HP Project and Portfolio Management Center

Software Version: 9.10

Installation and Administration Guide

Document Release Date: April 2011, Second Edition
Software Release Date: September 2010

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notices

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

Intel®, Intel® Itanium®, Intel® Xeon®, and Pentium® are trademarks of Intel Corporation in the U.S. and other countries.

Java™ is a U.S. trademark of Sun Microsystems, Inc.

Microsoft®, Windows®, Windows® XP, and Windows Vista® are U.S. registered trademarks of Microsoft Corporation.

Oracle® is a registered trademark of Oracle Corporation and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

Why version number 9.10?

PPM Center is an integrated part of the HP BTO Operations version 9 portfolio. Using this version number aligns PPM Center with other products that are releasing in the same time frame. PPM Center 9.10 builds on PPM Center 8.0x and is an extension of that product version family. Product releases within the HP BTO Operations version 9 portfolio will feature shared technology, common platforms, integrations, solutions, upgrade tools, and professional services offerings.

Documentation Updates

This manual's title page contains the following identifying information:

- Software version number, which indicates the software version
- Document release date, which changes each time the document is updated
- Software release date, which indicates the release date of this version of the software

To check for recent updates, or to verify that you are using the most recent edition of a document, go to h20230.www2.hp.com/selfsolve/manuals. You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

The following table lists changes made to this document.

Summary of Changes
Chapter 3 • Added a new section titled "Default Permissions for PPM Center Schemas" was added to the section "Preparing to Install PPM Center" • A note was added to the section "Preparing to Install PPM Center" to inform users about limits on physical memory for Windows Server 2003 Standard Edition. • In the "Creating the Database Schemas" section, a step was added ■ A section titled "Assigning Ports Outside of the Ephemeral Port Range" was added to the "Verifying Port Availability" section.
Chapter 4 ■ The section "(UNIX only) Setting the ulimit Value" was added to the chapter. ■ A section titled "Additional Considerations for Configuring Secure Web Logon" was added to the "Configuring Secure Web Logon" section.
Chapter 5 ■ The document now clearly states that a server cluster setup is mandatory for a PROD system. ■ The descriptions provided for the APP_SERVER_MULTICAST_PORT and MULTICAST_PORT parameters in Table 5-2 were modified to reflect their true functionality. ■ The example server.conf file was modified to correct inaccuracies.
Chapter 6 • The following note was added to the overview: The user experience logging off of a PPM Center instance depends on the SSO plug-in implemented. If a user logs off of a PPM Center instance that is integrated with SiteMinder, he is logged out of both PPM Center and SiteMinder, and does not need to close the browser tab. If a user logs off of a PPM Center instance that is integrated with a plug-in that does not support log-off, users signing out from PPM Center are directed to close the browser window in order to log off. ■ The procedure for Integrating PPM Center with an LDAP Server was corrected to indicate that valid values for the AUTHENTICATION_MODE parameter are ITG and LDAP. ■ The section titled "Disabling the LDAP_URL parameter" was added to accurately describe PPM Center behavior if you specify values for both LDAP_URL and LDAP_URL_FULL parameters.

Summary of Changes
Chapter 7 ■ A new section titled “Changing PPM Center Data” was added to advise against changing PPM Center master data directly in the database.
Chapter 9 ■ The section “Setting Stream Encoding for an Environment” was added to the “Environment Considerations” section of the chapter. ■ A description of the Workflow Migrator behavior option “Replace existing sub workflow?” was added to the list of behavior options.
Chapter 10 ■ The last several steps of the procedure described in “Migrating to a Windows Machine” were changed, and now include instructions on how to use KSVC.exe.
Appendix B ● The DASHBOARD_PAGE_AUTO_REFRESH_DISABLED parameter description was moved to the table that lists descriptions of Dashboard-related parameters. ■ The JSP_RECOMPILE_ENABLED parameter description was modified to indicate that a PPM Server detects JSP changes only after it is restarted.

Support

You can visit the HP Software Support Web site at: hp.com/go/hpsoftwaresupport

HP Software Support Online provides an efficient way to access interactive technical support tools. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most support areas require that you register as an HP Passport user and sign in. Many also require a support contract.

To find more information about access levels, go to: h20230.www2.hp.com/new_access_levels.jsp

To register for an HP Passport ID, go to: h20229.www2.hp.com/passport-registration.html

Contents

1 Introduction

Administering HP Project and Portfolio Management Center

This document provides information about how to install, configure, and maintain the HP Project and Portfolio Management Center (PPM Center) system. If you are not installing PPM Center for the first time, but need instructions on how to upgrade from an earlier version, see the *Upgrade Guide*.

The chapters in this document provide the following information about PPM Center and how to administer the system:

- Overview of PPM Center system architecture and of single-server and server cluster system configuration (Chapter 2, *System Overview*, on page 23)
- Information about product licensing and optional programs that you can install (Chapter 3, *Installing PPM Center*, on page 41)
- Instructions on how to create the required database schemas, verify installation, and install service packs and HP Deployment Management Extensions (Chapter 3, *Installing PPM Center*, on page 41)
- Details on how to configure all components of the PPM Center system and to start and stop the PPM Server (Chapter 4, *Configuring the System*, on page 93)
- Information that PPM Center users need to know in order to use the PPM Workbench (Chapter 4, *Configuring the System*, on page 93)

- Advanced configuration information, including details on how to configure an external Web server and PPM Server clusters (Chapter 5, *Advanced System Configuration*, on page 135)
- Information on how to integrate PPM Center with an LDAP server (Chapter 5, *Advanced System Configuration*, on page 135)
- Details on how to maintain the PPM Center and the database after installation and configuration (Chapter 7, *Maintaining the System*, on page 229)
- Information about the kinds of performance issues that can arise, and how to identify and resolve them (Chapter 8, *Improving System Performance*, on page 287)
- Information on how to migrate entire instances of PPM Center, and on how to migrate just the database schemas (Chapter 10, *Migrating Instances*, on page 371)
- Details on how to use the HP entity migrators to migrate specific kinds of PPM Center entities and associated objects between instances of PPM Center (Chapter 9, *Migrating Entities*, on page 327)
- PPM Server configuration parameters (Appendix A, *PPM Center Configuration Parameters*, on page 389)
- Details about PPM Center directories and the scripts and tools they contain (Appendix B, *Server Directory Structure and Server Tools*, on page 481)

This document is written for users who are moderately knowledgeable about enterprise application development and skilled in enterprise system and database administration. It is written for:

- Application developers and configurators
- System and instance administrators
- Database administrators (DBAs)

Related Documents

The following documents provide installation information for system administrators and DBAs:

- *System Requirements and Compatibility Matrix*

  Before you install PPM Center, check this document to make sure that your operating environment meets *all* of the minimum system requirements.

- *Release Notes*

  The *Release Notes* document provides product information that is not included in the regular documentation set.

- *Creating Portlets and Modules*

  Refer to this document for information on how to create and maintain your own PPM Dashboard pages, modules, and portlets for display in the standard interface.

- *Multilingual User Interface Guide and Reference*

  If your organization has offices in different countries, see this guide for information on how to set up the Multilingual User Interface (MLU) in PPM Center.

- *Upgrade Guide*

  If you plan to upgrade from an earlier version of PPM Center, see this guide for information on supported upgrade paths, what to do to prepare to upgrade, and how to perform and then verify the upgrade.

Additional documents that you might find useful as you configure or maintain PPM Center include:

- *HP-Supplied Entities Guide*
- *Deployment Best Practices Guide*
- *Commands, Tokens, and Validations Guide and Reference*
- *Open Interface Guide and Reference*
- *Data Model Guide*
- *Web Services Guide*
- *Getting Started*
- *Creating Portlets and Modules*
- *Security Model Guide and Reference*
- *HP-Supplied Entities Guide* (includes descriptions of all portlets, request types, and workflows in PPM Center)

Accessing PPM Center Documentation

To obtain all of the HP PPM Center documentation, go to the HP Software Product Manuals Web site (h20230.www2.hp.com/selfsolve/manuals). To access this Web site, you must first set up an HP Passport account.

Configuring the PPM Documentation Library

After you install PPM Center, you can configure the PPM Documentation Library so that your PPM Center users can access documentation from the product itself. For information on how to configure the PPM Documentation Library, see *Customizing the Standard Interface.*

2 System Overview

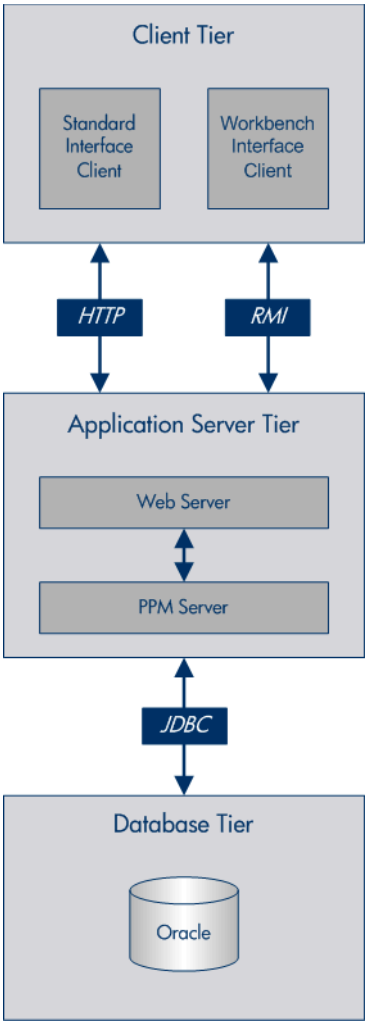
Overview of PPM Center Architecture

PPM Center is based on a three-tier architecture that consists of:

- Client browsers (*Client Tier*)
- One or more middle-tier J2EE servers (*Application Server Tier*)
- A single Oracle relational database (*Database Tier*)

This arrangement is shown in *Figure 2-1*.

Figure 2-1. PPM Center architecture



Browser clients use HTTP or HTTPS (HTTPS requires an external Web server) to communicate with the PPM Center Web and application servers. PPM Workbench clients (Java™ applet) use Remote Method Invocation (RMI). The following sections provide information about each tier.

Client Tier

The client tier of the system consists of:

- The PPM Center standard interface. The standard interface is rendered using Java Server Pages (JSP) and is accessed using a Web browser.
- The PPM Workbench interface is displayed using a Java applet installed on the client machine, and is started using the Sun Java plug-in to a Web browser.

The client and application server tiers communicate as follows:

- For the standard interface, the client and application server communicate using HTTP or HTTPS, with no code required on client machines. The client accesses information from the database through the J2EE application server using a shared database session pool.

  To use HTTPS, you must also use an external Web server.

- For the PPM Workbench interface, the client and application server communicate using Remote Method Invocation (RMI) or Secure Remote Method Invocation (SRMI), which is optimized for use in PPM Center.

  The architecture and communication protocols are created to minimize the number of round trips between the applet and server, and the volume of data transferred.

For more information about the PPM Center standard and PPM Workbench interfaces, see the *Getting Started* guide.

Application Server Tier

The application server (PPM Server):

- Runs on the Microsoft® Windows®, Sun Solaris, HP-UX, IBM AIX, Red Hat Linux, and SUSE Linux platforms
- Uses the JBoss Application Server
- Houses workflow, scheduling, notification, and execution engines that drive automated tasks such as code deployment to remote systems, running scheduled reports, and email notifications
- Can run on multiple machines as a cluster to improve performance and scale hardware as usage increases
- Can run with external Web servers such as Sun Java System Web Server, Microsoft IIS, Apache HTTP Server, Apache-based Web Server (from HP), and IBM HTTP Server

  For detailed information on which Web server versions PPM Center supports, and on which platforms, see the *System Requirements and Compatibility Matrix.*

- Maintains a database connection pool that caches connections to the database
- Eliminates the need to restart the PPM Server if the database shuts down for scheduled maintenance or because of system failure

The application server and the PPM Web server communicate using Apache JServ Protocol version 1.3, or AJP13. The AJP13 protocol is similar to HTTP that has been optimized for performance. The application server and database tiers communicate using Java Database Connectivity (JDBC).

