch ID) with a period context and plan context. Every plan context that participates in a run has a separate run-level context. Because a run cannot span periods, only one run-level context is associated with each plan context.

SCP SCBM XML message Abbreviation for *Supply Chain Planning Supply Chain Business Modeler Extensible Markup Language message.* PeopleSoft EnterpriseOne Supply Chain Business Modeler uses XML as the format for all data that it imports and exports.

search query You use this set of objects to pass a query string and operators to the search engine. The search index returns a set of matching results with keys to the source documents.

search/match In PeopleSoft Enterprise Campus Solutions and PeopleSoft Enterprise Human Resources Management Solutions, a feature that enables you to search for and identify duplicate records in the database.

seasonal address In PeopleSoft Enterprise Campus Solutions, an address that recurs for the same length of time at the same time of year each year until adjusted or deleted.

section In PeopleSoft Enterprise Incentive Management, a collection of incentive rules that operate on transactions of a specific type. Sections enable plans to be segmented to process logical events in different sections.

security event In commitment control, security events trigger security authorization checking, such as budget entries, transfers, and adjustments; exception overrides and notifications; and inquiries.

serial genealogy In PeopleSoft Manufacturing, the ability to track the composition of a specific, serial-controlled item.

serial in production In PeopleSoft Manufacturing, enables the tracing of serial information for manufactured items. This is maintained in the Item Master record.

service impact In PeopleSoft Enterprise Campus Solutions, the resulting action triggered by a service indicator. For example, a service indicator that reflects nonpayment of account balances by a student might result in a service impact that prohibits registration for classes.

service indicator In PeopleSoft Enterprise Campus Solutions, indicates services that may be either withheld or provided to an individual. Negative service indicators indicate holds that prevent the individual from receiving specified services, such as check-cashing privileges or registration for classes. Positive service indicators designate special services that are provided to the individual, such as front-of-line service or special services for disabled students.

session In PeopleSoft Enterprise Campus Solutions, time elements that subdivide a term into multiple time periods during which classes are offered. In PeopleSoft Contributor Relations, a session is the means of validating gift, pledge, membership, or adjustment data entry . It controls access to the data entered by a specific user ID. Sessions are balanced, queued, and then posted to the institution's financial system. Sessions must be posted to enter a matching gift or pledge payment, to make an adjustment, or to process giving clubs or acknowledgements.

In PeopleSoft Enterprise Learning Management, a single meeting day of an activity (that is, the period of time between start and finish times within a day). The session stores the specific date, location, meeting time, and instructor. Sessions are used for scheduled training.

session template In PeopleSoft Enterprise Learning Management, enables you to set up common activity characteristics that may be reused while scheduling a PeopleSoft Enterprise Learning Management activity—characteristics such as days of the week, start and end times, facility and room assignments, instructors, and equipment. A session pattern template can be attached to an activity that is being scheduled. Attaching a template to an activity causes all of the default template information to populate the activity session pattern.

setup relationship In PeopleSoft Enterprise Incentive Management, a relationship object type that associates a configuration plan with any structure node.

share driver expression In PeopleSoft Business Planning, a named planning method similar to a driver expression, but which you can set up globally for shared use within a single planning application or to be shared between multiple planning applications through PeopleSoft Enterprise Warehouse.

single signon With single signon, users can, after being authenticated by a PeopleSoft application server, access a second PeopleSoft application server without entering a user ID or password.

source key process In PeopleSoft Enterprise Campus Solutions, a process that relates a particular transaction to the source of the charge or financial aid. On selected pages, you can drill down into particular charges.

source transaction In commitment control, any transaction generated in a PeopleSoft or third-party application that is integrated with commitment control and which can be checked against commitment control budgets. For example, a pre-encumbrance, encumbrance, expenditure, recognized revenue, or collected revenue transaction.

speed key See *communication key.*

SpeedChart A user-defined shorthand key that designates several ChartKeys to be used for voucher entry. Percentages can optionally be related to each ChartKey in a SpeedChart definition.

SpeedType A code representing a combination of ChartField values. SpeedTypes simplify the entry of ChartFields commonly used together.

staging A method of consolidating selected partner offerings with the offerings from the enterprise's other partners.

standard letter code In PeopleSoft Enterprise Campus Solutions, a standard letter code used to identify each letter template available for use in mail merge functions. Every letter generated in the system must have a standard letter code identification.

statutory account Account required by a regulatory authority for recording and reporting financial results. In PeopleSoft, this is equivalent to the Alternate Account (ALTACCT) ChartField.

step In PeopleSoft Sales Incentive Management, a collection of sections in a plan. Each step corresponds to a step in the job run.

storage level In PeopleSoft Inventory, identifies the level of a material storage location. Material storage locations are made up of a business unit, a storage area, and a storage level. You can set up to four storage levels.

subcustomer qualifier A value that groups customers into a division for which you can generate detailed history, aging, events, and profiles.

Summary ChartField You use summary ChartFields to create summary ledgers that roll up detail amounts based on specific detail values or on selected tree nodes. When detail values are summarized using tree nodes, summary ChartFields must be used in the summary ledger data record to accommodate the maximum length of a node name (20 characters).

summary ledger An accounting feature used primarily in allocations, inquiries, and PS/nVision reporting to store combined account balances from detail ledgers. Summary ledgers increase speed and efficiency of reporting by eliminating the need to summarize detail ledger balances each time a report is requested. Instead, detail balances are summarized in a background process according to user-specified criteria and stored on summary ledgers. The summary ledgers are then accessed directly for reporting.

summary time period In PeopleSoft Business Planning, any time period (other than a base time period) that is an aggregate of other time periods, including other summary time periods and base time periods, such as quarter and year total.

summary tree A tree used to roll up accounts for each type of report in summary ledgers. Summary trees enable you to define trees on trees. In a summary tree, the detail values are really nodes on a detail tree or another summary tree (known as the *basis* tree). A summary tree structure specifies the details on which the summary trees are to be built.

syndicate To distribute a production version of the enterprise catalog to partners.

system function In PeopleSoft Receivables, an activity that defines how the system generates accounting entries for the general ledger.

TableSet A means of sharing similar sets of values in control tables, where the actual data values are different but the structure of the tables is the same.

TableSet sharing Shared data that is stored in many tables that are based on the same TableSets. Tables that use TableSet sharing contain the SETID field as an additional key or unique identifier.

target currency The value of the entry currency or currencies converted to a single currency for budget viewing and inquiry purposes.

tax authority In PeopleSoft Enterprise Campus Solutions, a user-defined element that combines a description and percentage of a tax with an account type, an item type, and a service impact.

template A template is HTML code associated with a web page. It defines the layout of the page and also where to get HTML for each part of the page. In PeopleSoft, you use templates to build a page by combining HTML from a number of sources. For a PeopleSoft portal, all templates must be registered in the portal registry, and each content reference must be assigned a template.

territory In PeopleSoft Sales Incentive Management, hierarchical relationships of business objects, including regions, products, customers, industries, and participants.

third party A company or vendor that has extensive PeopleSoft product knowledge and whose products and integrations have been certified and are compatible with PeopleSoft applications.

3C engine Abbreviation for *Communications, Checklists, and Comments engine.* In PeopleSoft Enterprise Campus Solutions, the 3C engine enables you to automate business processes that involve additions, deletions, and updates to communications, checklists, and comments. You define events and triggers to engage the engine, which runs the mass change and processes the 3C records (for individuals or organizations) immediately and automatically from within business processes.

3C group Abbreviation for *Communications, Checklists, and Comments group.* In PeopleSoft Enterprise Campus Solutions, a method of assigning or restricting access privileges. A 3C group enables you to group specific communication categories, checklist codes, and comment categories. You can then assign the group inquiry-only access or update access, as appropriate.

TimeSpan A relative period, such as year-to-date or current period, that can be used in various PeopleSoft General Ledger functions and reports when a rolling time frame, rather than a specific date, is required. TimeSpans can also be used with flexible formulas in PeopleSoft Projects.

trace usage In PeopleSoft Manufacturing, enables the control of which components will be traced during the manufacturing process. Serial- and lot-controlled components can be traced. This is maintained in the Item Master record.

transaction allocation In PeopleSoft Enterprise Incentive Management, the process of identifying the owner of a transaction. When a raw transaction from a batch is allocated to a plan context, the transaction is duplicated in the PeopleSoft Enterprise Incentive Management transaction tables.

transaction state In PeopleSoft Enterprise Incentive Management, a value assigned by an incentive rule to a transaction. Transaction states enable sections to process only transactions that are at a specific stage in system processing. After being successfully processed, transactions may be promoted to the next transaction state and "picked up" by a different section for further processing.

Translate table A system edit table that stores codes and translate values for the miscellaneous fields in the database that do not warrant individual edit tables of their own.

tree The graphical hierarchy in PeopleSoft systems that displays the relationship between all accounting units (for example, corporate divisions, projects, reporting groups, account numbers) and determines roll-up hierarchies.

tuition lock In PeopleSoft Enterprise Campus Solutions, a feature in the Tuition Calculation process that enables you to specify a point in a term after which students are charged a minimum (or *locked*) fee amount. Students are charged the locked fee amount even if they later drop classes and take less than the normal load level for that tuition charge.

unclaimed transaction In PeopleSoft Enterprise Incentive Management, a transaction that is not claimed by a node or participant after the allocation process has completed, usually due to missing or incomplete data. Unclaimed transactions may be manually assigned to the appropriate node or participant by a compensation administrator.

universal navigation header Every PeopleSoft portal includes the universal navigation header, intended to appear at the top of every page as long as the user is signed on to the portal. In addition to providing access to the standard navigation buttons (like Home, Favorites, and signoff) the universal navigation header can also display a welcome message for each user.

update access In PeopleSoft Enterprise Campus Solutions, a type of security access that permits the user to edit and update data.

See also *inquiry access.*

user interaction object In PeopleSoft Sales Incentive Management, used to define the reporting components and reports that a participant can access in his or her context. All Sales Incentive Management user interface objects and reports are registered as user interaction objects. User interaction objects can be linked to a compensation structure node through a compensation relationship object (individually or as groups).

variable In PeopleSoft Sales Incentive Management, the intermediate results of calculations. Variables hold the calculation results and are then inputs to other calculations. Variables can be plan variables that persist beyond the run of an engine or local variables that exist only during the processing of a section.

VAT exception Abbreviation for *value-added tax exception.* A temporary or permanent exemption from paying VAT that is granted to an organization. This terms refers to both VAT exoneration and VAT suspension.

VAT exempt Abbreviation for *value-added tax exempt.* Describes goods and services that are not subject to VAT. Organizations that supply exempt goods or services are unable to recover the related input VAT. This is also referred to as exempt without recovery.

VAT exoneration Abbreviation for *value-added tax exoneration.* An organization that has been granted a permanent exemption from paying VAT due to the nature of that organization.