For more information about configuring an external Web server, see *Configuring an External Web Server* on page 135.

Database Tier

The database tier consists of an Oracle database that contains the tables, procedures, PL/SQL packages, and other components that the PPM Center products use. All transaction, setup, and auditing data are stored in the database. PPM Center can run on a single database instance, or it can leverage Oracle RAC (Real Application Cluster) configuration for load balancing, redundancy, and failover.

The database consists of the following two database schemas:

- The central schema contains the core PPM Center data model and PL/SQL package code. The core data model contains all PPM Center configuration and transaction data.
- The Reporting Meta Layer (RML) schema contains a set of database views to facilitate reporting on PPM Center data.

PPM Center supports the following Oracle database features:

- A relational data model
- Use of Oracle stored procedures to implement business logic (for example, workflow processing)
- Use of a database connection pool to eliminate the need to create a separate database session for each user or transaction
- Database caching of frequently used data, programs, and procedures to improve performance
- A database schema (separate from the PPM Center database schema) for implementing operation reporting in PPM Center

For information about the BusinessObjects-based Operational Reporting solution for PPM Center, see the *Operational Reporting Administrator's Guide* and the *Operational Reporting User's Guide.*

System Configurations

The three-tier architecture of PPM Center supports a variety of system configurations. You can deploy PPM Servers in a single-server, stand-alone configuration or a server cluster configuration.

In a production environment, you *must* deploy a server cluster. A stand-alone configuration is adequate only in a development or testing environment. The following sections provide information about the possible ways you can configure your PPM Center instance.

Server Cluster Configurations (Recommended)

PPM Center uses JBoss clustering technology to enable you to configure *server clusters.* Clustering enables you to run PPM Center on several parallel servers called *nodes.* Server cluster configurations improve performance on systems that handle high transaction volumes or large numbers of concurrent users. In addition to handling higher user loads and providing greater scalability, the server cluster configuration supports load balancing and server failover features.

Because clustering distributes work load across different nodes, if any node fails, PPM Center is still accessible through other nodes in the cluster. You can continue to improve system performance by simply adding nodes to the cluster.

To leverage the clustering capabilities within PPM Center to support either background service isolation or the load-balanced user traffic across multiple nodes, you must configure the instance (collection of nodes) as a formal cluster.

To avoid problems with memory and performance, HP strongly recommends that you isolate background services from user traffic. For more information, see *Services Isolation* on page 35.

A server cluster configuration gives you more control over background services scheduling and execution, and enables you to designate specific nodes to run certain services. For example, you can run Service A on node 1 and run Service B on node 2. If node 1 fails, node 2 then runs Services A and B. As soon as node 1 is up and running again, new scheduled instances of Service A run on node1.

In a clustered environment with Java Message Service (JMS) and Quartz clustered scheduling, you can configure the following:

- Clustered nodes run across multiple machines
- JMS monitoring
- Number of consumers (listeners) per node for each queue
- Service failure rules per queue (that is, number of retries, log failures, server shut-downs)
- Notification messages to be sent to each queue
- Specific number of threads per node and cluster information for the scheduler
- Scheduler time zone. (This may be required if the database and PPM Servers are located in different geographies.)

To handle large numbers of concurrent users, server cluster configurations use either an external Web server or a hardware-based load balancer to distribute user connections evenly across multiple nodes. If more than one node in a cluster is dedicated to running services, and one of these services nodes shuts down, activities such as email notifications and executions scheduled to run on that node are automatically transferred to another available services node. This server failover feature helps ensure that PPM Center system services remain operational.

Any unsaved changes on a node that shuts down are lost and are not transferred to an available node. Users who log on to PPM Center after a node shuts down see only changes that were saved on that node.

Server cluster configurations contain two or more PPM Servers and an Oracle database. The first PPM Server installed and configured is the *primary node*. The other server (assuming a two-server setup) is the *secondary node*. The two servers can act as peers in a load-balancing situation, or one can act as a backup machine for the other.

A server cluster setup can include Oracle RAC. If a database in a setup such as this goes down, the Oracle JDBC driver manages database connectivity.

You can implement server cluster configurations on a single machine or on multiple machines. To run multiple PPM Servers on a single machine, the machine's memory capacity and CPU usage must meet the same memory and CPU requirements for multiple servers. To run multiple servers on multiple machines, the servers must share a common file system for reports, execution logs, attachment files, and server configuration files. Although each machine can contain its own instance of the PPM Center application code, only a single copy is required for each machine, regardless of the number of servers running on that machine.

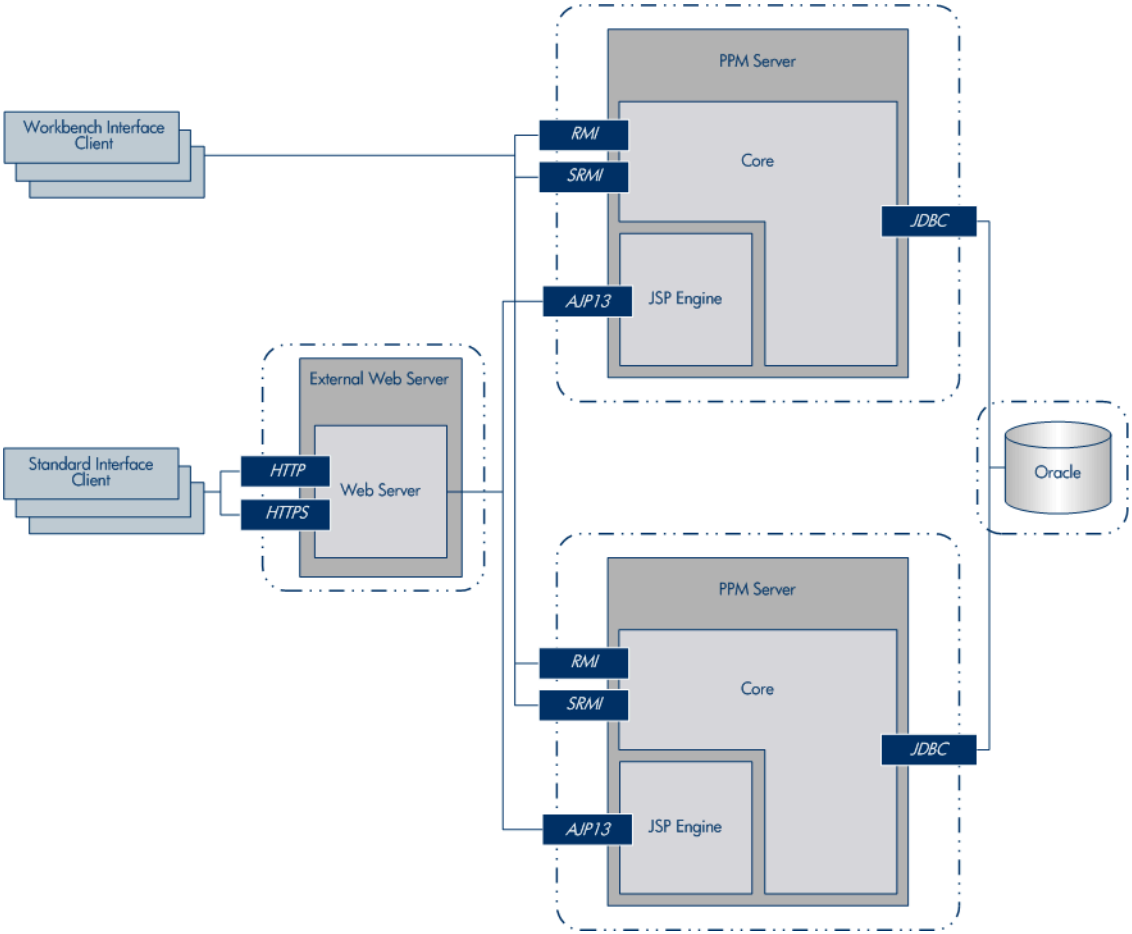
You can set up server clusters with an external Web server, or with a hardware load balancer. The following sections describe these two setups.

Server Cluster/External Web Server Configuration

The server cluster/external Web server configuration (see *Figure 2-2*) distributes client connections evenly among any number of nodes, based on Web traffic and server load. This configuration is typically used for organizations that need to load-balance Web traffic across multiple nodes (as an alternative to hardware-based load balancing). It can also be useful to an organization that already uses a standard Web server within its network infrastructure.

You can usually improve user load, transaction capacity, and system performance with this configuration. The extent of improvement depends on the number of nodes in the cluster and their available resources. This configuration supports load balancing and server failover features.

Figure 2-2. Server cluster/external Web server configuration



The Web server (internal or external) listens for HTTP or HTTPS requests from standard interface clients. Nodes run in the background and are transparent to users. Users access only the URL to the Web server.

The Web server forwards HTTP or HTTPS requests to one of the nodes. The PPM Web server and the nodes communicate using Apache JServ Protocol version 1.3 (AJP13).

The nodes also accept RMI or SRMI connections from PPM Workbench users who run applets in browsers to directly connect to the PPM Server using this protocol. The PPM Server uses JDBC to communicate with the Oracle database.

Because of Windows registry limitations, you cannot use just one Web server for multiple stand-alone PPM Center instances on a machine running Windows. If you must use an external Web server for multiple PPM Center instances, HP recommends that you either use a UNIX machine to host the Web server, or use a hardware load balancer.

Software Load Balancing

You can use the PPM Center Web server module as the software load balancer for a PPM Server cluster configuration. For this configuration, HP recommends that PPM Center nodes in the cluster *not* accept HTTP requests that are not routed through the Web server.

The request sequence is as follows:

1. A user submits an HTTP request to the Web server.
2. The Web server forwards the request to the HP PPM Web server module.
3. The HP PPM Web server module sends the request to a PPM Server.

Integrating with a Single Sign-On Product

For instructions on how to implement single sign-on with PPM Center, see Chapter 6, *Implementing User Authentication*, on page 203.

Using SSL Accelerators

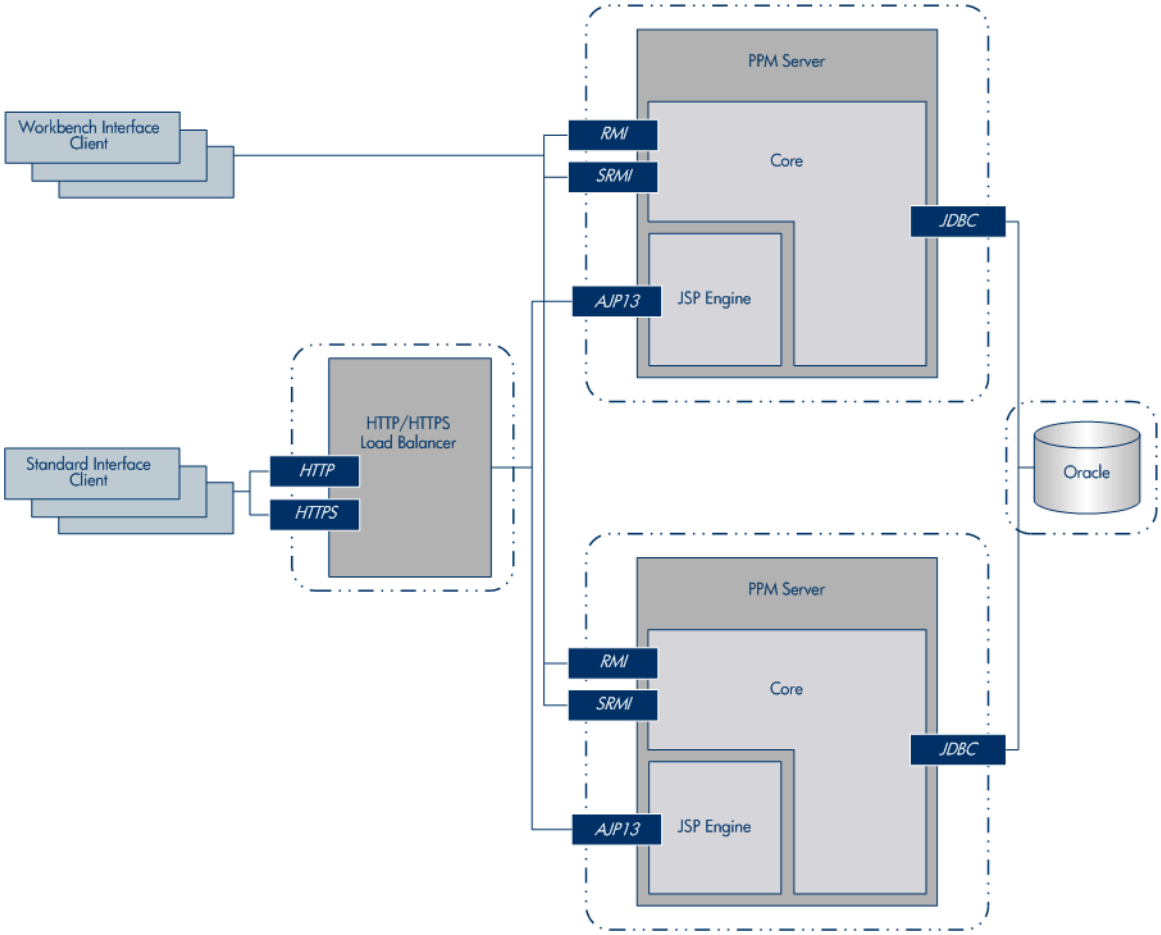
For PPM Server cluster configurations running HTTPS, you must integrate an external Web server that supports the appropriate accelerator to leverage a hardware-based SSL accelerator.