VAT suspension Abbreviation for *value-added tax suspension.* An organization that has been granted a temporary exemption from paying VAT.

warehouse A PeopleSoft data warehouse that consists of predefined ETL maps, data warehouse tools, and DataMart definitions.

work order In PeopleSoft Services Procurement, enables an enterprise to create resource-based and deliverable-based transactions that specify the basic terms and conditions for hiring a specific service provider. When a service provider is hired, the service provider logs time or progress against the work order.

worker A person who is part of the workforce; an employee or a contingent worker.

workset A group of people and organizations that are linked together as a set. You can use worksets to simultaneously retrieve the data for a group of people and organizations and work with the information on a single page.

worksheet A way of presenting data through a PeopleSoft Business Analysis Modeler interface that enables users to do in-depth analysis using pivoting tables, charts, notes, and history information.

worklist The automated to-do list that PeopleSoft Workflow creates. From the worklist, you can directly access the pages you need to perform the next action, and then return to the worklist for another item.

XML link The XML Linking language enables you to insert elements into XML documents to create a links between resources.

XML schema An XML definition that standardizes the representation of application messages, component interfaces, or business interlinks.

XPI Abbreviation for *eXtended Process Integrator.* PeopleSoft XPI is the integration infrastructure that enables both real-time and batch communication with EnterpriseOne applications.

yield by operation In PeopleSoft Manufacturing, the ability to plan the loss of a manufactured item on an operation-by-operation basis.

zero-rated VAT Abbreviation for *zero-rated value-added tax.* A VAT transaction with a VAT code that has a tax percent of zero. Used to track taxable VAT activity where no actual VAT amount is charged. Organizations that supply zero-rated goods and services can still recover the related input VAT. This is also referred to as exempt with recovery.

Index

A

ActiveX, installing controls 208
additional documentation xxii
Analytic Domain Summary page 82
analytic engine type
  of analytic server instances 84
Analytic Instance Load/Unload page 93
analytic instances
  accessing 79
  clearing trees from dimensions 94
  copying 87, 91
  copying by analytic type 91
  copying by model name 92
  copying by server state 92
  creating 87, 88
  deleting 87, 89
  deleting by analytic type 90
  deleting by model name 90
  deleting by server state 90
  dimensions for attaching/detaching trees 94
  identifying delete tables 86
  identifying the tree effective date 94
  identifying the tree name 94
  identifying the tree override record 94
  identifying the tree set control value 94
  identifying the tree set ID 94
  importing from a file 96
  loading and unloading 92
  loading asynchronously 95
  saving tree information 95
  selecting trees to attach 94
  specifying import file directory 96
  specifying load or unload message name 95
  specifying load or unload time out 95
  specifying quantities 81
  specifying search criteria for copying 92
  specifying search criteria for deleting 90
  specifying the tree discard level 95
  specifying the tree node 94
  specifying the tree start level 95
  understanding batch processing 80
  using secondary database connection 79
Analytic Model Viewer 93
analytic server domains
  administering 82
  identifying active analytic server instances 82
  identifying analytic server instances being loaded 82
  identifying analytic server instances marked as terminated 83
  identifying available analytic server instances 82
  identifying idle analytic server instances 82
  identifying machine name 82
  viewing active domain name 82
analytic server framework, *See* PSANALYTICSRV
analytic server instances
  administering 83
  identifying last user operation 85
  identifying latest operation 85
  identifying machine name 85
  identifying process identifier 85
  identifying registration date and time 85
  identifying the end time of latest operation 85
  identifying the start time of latest operation 85
  identifying time loaded 85
  identifying user ID 85
  search criteria 84
  shutting down 85
  terminating 85
  viewing by analytic engine type 84
  viewing by analytic type 84
  viewing by loaded analytic instance 84
  viewing by server state 84
analytic servers
  abnormal process termination 79
  accessing analytic instances 79
  administering domains 82
  administering instances 83

analytic engine 77
analytic engine type 77
analytic instance 77
analytic server instance 77
analytic type 77
configuring and starting 80
enabling PSANALYTICSRV 80
error handling behavior 79
managing 75
specifying instance quantities 81
starting PSANALYTICSRV 81
understanding 2, 77
understanding features 77
understanding process flow and behavior 78
understanding the framework 75
understanding the terms 77
using secondary database connection 79
Analytic Servers page 83
analytic tables
purging delete tables 85
synchronizing versions 86
analytic type
identifying delete tables 86
of analytic instances 90, 91
of analytic server instances 84
specifying for an analytic instance 88
Apache HTTP, configuring as an RPS 111
application class method
used with an analytic type 89
Application Designer 260
Application Engine, *See* PeopleSoft Application Engine
application fundamentals xxi
application package
used with an analytic type 89
application servers
configuring for cache/replay files 25
configuring PeopleCode debugging 52
enabling multiple server processes for debugging 259
loading the cache 227
replicating environments 273
running as a non-root user 146
setting database options 45
setting domain parameters 43
setting Integration Broker options 70
setting interface driver options 67
setting JRAD options 50
setting messaging server options 63
setting PSANALYTICSRV options 60
setting PSAPPSRV options 59
setting PSPPMSRV options 71
setting PSQCKSRV options 61
setting PSQRYSRV options 62
setting PSRENSRV options 71
setting PSSAMSRV options 60
setting PSTOOLS options 67
setting remote call options 58
setting security options 46
setting server process options 72
setting startup options 44
setting up the PeopleSoft Windows service 38
setting workstation listener options 46
sharing indexes with Process Scheduler 178
spawning 44
specifying cache settings 56
specifying domain settings 50
specifying search index file location 70
specifying SMTP settings 64
specifying timeouts 325
specifying trace options 53
specifying workstation settings 199
using PSADMIN 7
using PSADMIN menus 27
using Tuxedo connect string 201
attach/detach trees
clearing for analytic instance dimensions 94
identifying the override record 94
saving tree information 95
selecting for analytic instance dimensions 94
specifying the discard level 95
specifying the start level 95
specifying the tree node 94
viewing dimensions 94
viewing the effective date 94
viewing the set control value 94
viewing the set ID 94
viewing the tree name 94
auditing
database-level utilities 250
enabling database-level auditing 45
record cross-references 247

B

BEA Jolt
configuring 72
listener, *See* Jolt listener
BEA JRAD, *See* JRAD
BEA JRLY, *See* JRLY
BEA Tuxedo
booting domains 24
restarting server processes 51
running the Process Scheduler server 38
setting compression 47
setting the queue size 60
setting the spawn threshold 51
setting timeouts 47
BEA WebLogic
accessing the server console 99
adjusting JVM heap size 125, 303, 306
changing Execute thread pool size 307
changing multi-server listen ports 311
changing user passwords 114
common default domain settings 285
configuration files 310
configuring as an RPS 105
configuring SSL keys 122
converting SSL keys and certificates 118
determining the service pack level 126
distributed managed server 284
domain directories and files 292
domain topology 289
enabling RPS load balancing 106
enabling/disabling HTTP access log 127
enabling/disabling HTTP Keep-Alive 113
Gather utility 246
implementing SSL 115
importing SSL keys and certificates to the keystore 120
importing the server certificate using ImportPrivateKey 121
importing the server certificate using Pskeymanager 121
in a single-server domain 280
installing PIA 298
managed server architecture 279
monitoring Execute thread pool size 303
monitoring portal HTTP session count 309
monitoring resources 303
monitoring sessions via WebLogic Server 8.1 console 102
multi-server domain 281
multi-server example 315
obtaining SSL encryption keys 116
password protecting the private key 120
preparing SSL keys and certificates 118
restricting access to servlets 123
running as a Windows service 115
server analogies 288
server certificate chain of trust 119
server life cycle 299
session cookie name format 98
setting up an RPS 103
setting up HTTP session timeouts 113
single-server example 312
starting 99
starting and stopping a distributed managed server 303
starting and stopping multiple servers 300
starting and stopping PIA 299
starting on UNIX 100
starting on Windows 100
stopping 101
tuning performance 303
understanding SSL 116
understanding the PeopleSoft domain 97
web applications 279
browsers, enabling compression 215

C

cache files, workstation settings 185
caching
configuring application servers for cache files 25
enabling 57
loading application server caches 227
running LOADCACHE 229
setting cache file location 57
setting memory maximum 57
setting server caching mode 57
specifying settings 57
CAs, *See* certificate authorities (CAs)
certificate authorities (CAs)

adding for PeopleTools testing 141
sending certificate requests 141
character sets
checking data field length 222
setting for SMTP servers 65
setting for trace-log files 52
setting for Verity engines 253
setting the codepage for Microsoft applications 253
setting the default 253
specifying for data processing 68
client
setting up 195
setting up in Configuration Manager 211
setup in Configuration Manager 209
collections
creating 168
opening 167
column headings, customizing 233
command-line
configuring domains 12
creating domains 12
specifying options 207
understanding the PSADMIN interface 11
using options for PeopleSoft Process Scheduler 38
using PSADMIN options 11
comments, submitting xxvi
common elements xxvii
components, applying defaults 240
compression
enabling for browsers 215
setting for Tuxedo 47
setting the Jolt compression threshold 49
setting the message compression threshold 70
configuration files
archiving application server 24
configuring JRLY 266
editing for PeopleSoft Process Scheduler 37
editing PSNTSRV.CFG 41
replicating Process Scheduler 274
using PSADMIN 19
Configuration Manager
running client setup 209, 211
setting signon defaults 182
specifying command line options 207
specifying display settings 186
specifying startup settings 182
starting 182
understanding 181
using folder tabs 181
using shortcut links 196
using the Client Setup tab 195
using the Common tab 206
using the Crystal/Bus Interlink tab 189
using the Database/Application Server tab 199
using the Display tab 186
using the Import/Export tab 197
using the nVision tab 204
using the Process Scheduler tab 202
using the Profile tab 198
using the Remote Call/AE tab 194
using the Trace tab 191
using the Workflow tab 192
Configure Keystores page 122
configuring a workstation 181
connect ID, default password 185
connectivity, verifying 210
contact information xxvi
Copy Analytic Instance page 91
Copy File Archive page 243
Create Analytic Instance Page 88
cross-references xxv
Crystal reports 208
Customer Connection website xxii

D

Data Mover, workstation settings 207
database
database-level auditing 250
disabling Oracle FIRST_ROWS hint 45
displaying the name 185, 188
enabling database-level auditing 45
entering a name 184
initiating local connection to PeopleSoft database on the same machine 45
selecting the type 185
setting environment variables 45
setting sign-in values 44
setting Sybase TCP packet size 45
setting the default type 184
setting up remote database connections 241

specifying connect IDs 44
specifying name 44
specifying type 44
starting Process Scheduler servers automatically 41
validating signon 46
DB2 z/OS, defining tablespaces 230
DDL 231
debugging
configuring for PeopleCode 52
enabling for PeopleCode 72
enabling multiple PSAPPSRV server processes 259
entering Debugger mode 260
reproducing crashes 56
requesting PSDBGSRV server processes 259
setting the PSDBGSRV listener port 258
setting up PeopleCode Debugger 257
tracing PeopleCode 260
tracing SQL 261
using three-tier connections 258
using two-tier connections 257
using utilities 250
Delete Analytic Instance page 89
delete tables
purging 85
viewing for analytic types and instances 86
Design a Search Index page 168
detach directory 193
development environment 209
dirty-reads, using 63
display
adjusting page sizes 187
specifying settings 186
using the Font options 188
distributed managed server 284
documentation
printed xxii
related xxii
updates xxii
domains
administering 28
allowing dynamic changes 51
booting 24, 28
checking status 29
configuring 21, 32
configuring as a Windows service 39
configuring via command-line 12
creating 35
creating via command-line 12
deleting 35
entering network tracing level 52
forcing shutdown 29
loading configurations 23
logging 52
monitoring 24
purging the cache 30
restarting server processes 51
setting application server domain parameters 43
shutting down 28
specifying IDs 50
specifying settings 50
specifying the spawn threshold 51
specifying trace-log character set 52
specifying your database connectivity software directory 50
starting automatically 41
starting PSPPMSRV servers 73
stopping 24
understanding the PeopleSoft domain 97
using PSADMIN 7