The external Web server and PPM Servers communicate using AJP13, a proprietary protocol that can be more efficient than HTTP for communicating with PPM Servers using an external Web server. For information about how to set up a server cluster with an external Web server, see Chapter 5, *Advanced System Configuration*, on page 135.

Server Cluster Hardware Load Balancer Configuration

The server cluster/hardware load balancer configuration (shown in *Figure 2-3*) is similar to the server cluster/external Web server configuration. However, in place of an external Web server, a hardware load balancer is used to balance client HTTP sessions across nodes. This configuration ensures the even distribution of client connections among nodes based on server load and availability.

Figure 2-3. Server cluster/hardware load balancer configuration



In the server cluster/hardware load balancer configuration (shown in *Figure 2-3*):

- Standard interface clients communicate with nodes using HTTP (or, for secure communication, HTTPS) through the use of a hardware load balancer. The hardware load balancer listens for the HTTP or HTTPS requests that it distributes.

  To use HTTPS, you must also use an external Web server.

  Many hardware load balancers support handling HTTPS and forwarding plain HTTP. In this case, the hardware load balancer handles the encryption and decryption of requests, and the nodes perform other tasks. Setting up your system this way can improve its performance.

- PPM Workbench interface clients communicate directly with the PPM Server using RMI or, for secure communication, SRMI.

- The PPM Server and Oracle database reside on separate machines and communicate with each other using JDBC.

  Although *Figure 2-3* on page 33 shows multiple nodes and just a single database, the system can support Oracle RAC or a single database mirrored for redundancy across multiple machines.

Using this configuration improves user load distribution, transaction capacity, and system performance. The degree of improvement depends on the number of nodes in the cluster and the resources available to each. Load balancing and server failover features are supported in this configuration.

For information about how to set up a server cluster/hardware load balancer configuration, see Chapter 5, *Advanced System Configuration*, on page 135.

Services Isolation

PPM Center has many asynchronous background services that process data "behind the scenes" while the application is running. Depending on data characteristics of your PPM Center deployment, the overhead of these services in terms of CPU and memory demand are difficult to estimate. To reduce the impact of services on user response times, HP strongly recommends that you isolate the services on a separate JVM within the PPM Center server cluster.

Services isolation does not require isolation of services onto separate physical servers. A node that you dedicate to services can reside on the same machine that hosts nodes handling user traffic. Even in a shared host model, there is benefit if higher performance-risk services, which tend to be CPU-bound on the application tier, have a separate node.

PPM Center server clustering does not differentiate between primary or backup nodes in terms of configuration. The first node to start up attempts to be the "service primary." If a node that is considered to be a "backup" starts first, then it is the primary. The objective is to earmark a subset of nodes in the server cluster as services-capable. All of the nodes are peers, and "ownership" of services is based simply on startup order.

If a node that is running services fails, one of the other nodes enabled to run services assumes the role of primary. If the node that failed is restarted, services will not automatically "failback" to that node. To return services ownership to the node that failed and is restarted, you must stop, and then restart the node that took over services execution from the original services node.

HP recommends that you devote at least one PPM Center node to process PPM Center background services.

Dedicating one PPM Center node to your services enables you to:

- Minimize the effect that running PPM Center services has on users
- Better monitor the performance of the services

The more you monitor and understand how your services affect performance, the better you can tune them.

Single-Server Configurations

PPM Center test and development instances are typically single-server configurations that consist of one PPM Server and one Oracle database. The single PPM Server handles the entire user load and functions as the Web server. It also houses the file system for the program code, reports, execution logs, and attachments files. The Oracle database stores all other data.

You can set up the following types of single-server configurations:

- Single-server/single-machine configuration
- Single-server/multiple-machine configuration
- Single-server/external Web server configuration

When you install PPM Center (see Chapter 3, *Installing PPM Center*, on page 43), you specify whether you want a server cluster configuration (required for a production instance) or a *stand-alone* configuration. A stand-alone PPM Center deployment always includes just one PPM Server installation with a single-node configuration.

You can configure a server cluster that has just one node. The difference is that you can add nodes to the “server cluster” while you cannot add nodes to a stand-alone instance.

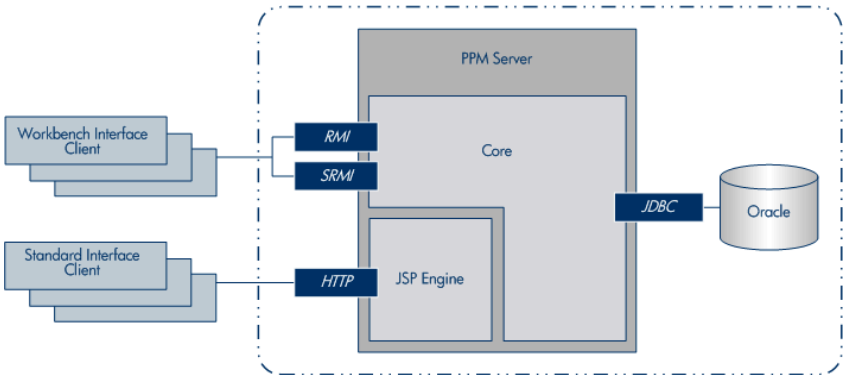
In a production environment, you must deploy a server cluster. A stand-alone configuration is adequate only in a development or testing environment.

Single-Server/Single-Machine Configuration for a Development or Testing Environment

The single-server/single-machine configuration shown in *Figure 2-4* consists of one machine that hosts both the PPM Server and the Oracle database.

HP strongly recommends that you use the single-server/single-machine configuration only for a stand-alone PPM Center deployment in either a testing or development environment, and never for a production instance.

Figure 2-4. Single-server/single-machine configuration



Standard interface clients communicate with the PPM Server using HTTP, or, for secure communication, HTTPS (requires that you use an external Web server). PPM Workbench interface clients communicate with the PPM Server using RMI, or, for secure communication, SRMI.

The machine that houses the PPM Server also contains the Oracle database. The PPM Server uses JDBC to communicate with the Oracle database.

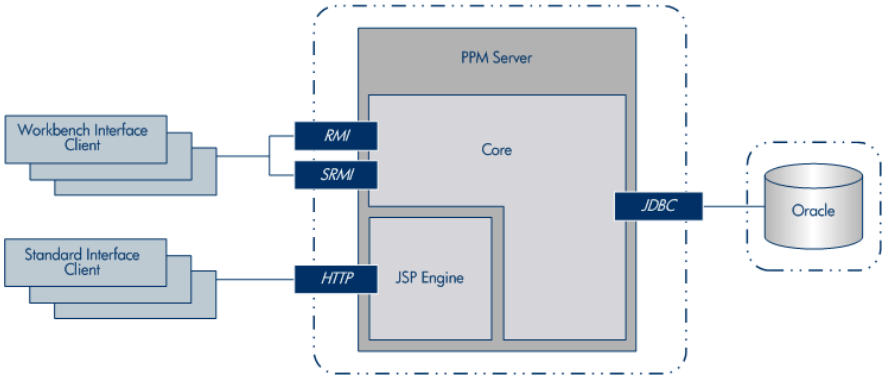
An organization typically uses this configuration if it requires a dedicated machine for all PPM Center services and database operations. User load, transaction capacity, and system performance depend on the available resources on a machine. This configuration does not support load balancing or server failover features.

For information about how to set up a single-server/single-machine configuration, see Chapter 3, *Installing PPM Center*, on page 41.

Single-Server/Multiple-Machine Configuration

In the single-server/multiple-machine configuration shown in *Figure 2-5*, the PPM Server and the Oracle database reside on separate machines. This configuration offers additional performance capacity and modularizes the maintenance of the application server and database tiers. The separate machines can run on different operating systems, thereby enabling greater flexibility.

Figure 2-5. Configuration with a single server and multiple machines



Standard interface clients communicate with the PPM Server using HTTP, or HTTPS for secure communication. (To use HTTPS, you must use an external Web server.) PPM Workbench interface clients communicate with the PPM Server using RMI, or SRMI for secure communication. The PPM Server and Oracle database use JDBC to communicate.

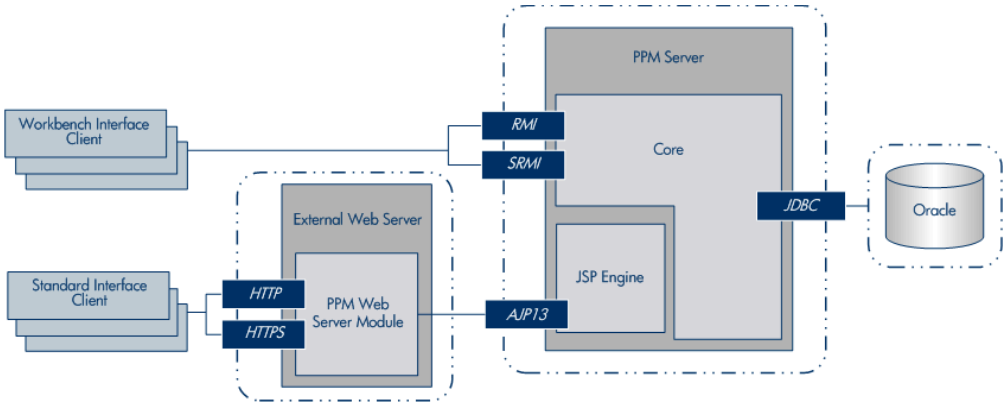
An organization typically uses the single-server/multiple-machine configuration if it requires a separate machine for database operations. User load, transaction capacity, and system performance depend on the resources available on the PPM Server machine. This configuration does not support load balancing or server failover features.

For information about how to set up a single-server/multiple-machine configuration, see Chapter 3, *Installing PPM Center*, on page 41.

Single-Server/External Web Server Configuration

In the single-server/external Web server configuration shown in *Figure 2-6*, Web traffic comes into the Web server and is then passed to PPM Center. The external Web server and the PPM Server communicate using AJP13, a proprietary protocol that is more efficient for this configuration type than HTTP or HTTPS.

Figure 2-6. Single-server/external Web server configuration



- Standard interface clients communicate with an external Web server using HTTP, or, for secure communication, HTTPS. The external Web server and PPM Servers use AJP13 to communicate.
- PPM Workbench interface clients communicate directly with the PPM Server using RMI, or, for secure communication, SRMI.
- The PPM Server and Oracle database server reside on separate machines. The PPM Server communicates with the Oracle database using JDBC.

This configuration is suitable if your organization:

- Already uses a standard Web server within the network infrastructure.
- Must prevent clients from having direct access to the PPM Server.

IT departments often have standards for the Web server used for HTTP traffic. Running the HTTP listener allows for PPM Center integration with enterprise-specific architecture.

System administrators typically prefer HTTP traffic configured on port 80. On UNIX® systems, processes must run as root to listen on a port number lower than 1024. However, HP recommends that you not run the PPM Server as root. If you want to configure HTTP traffic on a port number less than 1024, consider integrating with an external Web server.

As with other single-server configurations, user load, transaction capacity, and system performance depend on available resources on the PPM Server machine. This configuration does not support load balancing and server failover features.

HP recommends that you use the internal Web server built into the PPM Server unless you have the kind of special Web server requirements described in this section.

For information about how to set up a single-server/external Web server configuration, see Chapter 3, *Installing PPM Center*, on page 41 and Chapter 5, *Advanced System Configuration*, on page 135.

For a list of supported external Web servers, see the *System Requirements and Compatibility Matrix*. For information on how to access this and other PPM Center documents, see *Accessing PPM Center Documentation* on page 22.

3 Installing PPM Center

Key Considerations

To prepare to install PPM Center, review this section.

For information about how to upgrade to PPM Center version 9.10 from an earlier version, see the *Upgrade Guide.*

Installing for the First Time

To prepare to install PPM Center version 9.10:

1. Make sure that your organization has obtained the installation software.
2. Read the rest of this chapter.
3. Read the *System Requirements and Compatibility Matrix,* which is described in *Related Documents* on page 21.
4. Read the *Release Notes,* which are described in *Related Documents* on page 21.
5. If you plan to install the PPM Center multilingual user interface, see the *Multilingual User Interface Guide.*
6. To make sure that you have performed all required preinstallation tasks.

7. If you plan to install one of the HP Deployment Management Extensions, see the documentation for the product.

   For information on how to access documentation for HP Migrators and HP Deployment Management Extensions, see *Accessing PPM Center Documentation* on page 22.

8. Make sure that you have the valid licenses required for all of the products you plan to install. For information about how to determine which product licenses your organization has, see *Checking PPM Center License Status* on page 286.

9. Install PPM Center. For instructions on how to install PPM Center, see *Installing PPM Center* on page 74.

10. Configure the PPM Server and system environment.

    For information about how to configure PPM Center, see Chapter 4, *Configuring the System*, on page 93 and Chapter 5, *Advanced System Configuration*, on page 135.

11. Install and configure optional products you have purchased to work with PPM Center.

    After you install PPM Center, you can install Extensions, or the HP GL Migrator in any order you choose. For information about how to install and configure optional products, see *Optional Installations* on page 90.

Installing PPM Center Application Lifecycle Management Software

Business Technology Optimization software for PPM Center includes the Application Lifecycle Management (ALM) software. PPM Center ALM software provides entities such as request types, workflows, portlets, reports, and special commands that you can install in PPM Center to support Information Technology Infrastructure Library (ITIL) processes for change management and release management.

ALM enhances PPM Center functionality, and facilitates PPM Center integration with HP Service Manager, HP Universal Configuration Management Database (Universal CMDB), HP Quality Center, and HP Release Control. For detailed information about ALM software and how to install it, see the *HP Solution Integrations Guide.*

Plug-in for PPM

If you plan to integrate Microsoft Project with HP Project Management by installing the Plug-in for PPM, be aware of the following:

- The Plug-in for PPM is a VSTO .Net Office add-in that adds a menu to Microsoft Project. You can use the menu to synchronize Microsoft Project and HP Project Management data. All communication with the PPM Server is done using HTTP requests to the PPM Server, and the data is exchanged in XML format.
- Microsoft Project functionality is affected only while the currently opened project is integrated with PPM Center (for example, while loading tasks, filling in actual effort, and so on).

For information about the requirements and instructions for installing the Plug-in for PPM, see the *HP Project Management User's Guide.*

Installing BusinessObjects Enterprise for PPM Center Operational Reporting

The optional Operational Reporting solution for PPM Center enables you to provide rich, interactive reports on PPM Center data. You can use it to distribute weekly reports, provide customers with personalized service offerings, or to integrate business-critical information into corporate portals.

If you plan to use Operational Reporting, you must perform a separate installation. For more information, see the *Operational Reporting Administrator's Guide.*

Installing Document Management

The optional HP document management module in PPM Center gives you more control over document search and storage than you otherwise have in PPM Center. You can use the document management module to track, index, and search multiple versions of supporting documents attached to PPM Center entities in HP Demand Management, HP Program Management, and HP Project Management.

If you plan to use the document management module, you must perform a separate installation. For more information, see the *Document Management Guide and Reference.*

Installing HP Object Migrator or HP GL Migrator

If you are running PPM Center in the Oracle environment, and plan to use HP Object Migrator or HP GL Migrator software, you must consult not only the installation instructions in this document, but also the instructions in the HP Object Migrator or HP GL Migrator documentation.

For information about the HP Object Migrator and the HP GL Migrator, see the *HP Object Migrator Guide* and the *HP GL Migrator Guide*, respectively.

Installing an HP Deployment Management Extension

If you purchased an HP Deployment Management Extension, be sure to consult not only the installation instructions in this document, but also the instructions in the HP Deployment Management Extensions documentation.

To complete an Extension installation successfully, you must make sure that you have the required system privileges. For information about these privileges, and how to grant them, see *Key Decisions* on page 46.

Obtaining License Keys

Make sure that you have purchased the HP products you intend to install (you can purchase and install additional products later), and that you have obtained the required license file for the purchased version. PPM Center license keys are delivered in the `license.conf` file, which you can find in the `<PPM_Home>/conf` directory after installation.

Checking System Requirements

Before you start to install PPM Center, make sure that your system environment meets all the requirements. For information about the system requirements, see the *System Requirements and Compatibility Matrix*. For information on how to access this and other PPM Center documents, see *Accessing PPM Center Documentation* on page 22.

Key Decisions

This section addresses several decisions you must make before you begin to install your PPM Center products.

Table 3-1. Decisions to make before you install (page 1 of 2)

Decision	What to Consider
When do I configure the PPM Server?	Before you can start the PPM Server, you must configure it. The installer prompts you for several server parameter values. If you do configure during installation, the installer saves the values you provide to the server configuration file, and you can complete server configuration after installation, without having to reenter the values. If the server information you provide (for example, valid port numbers) is unavailable during installation, you must configure the PPM Server after you install it. For instructions, see *Configuring or Reconfiguring the PPM Server* on page 99.
When do I create the database schemas?	The PPM Server requires two database schemas to store application data. You can create these schemas before you install PPM Center, or you can create them automatically during installation. To create the schemas before installation, follow the instructions provided in *Creating the Database Schemas* on page 66. If you set up the schemas before installation, the installer populates them with the entities and data required to run the PPM Server.
When do I set up grants to the database schema?	To improve PPM Center performance, the installer rebuilds statistics for the Oracle optimizer during installation. You cannot successfully complete the installation until you grant privileges and rebuild the statistics.

Table 3-1. Decisions to make before you install (page 2 of 2)

Decision	What to Consider
What privileges do I grant the database schema user?	To rebuild the statistics, the PPM Center database schema user must be granted the following privileges (as SYS DBA on Oracle): • `grant select on v_$parameter to <PPM_Schema>` • `grant select on v_$mystat to <PPM_Schema>` • `grant select on v_$process to <PPM_Schema>` • `grant select on v_$session to <PPM_Schema>` • `grant execute on dbms_stats to <PPM_Schema>` If you have access to SQL*Plus, you can run the script `sys/GrantSysPrivs.sql` (located in the `<PPM_Extract>/ppm910/sys` directory), which grants all required privileges for you. You can run the script before installation (as SYS DBA) or during installation. If you are logged on as SYS DBA, you can run the script after installation. In this case, the installer does not gather statistics or install Best Practices content. For information about installing Best Practices content, see *About PPM Center Best Practices Installation*.
Do I run the install program in graphic mode or in console mode?	You can either install the PPM Server in graphic mode or in console mode.

About PPM Center Best Practices Installation

In addition to installing the foundation product, the database, and the application server (PPM Server), you can install Best Practices on your system. Best Practices provides you with experience-derived information and advice on how to configure and use HP Portfolio Management and HP Program Management.

Best Practices Content

HP Project Management and HP Portfolio Management access the request type content installed as part of Best Practices. This includes HP-supplied menu items that access these request types. *Table 3-2* lists the Best Practices request types and their associated workflows, and provides descriptions of the product functionality they provide. *Table 3-3* lists the functionality that Best Practices workflows content enabled in this version of PPM Center.

HP Demand Management and HP Portfolio Management Best Practices content also includes scripts that populate the Default Demand Set and Default Scoring criteria information.

Requirements for Installing Best Practices

You can install Best Practices if *all* of the following conditions are met:

- PPM Center database schema name
- PPM Center database username and password
- PPM Center administrator username and password
- You have licenses for both the HP Portfolio Management and HP Program Management.

For detailed instructions on how to install Best Practices separately, see *Installing HP Project and Portfolio Management Best Practices* on page 90.

The Best Practices content supplied with this version of PPM Center includes the request types and workflows listed in *Table 3-2* and *Table 3-3*.

In order for PPM Center software to function correctly, Best Practices request types must be installed on your system and correctly associated with the menu items and project types.

Table 3-2. Best Practices request types (page 1 of 3)

Request Type and Description	Menu Selections	Field Groups
PFM - Proposal Represents a project proposal within the HP Portfolio Management module.	• Create > Proposal • Open > Portfolio Management > Create Proposal	PFM Proposal
PFM - Asset Represents the ongoing costs and maintenance of the result of a project in the HP Portfolio Management system.	• Create > Asset • Open > Portfolio Management > Create Asset	PFM Asset
PFM - Project Represents data and process associated with project. This request type is connected to a project type.	• Create > Project • Open > Portfolio Management > Create Project	PFM Project
Project Issue Represents issues associated with a project.	• Create > Project Issue • Open > Project Management > Project Controls > Submit Project Issue • Search > Project Issues • Open > Project Management > Project Controls > Search Project Issues Also associated with a project type to enable creating from within the Project Overview page.	• Project Reference • Project Issue

Table 3-2. Best Practices request types (page 2 of 3)

Request Type and Description	Menu Selections	Field Groups
Project Risk Represents risks associated with a project.	• Create > Project Risk • Open > Project Management > Project Controls > Submit Risk • Search > Project Risks • Open > Project Management > Project Controls > Search Project Risks Also associated with a project type to enable creating from within the Project Overview page.	• Project Reference • Project Risk
Project Scope Change Request Represents scope changes associated with a project.	• Create > Project Scope Change • Open > Project Management > Project Controls > Submit Scope Change • Search > Project Scope Changes • Open > Project Management > Project Controls > Search Scope Changes Also associated with a project type to enable creating from within the Project Overview page.	• Project Reference • Project Scope Change
Program Issue Represents issues associated with a program.	• Create > Program Issue • Open > Program Management > Issues > Submit Program Issues • Search > Program Issues • Open > Program Management > Issues > Search Program Issues	• Program Issue • Program Reference

Table 3-2. Best Practices request types (page 3 of 3)

Request Type and Description	Menu Selections	Field Groups
Program Risk Represents risks associated with a program	• Search > Program Risks • Create > Program Risk • Open > Program Management > Issues > Search Program Risks • Open > Program Management > Issues > Submit Program Risk	• Program Reference • Program Risk
DEM - Application Enhancement Used to request new functionality in IT current applications.	• Create > Request • Open > Demand Management > Create Request • Search > Requests • Open > Demand Management > Search Requests	• Demand Management SLA Fields • Demand Scheduling Fields
DEM - Database Refresh Database refresh requests can be made for all IT Ops applications in the testing phase.	• Create > Request • Open > Demand Management > Create Request	• Demand Management SLA Fields • Demand Scheduling Fields
DEM - Application Bug Used to report problems in IT applications.	• Create > Request • Open > Demand Management > Create Request	• Demand Management SLA Fields • Demand Scheduling Fields
DEM - Initiative Used to request key projects for future quarters.	• Create > Request • Open > Demand Management > Create Request	• Demand Management SLA Fields • Demand Scheduling Fields

You can create your own versions of the Best Practices request types by adding the appropriate field group, and then either editing the menu XML files or associating the request type with the project type (for Project Issue, Project Risk, and Project Scope Change).