E

encryption
enabling for Jolt listener 48
enabling for workstation listener 46
Enrollment page 142
environment
reconfiguring Environment Management components 275
replicating application servers 273
replicating Process Scheduler 274
replicating web servers 272
solving replication problems 272
understanding replication 271
Environment Management
reconfiguring agents 275
reconfiguring the hub 276
reconfiguring the viewer 276
replicating components 275
environment replication 271
environment variables
accessing class libraries 67
improving system performance 45
errors

configuring access/error logging for internal web server HTTP transport 146
logging 56
notifying by mail 56
suppressing application error messages 69
suppressing SQL error messages 70
event notifications 73
export Configuration Manager 197

F

failover 201
file system indexes
building 164, 171
defining what to index 172
setting options 171
understanding 162
Filesystem Options page 171
functions, XML link function registry 234

G

Gather utility
application server data 245
environmental data 245
getting started 244
in UNIX 245
in Windows 245
including additional files 247
understanding 244
web server data 246
globalization, *See* international settings
glossary 331

H

handlers
specifying Jolt handler quantity 48
specifying maximum clients per Jolt handler 48
specifying maximum clients per workstation handler 47
specifying maximum Jolt handler quantity 48
specifying maximum workstation handler quantity 47
specifying workstation handler quantity 47
help
setting PeopleTools options 223
viewing the Sytem Information help page 214
high availability 201
HTTP Gateway page 174
HTTP Keep-Alive 113
HTTP servers 133
configuring Apache HTTP as an RPS 111
disabling directory browsing in IBM HTTP 147, 148
setting up 132
setting up SSL 139
testing SSL 143
HTTP spider indexes
building 164, 174
defining gateway settings 174
defining what to index 175
understanding 162
httpd.conf file, editing 142

I

IBM WebSphere
accessing InfoCenter 159
adding JVM system properties 145
adding ports to default_host 145
changing HTTP ports 144
changing the JVM heap 145
configuring access/error logging for internal web server HTTP transport 146
configuring the HTTP server for RPS 136
creating servers 145
deploying multiple PIA instances 151
Gather utility 246
installing additional PIA to an instance 150
installing and uninstalling features 129
installing HTTP Server and RPS plug-in 133
installing RPS plug-in for Microsoft IIS 135
installing RPS plug-in for Sun ONE 135
installing RPS plug-in only 134
regenerating the web server plug-in 145
reserving PIA ports 158
setting keep-alive 146
setting up a machine for RPS 133
setting up RPS 132

setting up SSL 137
setting up SSL on HTTP server 139
setting Verbose GC 144
starting/stopping the administrative console 143
starting/stopping the server 144
troubleshooting 144
understanding RPS 133
uninstalling 130
uninstalling embedded messaging 131
uninstalling PIA 150
using multiple base instances 152
using silent install 129
using single base instance 154
using the administration console 143
working with 129
images
converting 207
creating 56
ImportPrivateKey, *See* SSL
indexes, *See* search indexes
Informix, specifying server names 45
installation
setting up the development environment 209
using workstations 196
Integration Broker
enabling profiling 70
setting maximum message display size 70
setting message size compression threshold 70
interface driver 67
international settings
administering time zones 252
managing multiple languages 252
preferences 251
sizing process field 252
iPlanet
configuring as an RPS 107
using the plug-in 110

J

J2EE web applications, *See* web applications
Java
adding to CLASSPATH 67
adjusting JVM heap size 125, 303, 306
Java VM options 67
Jolt, *See* BEA Jolt
Jolt Internet Relay 263
See Also JRLY
example 264
implementation considerations 265
understanding the architecture 263
Jolt listener
assigning a port 48
configuring 72
configuring BEA Jolt 50
enabling encryption 48
setting client connection modes 49
setting client connection request binding time 49
setting the address 48
setting the compression threshold 49
setting timeout 49
specifying handler quantity 48
specifying maximum clients per handler 48
specifying maximum handler quantity 48
understanding 48
Jolt Relay, *See* JRLY
Jolt Relay Adapter, *See* JRAD
JRAD
configuring 72, 268
setting the listener address 50
setting the listener port 50
understanding 50, 263
JRLY
assigning the JRAD listener port 50
configuring 266
configuring JRAD 72
running 269
running on UNIX 270
running on Windows 269
understanding 263
using the administration program 269
JVM heap
changing 145
monitoring 303
sizing 125, 303, 306
JVM system properties 145

K

Keep-Alive, HTTP 113, 146
keys
creating key files for SSL 140
modifying VdkVgwKey 179
updating keystore properties 122

keystores, updating properties 122

L

language preference 187
languages
  managing 252
  specifying PeopleTools settings 219
  using the spell check system dictionary 225
latest operation
  viewing for analytic server instances 85
latest operation start time
  viewing for analytic server instances 85
Load Application Server Cache page 229
load balancing 106
LOADCACHE 229
loading and unloading 92
Log Analyzer 149
logging
  configuring access/error logging for internal web server HTTP transport 146
  editing log files 33
  enabling for optimization engine components 55
  enabling/disabling HTTP access log 127
  setting severity level for PSRENSRV process 71
  setting the level for SQL tracing for all clients 53
  setting the level for SQL tracing for individual clients 53
  tracing email details to log file 65
  using the Log Analyzer 149
  viewing crash log information 56
  writing error information 56
look-up pages 60