Table 3-3 lists the functionality that Best Practices workflows content enabled in this version of PPM Center.

Table 3-3. Best Practices workflows

Best Practices Workflow	Description
PFM - Proposal	HP Portfolio Management process for requesting a new project.
PFM - Asset	HP Portfolio Management process for an asset life cycle.
PFM - Project	HP Portfolio Management process for a project life cycle.
DEM - Enhancement Request Process	HP Demand Management process for enhancement requests for new functionality in applications.
DEM - Database Refresh	HP Demand Management process for database refresh requests.
DEM - Bug Request Workflow	HP Demand Management process for application bug requests.
DEM - Project Initiative Process	HP Demand Management process for initiative requests for key projects.
Program Risk Management Process	Automated process for program risk management.
Risk Management Process	Automated process for project risk management.
Scope Change Request Process	Automated scope change request process with three levels of severity.
Issue Management Process	Automated process for issue management.

Language Support for Best Practices

PPM Center Best Practices content is available for all supported language packs. HP recommends that you deploy language packs soon after you install Best Practices so that the its content matches in all languages.

For detailed information about the language packs and how to install them, see the *Multilingual User Interface Guide.*

Preparing to Install PPM Center

Before you start to install PPM Center, complete the following tasks:

1. Check the *System Requirements and Compatibility Matrix* to make sure that your system meets *all* of the minimum requirements.

2. Make sure that you have at least 300 MB temporary space and 0.5 to 1 GB swap space on your operating system.

   Limits on physical memory for 32-bit Windows operating systems depend in part on whether the Physical Address Extension (PAE) is enabled. The PAE allows some 32-bit Windows systems (Windows Server 2008 Datacenter and Windows Server 2008 Enterprise) to use more than 4 GB of physical memory. (To enable PAE, use the /PAE switch in the `Boot.ini` file.)

   Because PAE is not available for Windows Server 2003 Standard Edition, the total available physical RAM on this system is limited to 4 GB. For detailed information about memory support and memory limitations on Windows operating systems, see Microsoft Support online.

3. Set several Oracle database parameters to the values recommended for the system environment and optimum system performance. For details, see *Configuring or Reconfiguring the Database* on page 119.

4. Enable the Oracle Java Virtual Machine (OracleJVM).

   PPM Center uses Java Stored Procedures in Oracle. Java Stored Procedures enable you to call Java code from PL/SQL. To use this feature, you must enable the Oracle Java virtual machine (OracleJVM). For information about how to install and configure the Oracle Java virtual machine (JVM),

see the *Oracle® Database Java Developer's Guide* for your Oracle software version.

HP strongly recommends that you automate memory so that the size of the JAVA pool (Oracle `JAVA_POOL_SIZE` parameter setting) is allocated automatically.To automate memory allocation for Oracle 11g databases, use Automatic Memory Management (AMM).

5. Collect the information required for installation.

   For information about what information is required, see *Collecting Required Information* on page 57.

6. Obtain the installation files, and save them to a temporary directory (`<PPM_Extract>`).

   The placeholder `<PPM_Extract>` represents the root directory to which you save the installation files. The name and location of this directory are up to you.

7. Install the JDK.

   For information on which version of the JDK to install, see the *System Requirements and Compatibility Matrix*. For information on how to install the JDK, see *Installing the Java Development Kit (JDK)* on page 62.

8. Verify that the `JAVA_HOME` environment variable is set.

9. (Windows systems only) On each Windows server with which PPM Center is to interact, download and install Cygwin. (For information about this UNIX emulator and how to install it, go to cygwin.com.)

10. (Windows systems only) Make sure that the `PATH` and `CLASSPATH` environment variables are set and that the directory paths contain no spaces.

11. Install Oracle client on the PPM Server.

12. Verify that the `ORACLE_HOME` environment variable is set.

13. Create a system account for PPM Center.

    To install PPM Center and maintain the system after installation, you must create a system account. After you do, always log on to the server machine

as this user to perform any PPM Server maintenance—for example, stopping and restarting the PPM Server. This helps to avoid file system permission issues, which can be difficult to track.

For instructions on how to create a system account, see *Creating a System Account for PPM Center* on page 65.

14. Set up the Oracle tablespaces required to create the schemas and database objects.

    To create schemas and database objects, you must first create the data, index, and character large object data type (CLOB) tablespaces.

15. Set the Oracle database parameter `NLS_LENGTH_SEMANTICS` value to `CHAR`.

    For information about setting `NLS_LENGTH_SEMANTICS` and other database parameters, see *Configuring or Reconfiguring the Database* on page 119.

16. Verify that the required ports are open through the firewall and that other applications are not using them. (See *Verifying Port Availability* on page 72.)

The following sections provide detailed information about each of these tasks.

The placeholder `<PPM_Home>`, which is used throughout this document, refers to the root directory where PPM Center is installed. The name and location of this directory are up to you.

Do not unzip the installation files in your `<PPM_Home>` directory—instead, choose a temporary directory in another location. The directory to which you extract the installation files is referred to in this document as `<PPM_Extract>`.

Obtaining PPM Center Software

After your organization's PPM Center software purchase order is processed, you receive a welcome letter email from HP, which lists the language-specific download links and provides instructions on how to activate your software license, access support, and how to log an incident. This section describes how to download the PPM Center software after you receive the HP welcome letter.

To download the PPM Center software:

1. Go to the Electronic download page on the HP order fulfillment Web site (https://h20348.www2.hp.com/ecommerce/efulfillment/downloadpage.do).
2. If you do not yet have an HP Passport, click **Register,** and then complete the registration process to obtain an HP Passport.
3. Once you have an HP Passport, click **Login.**
4. In the **UserID** box, type your HP Passport user ID.
5. In the **Password** box, type your HP Passport password.
6. Review, and then accept, the terms and conditions of the download.
7. Click **Sign-in.**

   The purchased software bundle download links become accessible for download (or redownload) for up to 90 days after the purchase date.
8. Click the link for the PPM Center software version you plan to install, and follow the instructions to complete the download.

Software Update Manager

If you have the required service agreement ID (SAID), you can get PPM Center software updates through the Software Update Manager (SUM) site (`http://www1.itrc.hp.com/service/sum/home.do`).

Collecting Required Information

The PPM Center installer prompts you to enter information that it uses to create and configure the PPM Server. The installer validates each value you enter before it continues the installation. *Table 3-4* lists the information required for installing PPM Center on either a single PPM Server or the primary node for a server cluster.

For additional information that you must provide if you are installing the primary PPM Server for a server cluster, see *Configuring a Server Cluster* on page 172.

Table 3-4. Required installation information (page 1 of 4)

Prompt	Description
CLASSPATH	Environment variable that specifies the directory in which Java class files reside. **Note:** The directory path must not contain spaces.
Software installation location	Directory in which the PPM Server is to be installed and configured. If the directory does not exist, the installer creates it. The directory path cannot contain spaces. **NOTE:** Do not map the <*PPM_Home*> directory so that it is accessible from an external Web server. This introduces a potential security risk. HP recommends that you not share this directory.
Path to the `license.conf` file	The `license.conf` file contains valid PPM Center license keys. The PPM Server is enabled by license keys provided in a `license.conf` file, which you must obtain before installation. If you do not have a valid `license.conf` file, contact HP Software Support Web site (hp.com/go/hpsoftwaresupport).
Path to JAVA_HOME	The directory in which Java is installed. On UNIX systems, this environment variable is set in the profile file (a `*.profile` or `*.cshrc` file) of the user who is installing PPM Center. Windows example `C:\j2sdk1.6` **Note:** Make sure that the value specified for JAVA_HOME contains no spaces.

Table 3-4. Required installation information (page 2 of 4)

Prompt	Description
Database Access page (Displayed if you chose to have the installer create the database schemas)	
System access username	System database user name to give the installer access to the database.
System password	System database password to give the installer access to the database.

JDBC URL	JDBC URL the PPM Server uses to connect the Oracle database.
	Short format (non-RAC):
	`jdbc:oracle:thin:@<Host_Name>:<Port>:<SID>`
	
	where
	• *<Host_Name>* is the host name or IP address of the computer running the database
	• *<Port>* is the port number that SQL*Net uses to connect to the database. To get the actual value, look at the corresponding entry in `tnsnames.ora`
	• *<SID>* is the security identifier of the database. This is usually identical to the database connect string. If it is different, an extra parameter is required.
	RAC format:
	`RAC (description=(address_list=(address=(protocol=TCP)(host=<Host_Name1>)(port=<Port>)) (address=(protocol=TCP)(host=<Host_Name2>) (port=<Port>))(load_balance=YES))(connect_data=(server=DEDICATED)(service_name=<Service_Name>))`
	Example of database access information used to enable the PPM Server to communicate with databases on two servers named Jaguar1 and Jaguar2:
	`jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP) (HOST=jaguar1)(PORT=1521)) (ADDRESS=(PROTOCOL=TCP)(HOST=jaguar2)(PORT=1521))) (CONNECT_DATA=(SERVICE_NAME=J920)))`

ORACLE_HOME	Home directory for the Oracle client tools on the PPM Server machine. The directory path cannot contain spaces.

Table 3-4. Required installation information (page 3 of 4)

Prompt	Description
PATH	Environment variable that specifies the directories to be searched to find a command.
SQL*PLUS	Location of the SQL*Plus utility. SQL*Plus is not required for installation, but is required for the PPM Server. Example `C:\Oracle\bin\sqlplus.exe` If the `ORACLE_HOME` environment variable is set, then this parameter is detected automatically.
PPM Center Schema page (Displayed if you created the database schemas before installation)	
Username	PPM Center database schema user name.
Password	PPM Center database schema password.
RML Schema page	
Username:	Username for the PPM Center Reporting Meta Layer (RML) schema.
Password	Password for the PPM Center Reporting Meta Layer (RML) schema.
Tablespaces page	
Table	Data type tablespace in the Oracle database
Index	Index type tablespace in the Oracle database
CLOB data	Character large object data type (CLOB) tablespace in the Oracle database
NT Service	
Service name	Name of the Windows service for the PPM Server.

Table 3-4. Required installation information (page 4 of 4)

Prompt	Description
Regional Settings	
Holiday schedule	Holiday schedule on which to base the PPM Center regional calendar. If your holiday schedule is not listed, you can select **None.** In that case, a new calendar with no holidays is set as the system default regional calendar.
Currency code	Three-letter code for the default currency. The system default is in US dollars (USD). For information on currency codes for other countries, see the *HP Financial Management User's Guide.* **Warning:** Once you choose your default currency during installation, you cannot change it.
Region name	Name of the region for the installation, which is defined by a combination of calendar and currency. If your organization operates in only one region, use "Enterprise" or your company name.

Multiple Stand-Alone PPM Center Instances on a Single Machine

If you have multiple stand-alone PPM Center instances installed on a single machine for testing or development purposes, you must make sure that each of these instances has a unique value for the parameters listed in *Table 3-5*.

A server cluster involves many more required server configuration parameter settings. For information about the required server configuration parameter settings for a server cluster, see *Configuring a Server Cluster* on page 172.

Table 3-5. Parameter settings required for multiple stand-alone instances installed on the same machine

Parameter	Description
APP_SERVER_NAMING_SERVICE_RMI_PORT	JBoss Naming Service RMI port. For a stand-alone PPM Server, specify a port that is unique for the instance. For a PPM Server in a server cluster, specify a port that is unique for the node in the cluster.
APP_SERVER_NAMING_SERVICE_BINDING_PORT	JBoss Naming Service Binding port. For a stand-alone PPM Server, specify a port that is unique for the instance. For a PPM Server in a server cluster, specify a port that is unique for the node in the cluster.
APP_SERVER_JRMP_INVOKER_RMI_PORT	JBoss JRMP Invoker RMI port to which RMI clients connect when communicating through the proxy interface. For a stand-alone PPM Server, specify a port that is unique for the instance. For a PPM Server in a server cluster, specify a port that is unique for the node in the cluster.
APP_SERVER_UIL2_BINDING_PORT	Protocol listening port for the JBoss Application Server UIL2 service. For a PPM Server in a server cluster, specify a port that is unique for the node in the cluster.

Unzipping the Installation Files

Before you run the installation driver script, extract your installation files for the PPM Center software to the `<PPM_Extract>` directory. The extraction procedure creates a new subdirectory named `ppm910`. Run the extraction command in a directory other than the `<PPM_Home>` directory.

Do not change the `ppm910` directory name.

Installing the Java Development Kit (JDK)

Because the PPM Server is based on Java, the machine that hosts it must also host a Java Virtual Machine (JVM), which is part of the Java Development Kit (JDK). JDKs native to the operating systems supported by PPM Center are available from either Oracle or from the operating system vendor.

You must install the complete JDK. The Java Runtime Environment (JRE) alone is not supported.

For a list of required JDKs, see the *System Requirements and Compatibility Matrix*. For information on how to access this and other PPM Center documents, see *Accessing PPM Center Documentation* on page 22.

To install the JDK:

1. Download the JDK for your operating system from Oracle or from your operating system vendor's Web site.
2. Install the JDK according to the instructions provided by the vendor.

   Many operating systems require that you apply operating system-specific patches before you install the JDK. Make sure that you follow all instructions that the vendor provides.

Some vendors provide custom installation packages that you can install automatically using a command such as `pkgadd`. Other vendors provide a TAR file that you must extract.

The directory path name must not contain spaces.

3. Verify that your user name has the Java executable in its path by logging on and running the following the command:

   ```
   java -version
   ```

   This returns the Java version. If an error message is displayed, modify the path environment variable, as required.

   For information on supported JDK software, see the *System Requirements and Compatibility Matrix.*

4. Verify that the `JAVA_HOME` environment variable is set correctly. If the path set for `JAVA_HOME` is not correct, set it to the correct value.

   For information about how to check for and set the `JAVA_HOME` environment variable, see *Verifying that the JAVA_HOME Environment Variable Is Set*.

Verifying that the JAVA_HOME Environment Variable Is Set

PPM Center requires that you set JAVA_HOME in the system environment of the user account to be used to start the PPM Server. It is important that the JAVA_HOME environment variable be set for the same shell and user who runs the installation.

Make sure that the value specified for JAVA_HOME contains no spaces.

Determining the JAVA_HOME Value in DOS

To determine the JAVA_HOME value in DOS, at the command line, type **`echo %JAVA_HOME%`**.

Determining the JAVA_HOME Value in UNIX

To determine the JAVA_HOME value in a UNIX shell (SH, BASH, or KSH), at the UNIX prompt, type **`echo $JAVA_HOME`**.

Setting the JAVA_HOME Value in Windows

The steps described in the following procedure are for Windows XP. The exact steps may differ, depending on your Windows operating system.

To set the value of JAVA_HOME in Windows:

1. Open the Control Panel.
2. Open the System Properties window.
3. Click the **Advanced** tab.
4. Click **Environment Variables.**
5. Under **System Variables,** click **New.**

   The New System Variable dialog box opens.
6. In the **Variable name** box, type **`%JAVA_HOME%`**.
7. In the **Variable Value** box, type the full Java install directory path.

Setting the JAVA_HOME Value in DOS

To set the value of JAVA_HOME in DOS, run the following:

```
set JAVA_HOME=<JVM_Install_Directory>
```

Setting the JAVA_HOME Value in UNIX

To set the value of `JAVA_HOME` in UNIX using the Bourne shell (SH, BASH, or KSH), run the following:

```
JAVA_HOME=<JVM_Install_Directory>; export JAVA_HOME
```

Creating a System Account for PPM Center

To install PPM Center and maintain the system after installation, you must create a system account. After you do, always log on to this account on the server machine to perform any PPM Server maintenance—for example, stopping and restarting the PPM Server. This helps to avoid file system permission issues, which can be difficult to track.

Configuring a System Account for PPM Center in Windows

In Windows, configure the user to be a member of the Administrators and Domain Users groups, at a minimum. Provide the user with full access to the installation directory for PPM Center and all of its subdirectories. Provide the Administrators screen group with at least read access to these directories.

Configuring a PPM Center User for PPM Center in UNIX

In UNIX, PPM Center does not require root access for installation. Do not install the server as the root user.

Configure your PPM Center user with the following:

- In the `.profile` file, set the `JAVA_HOME` environment variable.
- In the `.profile` file, set the `ORACLE_HOME` environment variable.
- Set the term to dumb option.

Installing a UNIX Emulator and Telnet Server (Windows)

To run PPM Center on Microsoft Windows, you must have a UNIX emulator such as Cygwin, and a Telnet server such as Microsoft Telnet. For a list of supported UNIX emulators and Telnet servers, see the *System Requirements and Compatibility Matrix.*

To configure private key authentication with secure shell (see *Configuring Private Key Authentication with Secure Shell* on page 105), you use the ssh-keygen utility, which is part of the Cygwin installation. To get this utility, you must enable the Open SSH components during Cygwin installation.

Creating the Database Schemas

To create the empty database schemas (with tables to be populated during installation):

1. Set up the required data, index, and CLOB tablespaces for the PPM Center database schema.

   For information on the minimum size recommended for these tablespaces, see the *System Requirements and Compatibility Matrix.*

   Use locally-managed SYSTEM tablespaces with automatic segment-space management.

   Locally-managed tablespaces eliminate extent fragmentation and provide better performance than dictionary-managed tablespaces.

2. Create each tablespace as shown in the following example for a data tablespace.

```
CREATE TABLESPACE <PPM_Data>
datafile <'/u0/oracle/oradata/G1120/ppm_data01.dbf'>
size <1024m>
AUTOEXTEND ON MAXSIZE <4096m>
EXTENT MANAGEMENT LOCAL AUTOALLOCATE
SEGMENT SPACE MANAGEMENT AUTO;
```

   Oracle has the default TEMP tablespace, which you can resize to improve performance.

The PPM Server requires two separate database schemas to store application data. A DBA can create these schemas before installation. Creating database schemas requires privileges that a DBA might not want to grant to a PPM Center administrator. Either create the database schemas before installation or make sure that a DBA is available during installation.

To create the database schemas and grant the permissions between them:

1. Unpack the PPM Center installation bundle as outlined in *Installing PPM Center* on page 74.

   The `<PPM_Extract>/ppm910` directory is created. The `<PPM_Extract>ppm910/sys` and `<PPM_Extract>/ppm910/system` directories contain the scripts required to create the database schemas.

2. Run the script `CreateKintanaUser.sql` (located in `<PPM_Extract>ppm910/system`) against the database into which you plan to install PPM Center.

   The script prompts for a user name and password, and the tablespaces that the PPM Center database schema are to use.

   ```
   sh> sqlplus system/<Password>@<SID> \
   @CreateKintanaUser.sql \
   <PPM_Username> \
   <Password> \
   <Data_Tablespace> \
   <Index_Tablespace> \
   <CLOB_Tablespace>
   ```

3. To enable the PPM Center database user to create views and synonyms in the RML schema, connect to the database that contains the RML schema, and then issue the following SQL statements:

   ```
   grant create any synonym to &KNTA_USERNAME;
   grant create any view to &KNTA_USERNAME;
   grant drop any synonym to &KNTA_USERNAME;
   grant drop any view to &KNTA_USERNAME;
   grant comment any table to &KNTA_USERNAME;
   ```

4. Run the `CreateRMLUser.sql` script, which is located in the `/system` directory.

   The script prompts for a user name and password for the Reporting Meta Layer (RML) schema, tablespace information, and the PPM Center

database schema user name. The script creates the RML schema and establishes the permissions between the RML and the PPM Center database schema.

Because the RML schema contains only views (and no physical objects), it does not require a separate tablespace.

```
sh> sqlplus system/<Password>@<SID> \
@CreateRMLUser.sql \
<RML_Username> \
<RML_Password> \
<Data_Tablespace> \
```

5. As the SYS DBA user, run the `GrantSysPrivs.sql` script, which is located in the `<PPM_Extract>/ppm910/sys` directory.

   This script grants the privileges that the PPM Server requires.

If you created the schemas before installation, select **Please use existing schemas** when prompted during installation. Supply the same values as those used in this procedure (that is, the values `<PPM_Username>` and `<RML_Username>`).

Default Permissions for PPM Center Schemas

By default, the PPM Center database schema and RML database schema accounts are granted Oracle database privileges. Some of these privileges are required, but some are not, and can be revoked without affecting the PPM Center system.

The PPM Center database schema account is granted Oracle CONNECT role privileges. If this presents a problem for your organization, you can have your DBA revoke the CONNECT role privilege for the PPM Center database schema account.

Although revoking the CONNECT role privilege does not affect the PPM Center system, the PPM Center schema does require the following grants:

- grant create session
- grant create database link
- grant create procedure

- grant create sequence
- grant create synonym
- grant create table
- grant create view
- grant create trigger

The RML database schema account is granted Oracle RESOURCE role privileges. Because the RML schema requires the RESOURCE role privilege, it cannot be revoked. You can, however, revoke the following privileges, which are also granted to the RML database schema account:

- CREATE CLUSTER
- CREATE INDEXTYPE
- CREATE OPERATOR

Creating a Shared Folder for the server.conf File

In order to implement a server cluster (recommended) you must have a shared folder for the server configuration file (`server.conf`). In addition to giving all nodes in a cluster access to the same `server.conf` file, the shared folder simplifies maintenance of the `server.conf` file.

This section provides instructions on how to prepare the shared folder on both Windows and UNIX systems.

The shared folder described in this section is also required to give users access to the Administration Console interface after you have deployed your PPM Center instance. For information about the Administration Console, see *Administration Tools in the Administration Console* on page 241.

Preparing a Shared Folder for server.conf on a Windows System

To prepare a shared folder for the `server.conf` file on Windows:

1. Create a shared folder on a file server.

   If you plan to configure the server cluster configurations on multiple machines, keep in mind that the nodes in the cluster must all run on the same operating system. Shared access to the `server.conf` file does not support mixed operating systems.

2. Attach the shared folder to each machine that is to host PPM Center.
3. If you plan to host multiple PPM Server clusters (instances) under the same account on a single machine, do the following. Otherwise, proceed to step 4.

   a. Using a text editor, create a file named “ppm_server_conf.env”, and add to it the following text:

      ```
      export PPM_SERVER_CONF_DIR=//<IP_Address>/shared_folder
      ```

   b. Save the `ppm_server_conf.env` file in the *<PPM_Home>* directory and close the file.

4. Open the Control Panel and define an environment variable named `PPM_SERVER_CONF_DIR` for an account that is to run PPM Center nodes on

Windows. The value of the environment variable is the location of the shared folder

Make sure that you use Universal Naming Convention (UNC) notation (`//<IP_Address>/<Shared_Folder>` or `<File_Server_Name>/<Shared_Folder>`) to specify the location of your shared folder.

Preparing a Shared Folder for server.conf on a UNIX System

To prepare a shared folder for the `server.conf` file on UNIX:

1. Create a shared folder on a file server.

   If you plan to configure the server cluster configurations on multiple machines, keep in mind that the nodes in the cluster must all run on the same operating system. Shared access to the `server.conf` file does not support mixed operating systems.