M

machine name
  for analytic server domains 82
  viewing for analytic server instances 85
Manage Installed Languages page 252
MCF servers, configuring 73
menus
  using PSADMIN 27
  using Quick-Configure 10
merchant integration 235
Message Catalog page 223
Message Monitor 70
messaging
  adding/maintaining system messages 223
  setting server options 63
metadata caching 227
Microsoft Internet Information Server (ISS) 103
Microsoft Windows services
  administering PeopleSoft services 41
  configuring the PeopleSoft service 39
  editing PSNTSRV.CFG 41
  monitoring the executables 40
  understanding 38
MIME indexing 164, 166, 173
MMA Partners xxii
Mobile Applications 244
model name
  of analytic instances 90, 92
MS IIS, configuring as an RPS 103
Multipurpose Internet Mail Extensions, *See* MIME indexing

N

navigator 189
network tracing 52
non proxy hosts
  specifying for PSTOOLS 68
notes xxv
notifications
  configuring event 73
  designing real time event notification (REN) 71
  logging error 56
nVision workstation settings 204

O

ODBC driver 208
operator
  specifying signon settings 182
  using overrides 185
optimization
  configuring engines 73
  logging for engine components 55
  utilities 254
Options page, converting panels to pages 238
Oracle

disabling the FIRST_ROWS hint 45
initiating local connection to PeopleSoft database 45

P

pages
converting from panels 237
displaying 188
displaying in navigator 188
panels, converting to pages 237
parameter record fields
populating for copying an analytic instance 92
populating for creating an analytic instance 89
populating for deleting an analytic instance 91
passwords
changing for WebLogic users 114
specifying for connect IDs 44
specifying for users 44
PeopleBooks
ordering xxii
PeopleCode
configuring debugging for 52
enabling debugging 72
setting up the Debugger 257
using the debugger 260
PeopleCode trace settings 191
PeopleCode, typographical conventions xxiv
PeopleSoft Application Designer, *See* Application Designer
PeopleSoft Application Engine
activating program tracing 55
running LOADCACHE 227
PeopleSoft application fundamentals xxi
PeopleSoft application server process, *See* PSAPPSRV
PeopleSoft Mobile Applications, *See* Mobile Applications, PeopleSoft
PeopleSoft Open Query 196
PeopleSoft Performance Monitor
disabling the agent 67
enabling the agent 56
starting PSPPMSRV servers 73
PeopleSoft Ping Chart page 256
PeopleSoft Ping page 254
PeopleSoft Process Scheduler
configuring servers 36, 37
creating servers 37
deleting servers 37
editing the configuration file 37
replicating environments 274
running servers as standalone or Tuxedo-controlled 38
setting up the PeopleSoft Windows service 38
sharing indexes with application servers 178
starting servers 36, 38
starting servers automatically 41
stopping servers 36, 38
timeouts 328
using command-line options 38
using menu options 35
working with analytic server framework 80
PeopleTools
adding system messages 223
converting panels to pages 237
copying file attachments 243
debug utilities 250
grouping records 235
international utilities 251
maintaining system messages 223
maintaining URLs 242, 244
optimization utilities 254
setting general options 219
setting help options 223
setting up remote databases 241
specifying language settings 219
testing 141
using administration utilities 216
using audit utilities 247
using DDL Model Defaults page 231
using Gather utility 244
using merchant integration utilities 235
using PeopleSoft Ping 254
using Sync ID utilities 244
using tablesets 235, 236
using Tablespace Utility 230
using the spell check system dictionary 225
using translate values 227
using update utilities 240
using XML Link Function Registry 234
utilities 213
PeopleTools Test Utilities page 250
performance collators, configuring 73

Performance Monitor, *See* PeopleSoft Performance Monitor
permission lists
adding user IDs 44
applying timeouts 329
personal certificate requests
adding from Enrollment page 142
creating 141
sending to CAs for enrollment 141
PIA
installing additional PIA to WebSphere instance 150
installing on WebLogic 298
reserving WebSphere PIA ports 158
starting and stopping on WebLogic 299
understanding multiple instances 151
uninstalling on WebSphere 150
using a single WebSphere base instance 154
using multiple instances on WebLogic 315
using multiple instances on WebSphere 5.1.1 151
using multiple WebSphere base instances 152
using single instance on WebLogic 312
using WebSphere ND 156
pinging
charting 256
using PeopleSoft Ping 254
plug-ins
configuring web server 139
iPlanet 107
regenerating 145
using Apache HTTP server 111
using iPlanet 110
popup menus 188
portals
monitoring HTTP session count on WebLogic 309
search technology 162
ports
adding ports to default_host 145
assigning for Jolt listener 48
assigning the JRAD listener port 50
assigning the REN servers HTTP port 71
assigning the workstation listener port number 46
changing WebSphere HTTP ports 144
reserving WebSphere PIA ports 158
setting the PSDBGSRV listener port 258
specifying for failover mail servers 65
specifying for mail servers 64
specifying the proxy server port 68
using PeopleCode Debugger 257
WebSphere 159
prerequisites xxi
printed documentation xxii
process identifier
viewing for analytic server instances 85
Process Scheduler, *See* PeopleSoft Process Scheduler
Process Scheduler workstation settings 202
proxy servers
configuring Apache HTTP as an RPS 111
setting up 103
specifying for PSTOOLS 67
specifying the port 68
PS_HOME Access 210
PSADMIN
administering application servers 27
administering domains 28
archiving configuration files 24
booting domains 24, 28
checking domain status 29
configuration templates 9
configuring application servers 27
configuring domains 21, 32
configuring PeopleCode debugging 52
configuring the PeopleSoft service 39
creating domains 35
deleting domains 35
editing configuration and log files 33
forcing domain shutdown 29
loading configurations 23
monitoring domains 24
purging the domain cache 30
setting application server domain parameters 43
setting database options 45
setting database sign-in options 44
setting Integration Broker options 70
setting interface driver options 67
setting JRAD options 50
setting messaging server options 63
setting PSANALYTICSRV options 60

setting PSAPPSRV options 59
setting PSPPMSRV options 71
setting PSQCKSRV options 61
setting PSQRYSRV options 62
setting PSRENSRV options 71
setting PSSAMSRV options 60
setting PSTOOLS options 67
setting remote call options 58
setting security options 46
setting server process options 72
setting workstation listener options 46
shutting down domains 28
specifying cache settings 56
specifying domain settings 50
specifying search index file location 70
specifying SMTP settings 64
specifying trace options 53
starting 8
stopping domains 24
understanding 1, 7
using 9
using command-line options 11
using executables and configuration files 19
using menu options 27
using Process Scheduler menu 35
using Quick-Configure menu 10
PSANALYTICSRV
administering domains 82
enabling 80
setting server process options 60
setting the idle timeout 60
setting the maximum number of analytic server instances 60
setting the minimum number of analytic server instances 60
starting 81
PSAPPSRV
enabling prompting on look-up pages 60
moving services into PSQCKSRV 72
setting PSQCKSRV options 61
setting the maximum fetched row storage 59
setting the maximum number of servers 59
setting the number of startup servers 59
setting the service failure threshold for server process restarts 59
setting the service request threshold for server termination/restart 59
setting the service request wait time 59
setting the Tuxedo queue size 60
understanding 59
Pskeymanager, *See* SSL
PSPPMSRV
setting options 71
starting 73
PSPRCS.CFG, editing 37
PSQCKSRV
moving PSAPPSRV services into 72
setting server process options 61
setting the maximum fetched row storage 62
setting the maximum number of servers 62
setting the number of startup servers 62
setting the request wait time 62
setting the service failure threshold for server process restarts 62
setting the service request threshold for server termination/restart 62
PSQRYSRV
enabling reading uncommitted data 63
moving long-running queries into 72
setting options 62
setting the maximum number of servers 63
setting the maximum result set size 63
setting the number of startup servers 62
setting the request wait time 63
setting the service failure threshold for server process restarts 63
setting the service request threshold for server termination/restart 63
PSRENSRV
assigning the REN servers HTTP port 71
configuring event notification 73
setting the application server domain name 71
setting the log severity level 71
setting the TCP buffer size 71
understanding 71
PSSAMSRV
setting server process options 60
setting the maximum fetched row storage 61

setting the maximum number of servers 61
setting the number of startup servers 61
setting the request wait time 61
setting the service failure threshold for server process restarts 61
setting the service request threshold for server termination/restart 61
PSTOOLS
accessing class libraries 67
disabling the Performance Monitor agent 67
enabling/disabling %UpdateStats 69
setting advanced configuration parameters 67
specifying character sets 68
specifying Java VM options 67
specifying proxy servers 67
specifying servers for direct connection 68
specifying the proxy server port 68
suppressing application error messages 69
suppressing SQL error messages 70
pub/sub servers
booting 72
configuring 72
Purge Delete Tables page 85

Q

queries
moving long-running queries into PSQRYSRV 72
setting PSQRYSRV options 62

R

RDA 241
real time event notification (REN) 71
record with parameters
specifying for copying an analytic instance 92
specifying for creating an analytic instance 89
specifying for deleting an analytic instance 91
record-based indexes
adding subrecords 171
building 164, 168
modifying properties 168
understanding 162
records
building record-based indexes 168
grouping 235
registration date and time
viewing for analytic server instance 85
viewing for analytic server instances 85
registry settings 181
related documentation xxii
remote call
options 58, 194
remote data access (RDA) 241
remote databases
ensuring security 242
setting up connections 241
REN 71
Resource Analyzer 149
reverse proxy servers, *See* RPS
RPS
configuring Apache HTTP 111
configuring iPlanet 107
configuring MS IIS 103
configuring the WebSphere HTTP server 136
configuring WebLogic 105
installing HTTP Server and RPS plug-in 133
installing RPS plug-in for Microsoft IIS 135
installing RPS plug-in for Sun ONE 135
installing RPS plug-in only 134
load balancing 106
setting up 103
setting up a WebSphere machine 133
setting up with WebSphere 132
understanding WebSphere application 133

S

Scope page, converting panels to pages 237
SCP server 67
search indexes
administering 175, 177
building and maintaining 161
common controls 166
creating collections 168
editing properties 177
file system 162