2. Mount the shared folder to each machine that is to host PPM Center.
3. If you plan to host multiple PPM Server clusters under the same account on a single machine, do the following. Otherwise, proceed to step 4.
   a. Using a text editor, create a file named “ppm_server_conf.env”, and add to it the following text:

      ```
      export PPM_SERVER_CONF_DIR=//<IP_Address>/shared_folder
      ```

   b. Save the file to the `<PPM_Home> directory` and close the file.
4. In the `$HOME/.profile` file of the account that is to run PPM Center, add the following line:

   ```
   export PPM_SERVER_CONF_DIR=<Mount_Point>/<Shared_Folder>
   ```

Verifying Port Availability

To successfully install and configure PPM Center, specific ports must be available through the firewall. To expedite installation, make sure that the ports are available before you start to install the product. *Table 3-6* contains summary information about the ports and protocols that PPM Center system components use to communicate.

If you are using an external Web server, you must assign it a port number other than the one assigned to the internal Web server.

Table 3-6. f PPM Center ports and protocols (page 1 of 2)

Communication Channel	Protocols	Ports
Web Browser and Web Server	HTTP/HTTPS	80/443 (configurable)
	If you do not use the default port, you must specify the port number in the URL. Example: `http://Xyz.com:<Port>` You may also be required to open the firewall for ports other than the defaults. On UNIX systems, only processes started by the root user can be assigned a port number that is less than 1024.	
PPM Workbench and App Server	RMI / SRMI	1099 (configurable)
External Web Server and App Server	AJP13	8009 (configurable)
App Server and Database	JDBC	1521 (configurable)
App Server and Mail Server	SMTP	25
App Server and LDAP Server	LDAP	389
App Server and LDAP Server	LDAP over SSL	636
App Server and External System	Telnet	23

Table 3-6. f PPM Center ports and protocols (page 2 of 2)

Communication Channel	Protocols	Ports
App Server and External System	SSH	22
App Server and External System	FTP (control)	21
App Server and External System	FTP Data	Dynamic
App Server and External System	SCP (Secure Copy)	22

Assigning Ports Outside of the Ephemeral Port Range

PPM Center uses many client sockets for its internal communications. These are allocated randomly from the operating system's ephemeral port range.

To prevent conflicts between internal client socket use and your PPM Server port settings, you must check to make sure that the ephemeral port range set for your operating system does not conflict with any of the ports assigned in your `server.conf` file. For information about the default ephemeral port range on your operating system and how to adjust it, see the documentation for your specific operating system.

Installing PPM Center

This section provides the detailed steps used to install the database objects and data that the PPM Server uses. You can perform these steps on any computer (running UNIX or Windows) that has SQL*Net connected to the database on which the PPM Center database objects are to be installed.

Make sure that you have at least 300 MB temporary space and 0.5 to 1 GB swap space available on your operating system.

Installing PPM Center on a Windows System

The installation utility for a Windows server is an executable file that performs the steps required for a basic server installation. The executable and supporting files are contained in a Zip file. The typical installation automatically installs the following components on the server:

- PPM Center program files
- PPM Center database objects
- **Start** menu item
- Windows service

You cannot install PPM Center unless you have SYS DBA privileges or a DBA has already created the required schemas. For more information, see *Key Decisions* on page 46.

To install the PPM Server on Windows:

1. Make sure that you have a UNIX emulator (such as Cygwin) and a Telnet server (such as Microsoft Telnet) installed.

   For a list of supported UNIX emulators and Telnet servers, see the *System Requirements and Compatibility Matrix.*

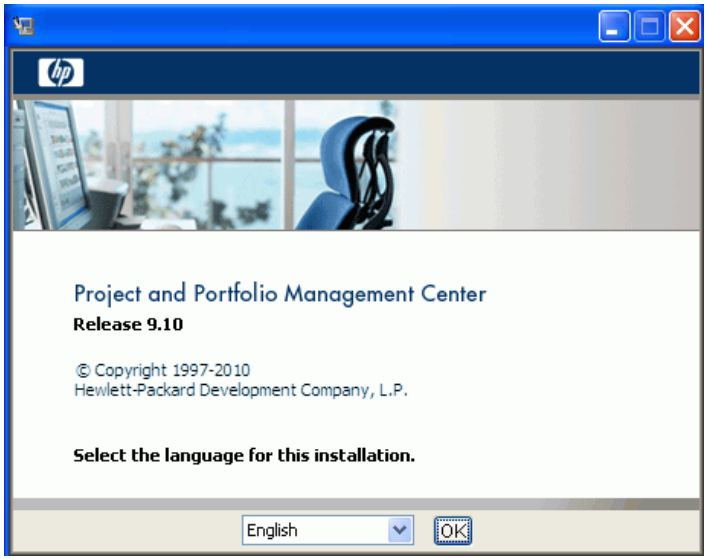
   To configure private key authentication with secure shell (see *Configuring Private Key Authentication with Secure Shell* on page 105), you use the ssh-keygen utility, which is part of the Cygwin installation. To get this utility, you must enable the Open SSH components during Cygwin installation.

2. Extract all files from `ppm-910-install.zip` to the file system.

   The extraction creates the `<PPM_Extract>` `ppm910` directory, which includes the `install.sh` file.

3. From the `ppm910` directory, use Windows Command or Cygwin to run the `install.sh` script.

   The PPM Center installer starts.



4. From the **Select the language for this installation** list, select the language that you want the installer to use to display the installation steps.

Depending on the operating system language of the host machine, the languages available for displaying the installation wizard steps are limited, as shown in the following table.

Operating System Language	Languages Available for Displaying Installation Wizard Steps
English Spanish Italian French Dutch German Portuguese (Brazilian)	English German Spanish French Italian Dutch Portuguese (Brazilian)
Russian	Russian English
Chinese	Chinese English
Korean	Korean English
Japanese	Japanese English
Turkish	Turkish English

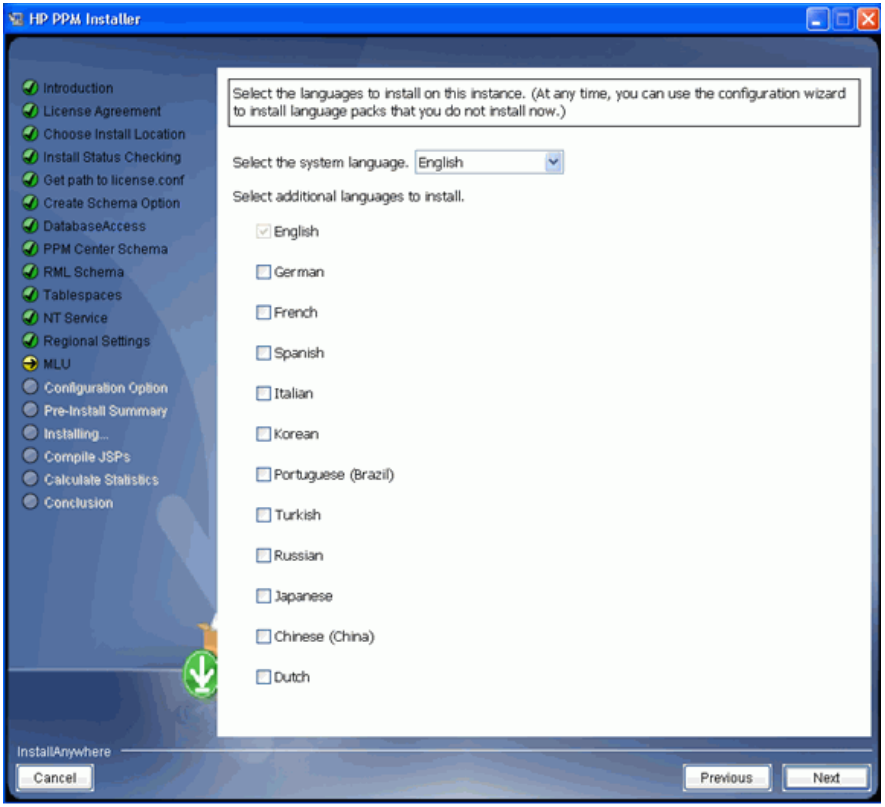
So, for example, if your operating system is Chinese, you can view the installer steps in either Chinese or English. This option in no way affects the language packs that you can install.

The language you select here is not the same as the system language, which you select at a later step.

5. On the License Agreement page, read the agreement carefully, and then select **I accept the terms of the License Agreement.**

6. On the next several pages, provide the information you collected to prepare for the installation (see *Table 3-4* on page 57 in *Collecting Required Information*).

   The MLU (Multilingual User Interface) page is used to select a *system language* and additional languages to deploy on this PPM Center instance.

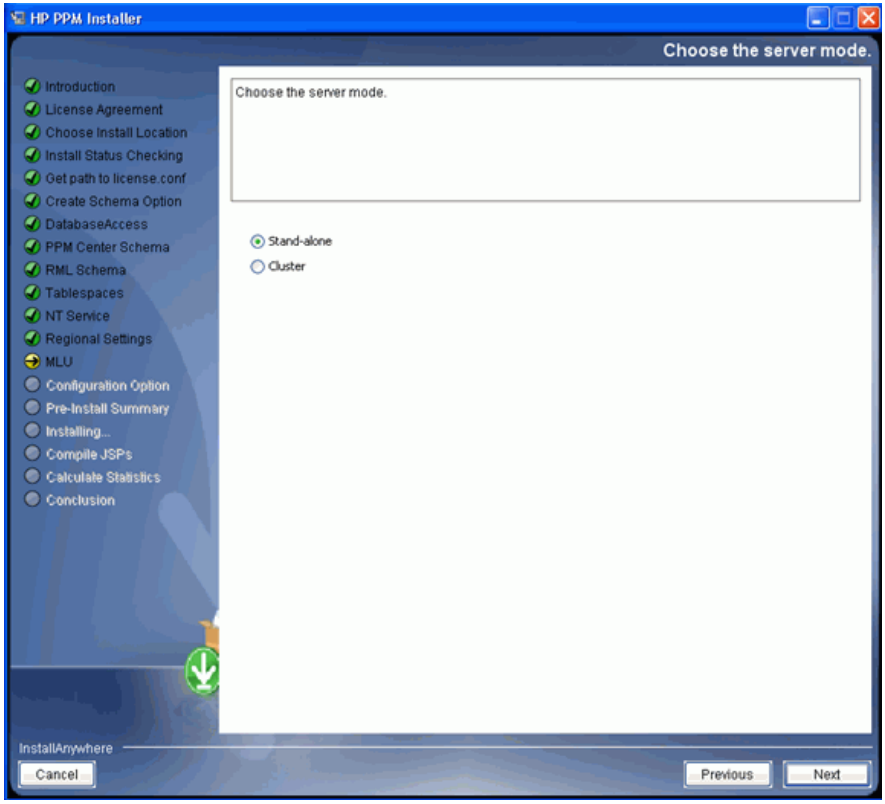


7. From the **Select the system language** list, select the PPM Center system language to use.

   The system language is the language used to generate system-level information such as server start-up and shut-down messages. The system language is also used to display attributes of system data that does not support multiple translations. For more information, see the *Multilingual User Interface Guide.*

8. Select the check boxes for any languages you want to deploy in addition to the system language on your instance.

You can always install languages later, as needed, by running the `kDeploy.sh` script. For information on how to deploy languages after you install PPM Center, see the *Multilingual User Interface Guide.*