See Also file system indexes
HTTP spider 162
See Also HTTP spider indexes
MIME types 166
modifying VdkVgwKey 179
on z/OS 164
opening collections 167
portal technologies 162
record-based 162
See Also record-based indexes
record-based indexes 168
scheduling administration 178
search architecture 162
search utilities 163
sharing between application servers and Process Scheduler 178
specifying file locations 70
specifying locations 176
types 162
understanding 3, 161
understanding limitations 164
understanding searches 165
verity technologies 163
searches, *See* search indexes
security
adding user IDs to permission lists 44
implementing SSL on WebLogic 115
implementing SSL on WebSphere 137
restricting access to servlets 123
setting sign-in options 46
using remote databases 242
validating signon with database 46
self-signed certificate 137
server state
of analytic instances 90, 92
of analytic server instances 84
servers
analytic server features 77
analytic server terms 77
creating 145
managing analytic servers 75
replicating environments 271
setting process options 72
setting PSPPMSRV options 71
setting up the PeopleSoft Windows service 38
specifying names 45
understanding the framework 75
using PSADMIN 7
service packs
determining the level 126
viewing system information 215
service start delay 41
servlets, restricting access 123
session cookies
WebLogic name format 98
session timeouts, *See* timeouts
shortcut links 196
signon
defaults 183
setting defaults 182
using connect ID 185
using default application server 184
using default database name 184
using default database server 184
using default operator ID 184
using operator overrides 185
Simple Mail Transfer Protocol, *See* SMTP
SMTP
delivering TriggerBusinessEvent email via messaging system 65
enabling send times for messages 66
further considerations 66
specifying character set of sender's machine 65
specifying DLL for translating mail 65
specifying failover mail server host name and IP address 64
specifying failover mail server port 65
specifying mail server host name and IP address 64
specifying mail server port 64
specifying reply internet address for Blackberry email 65
specifying sender's internet address 65
specifying sender's source machine 65
specifying settings 64
time to wait for result 66
tracing email details 65
user name for authentication 66
user name for failover 66
user password for authentication 66
user password for failover 66
spawning application servers 44
setting the threshold 51
Spell Check System Dictionary page 225
spell checking 225
case sensitivity 226
table structure for word storage 226
spider indexes 162

See Also file system indexes; HTTP spider indexes
SQL
specifying trace settings 191
tracing 261
SQRs
customizing column headings 233
SSL
configuring WebLogic keys 122
converting keys and certificates for WebLogic use 118
implementing certificates 115
implementing keys 115
importing keys and certificates to the WebLogic keystore 120
importing the server certificate using ImportPrivateKey 121
importing the server certificate using Pskeymanager 121
obtaining encryption keys for WebLogic 116
preparing keys and certificates for WebLogic 118
server certificate chain of trust 119
setting up on IBM HTTP server 139
setting up on WebSphere 137
testing on IBM HTTP server 143
used with WebLogic 116
WebLogic private key 120
strings table 233
Structured Query Reports, *See* SQRs
suggestions, submitting xxvi
Supply Chain Planning (SCP) server 67
supporting applications, verifying 210
Sybase
setting TCP packet size 45
specifying server names 45
Synchronize Table Versions page 86
System Information page 213

T

tables
maintaining URLs 242, 244
sharing 235
storing words 226
translate 227
tablesets
controlling 236
creating IDs 235
trees 237
tablespaces
adding SQL space 230
deleting SQL space 230
managing 231
renaming SQL space 231
using Tablespace Utility 230
templates, configuration 9, 35
terminating analytic server instances 85
terms 331
test utilities 250
testing
adding certificate authorities for PeopleTools 141
using WebSphere setup 139
three-tier workstation settings 199
time loaded
viewing for analytic server instances 85
Time Zones utility 252
timeouts
application servers 325
PIA 329
Process Scheduler 328
setting analytic instance idle time 60
setting client connection idle time 47
setting client connection request binding time 47
setting for HTTP sessions 113
setting the PSAPPSRV wait time for server requests 59
settings 323
web servers 323
Trace PeopleCode page 260
Trace SQL page 261
tracing
activating for Application Engine programs 55
configuring PeopleCode 260
enabling for email details 65
enabling memory image creation for crashes 56
enabling Performance Monitor agent 56
logging error reports 56
logging for optimization engine components 55
mailing error notifications 56
reproducing crashes 56
setting in Configuration Manager 191
setting the level for all client-generated activity 53

setting the logging level for all clients 53
setting the logging level for individual clients 53
specifying options 53
specifying trace options to be written to the trace file 54
specifying trace-log character set 52
SQL 261
using page processor activity 54
viewing crash log information 56
Translate Values page 227
translating languages 219
troubleshooting IBM WebSphere 144
Tuxedo, *See* BEA Tuxedo
Tuxedo connect string 201
typographical conventions xxiv

U

uninstall workstation 208
Update utilities 240
upgrade, image conversion 207
URLs
adding new entries 242
maintaining 242, 244
user ID 185
specifying 44
specifying the default 184
viewing for analytic server instances 85
user operation
viewing for analytic server instances 85

V

variables, *See* enviroment variables
Verbose GC 144
Verity
search index limitations 164
search technologies 163
setting the character set 253
virtual hosting
understanding multiple PIA instances 151
visual cues xxv
vspider
building file system indexes 171
building HTTP spider indexes 174
building indexes 164

W

warnings xxv
web applications 279
changing server targets on WebLogic 311
descriptor files 296
using Console 279
using HttpClusterServlet 279
using HttpProxyServlet 279
web container
generating a self-signed certificate 137
modifying to support SSL 139
web servers
Gather utility 246
installing and uninstalling WebSphere features 129
replicating environments 272
setting PSRENSRV options 71
timeouts 323
understanding 2
uninstalling WebSphere 130
uninstalling WebSphere embedded messaging 131
using WebSphere silent install 129
WebLogic 279
working with WebLogic 97
working with WebSphere 129
WebLogic, *See* BEA WebLogic
domain types 280
WebSphere, *See* IBM WebSphere
WebSphere ND, using 156
WebSphere ports 159
What to Index page 172
Windows services, *See* Microsoft Windows services
workflow Configuration Manager settings 192
workstation listener
assigning the port number 46
enabling encryption 46
setting client connection request binding time 47
setting the address 46
setting timeout 47
setting Tuxedo compression 47
specifying handler quantity 47
specifying maximum clients per handler 47

specifying maximum handler quantity 47
understanding 46
workstations
importing and exporting settings 197
installing 195
setting up 181
uninstalling 208

X

XML Link Function Registry 234

Z

z/OS
defining passwords/user IDs 46
PeopleCode debugging 257