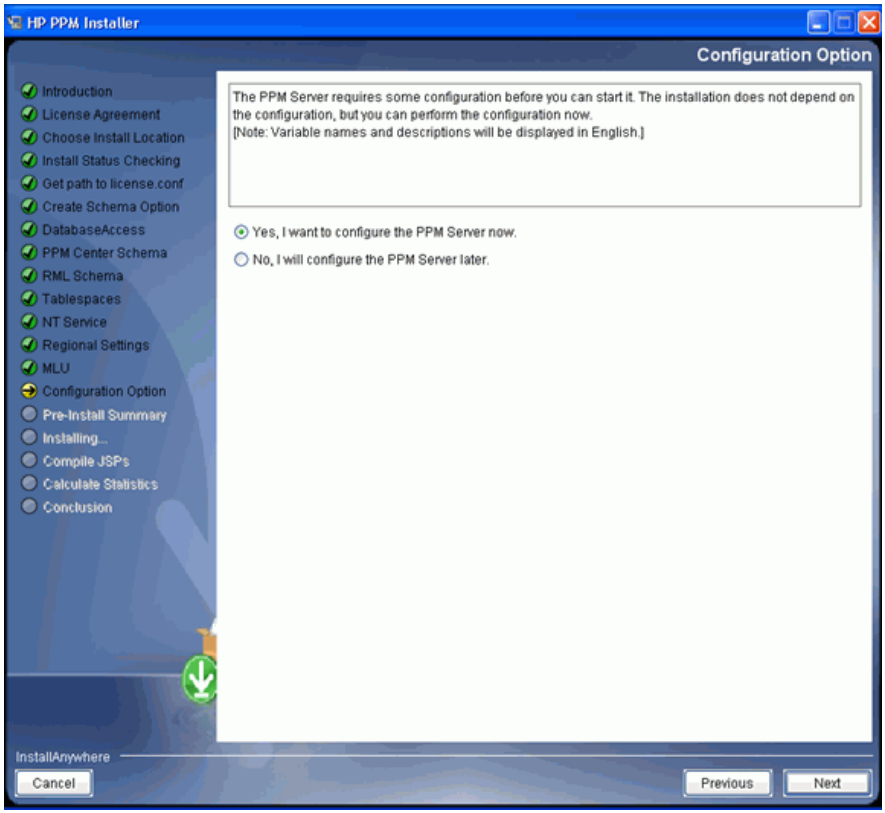


9. On the server mode step, do one of the following.

In this context, the term “server mode” refers to a stand-alone or a clustered type of installation, and is different than the server mode addressed in *Setting the Server Mode* on page 95.

- If you are installing a primary PPM Server for a production instance, select **Cluster.**
- If you are installing a PPM Center testing instance or development instance that will consist of a single PPM Server, leave **Stand-alone** selected.

The Configuration Option page opens next.



10. On the Configuration Option page, indicate whether you want to configure the PPM Server during installation, or later, after installation.

For instructions on configuring the server later, see Chapter 4, *Configuring the System,* on page 93.

11. If you choose to configure the server, the installer displays several pages of server configuration parameters. Provide values for all of the required parameters, which are displayed in red text, and for any optional parameters you want to set.

    For descriptions of and valid values for all listed parameters, see Appendix A, *PPM Center Configuration Parameters,* on page 389.

After you provide all required information, the PreInstall Summary page displays summary information about the installation to be performed.

12. To proceed with the installation, click **Install.**

➤ The installer displays a progress bar that enables you to monitor installation.

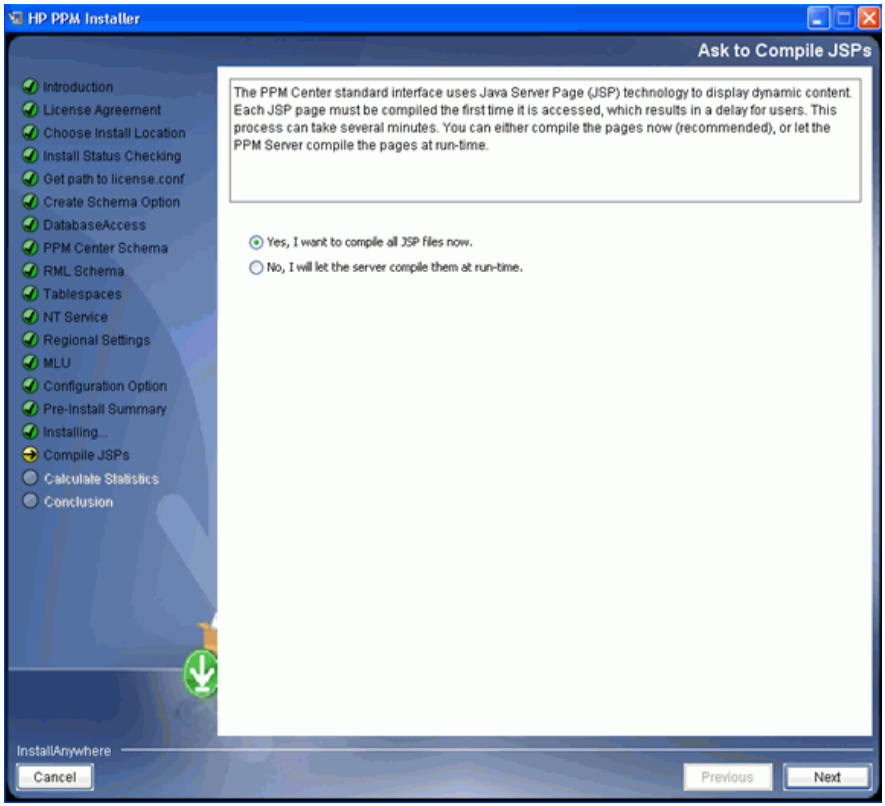
To optimize system performance, the installation script rebuilds statistics for the Oracle optimizer. For the installation procedure to perform this step, you must have the following grants to the schema in place:

- `grant select on v_$parameter to <PPM_Schema>`
- `grant select on v_$mystat to <PPM_Schema>`
- `grant select on v_$process to <PPM_Schema>`
- `grant select on v_$session to <PPM_Schema>`
- `grant execute on dbms_stats to <PPM_Schema>`

The `GrantSysPrivs.sql` script (located in the `<PPM_Extract>/ppm910/sys` directory) performs these required grants.

If you did not run the `GrantSysPrivs.sql` script before you started installation, do it now (with SYS DBA access).

13. The first time a user requests a page in the PPM Center standard interface, the server must compile the page.



To eliminate this initial performance drag, leave **Yes, I want to compile the JSP files now** selected. To skip this step, select **No, I will let the server compile them at run-time.**

For information on how to run the `kJSPCompiler.sh` script to precompile all of the JSP pages before users request the, see *kJSPCompiler.sh* on page 495.

14. After successful installation, PPM Center is installed as a Windows service. You can view the properties for this service through the Services Control Panel item. To complete the service setup, start the PPM Center service from the Control Panel (**Start > Administrative Tools > Services.**)

HP recommends that you set the startup type to **Automatic** so that the PPM Server restarts automatically after the computer is restarted. If you have generated a custom PPM Center user, specify this user name for the "Log On As" value.

PPM Center comes with an Administrator user with the default username/password combination admin/admin. HP recommends that you change the password for the administrator user after you install PPM Center.

An item that corresponds to the Windows service name that you specified during installation is added to the **Start** menu. The menu provides links to PPM Center documentation and an uninstall program.

If you did not configure the PPM Server during installation, see *Configuring or Reconfiguring the PPM Server* on page 99.

Installing PPM Center on UNIX Systems

To install the PPM Center on UNIX:

1. Download the bundle and copy it to a directory, referred to as `<PPM_Extract>` in the following steps.
2. To extract the files into the `<PPM_Extract>` directory from the download bundle, at a command prompt, type one of the following:

   ```
   unzip ppm-910-install.zip
   ```

   Alternatively,

   ```
   jar xvf ppm-910-install.zip
   ```

   All the files and scripts required for PPM Center installation are extracted (to `<PPM_Extract>`). The installer prompts for the software install directory. You can specify any directory for installation.

   The `<PPM_Extract>/ppm910` directory that results from the extraction contains the `install.sh` shell script.

3. To start the installation, run the installation script (as the system account for PPM Center) and specify the installation mode.

   Example

   ```
   sh ./install.sh -i[swing|console]
   ```

   where

-i swing	Swing mode is an interactive, GUI-based installation mode that requires an X Window session. A wizard guides you through the installation steps.
-i console	Console mode is the interactive command-line mode. The installation script runs within the terminal session and you respond to the prompts.

   To optimize system performance, the installation script rebuilds statistics for the Oracle optimizer. For the installation procedure to perform this step, you must have the following grants to the schema in place:

   - `grant select on v_$parameter to <PPM_Schema>`
   - `grant select on v_$mystat to <PPM_Schema>`
   - `grant select on v_$process to <PPM_Schema>`
   - `grant select on v_$session to <PPM_Schema>`
   - `grant execute on dbms_stats to <PPM_Schema>`

   The `GrantSysPrivs.sql` script (located in the `<PPM_Extract>/ppm910/sys` directory) performs these required grants.

4. If you did not run the `GrantSysPrivs.sql` script before you started installation, do it now (with SYS DBA access).

PPM Center comes with an Administrator user with the default username/password combination admin/admin. HP recommends that you change the password for the administrator user after you install PPM Center.

Configuring the FTP Server on Windows

PPM Center uses FTP to move files between machines. To transfer files between machines on a network, each source and destination machine must be running an FTP server. On UNIX platforms, this is standard functionality, but machines running Windows require additional FTP server configuration to function with PPM Center.

Before you configure the FTP server on a machine, make sure that the Windows user account (which PPM Center uses to open a connection) has access to the directories to which files are to be moved. Some FTP servers require that you map these directories to FTP aliases, and a configuration utility is usually provided for this (for example, for Microsoft IIS, the utility is Internet Services Manager).

On Windows, most FTP servers, including Microsoft IIS, do not support drive letters. If you use FTP in PPM Center, the drive letter is removed from the base path. If your base path is `d:\ppm910`, then FTP tries to start from the ftp root directory and FTP fails.

To work around this, you must create an FTP alias. (For example, map `/ppm910` to `D:\ppm910`.) This way, FTP and Telnet point to the same disk location.

Configure the FTP server according to directions that the vendor has provided. For the File and Directory Chooser components to work, you must set the FTP server directory listing style to UNIX, and not to MS-DOS.

To set the directory listing style to UNIX:

1. In Windows, open the Internet Services Manager.
2. In the left pane, under **Console Root,** open the Internet Information Server.
3. Select the machine name.
4. Right-click the Default FTP site displayed in the right pane, and then click **Properties** on the shortcut menu.

   The Default FTP Site window opens.
5. Click the **Home Directory** tab.

6. Under **Directory Listing Style,** click **UNIX.**

7. Test the connection by trying to open a session manually. If you can open an FTP session and navigate from one directory to another, then PPM Center can do this too.

Contacting Support

If you encounter problems with your installation or if have questions, contact HP Software Support Web site (hp.com/go/hpsoftwaresupport). Before you contact HP Software Support, have the following information ready:

1. Open the `ppm_install.log` file (located in the `<PPM_Home>/install_910/logs` directory) in a text editor.

   This file provides information about what part of the installation failed.

2. Search the `ppm_install.log` file for an error message that is specific to installation failure.

3. Place all of the files in the `<PPM_Home>/install_910/logs` directory in a compressed file.

   The installation utility creates a separate log directory for each installation attempt. In the most recent directory, examine each file to determine exactly where the PPM Server has failed. The log file contains information about which failed action it attempted.

Downloading and Installing Service Packs

HP occasionally delivers product service packs to licensed PPM Center customers. You can use the `kDeploy.sh` script (a command-line tool) to install service packs.

To install a service pack, you must make sure that you have the required system privileges. For information about these privileges, and when to grant them, see *Key Decisions* on page 46.

PPM Center service packs are distributed as deployments. Deployments are software bundles that contain files and data. You can get service packs from the HP support Web site.

To locate and download a service pack to install on your PPM Center instance:

1. Go to the Patches search page on the HP support Web site.
2. In the **Select your search criteria** section, provide the information listed in the following table:

Field	Information
Product	Select **Project and Portfolio Management.**
Product version	Select the version of the PPM Center software you have installed.
Operating system	Select the operating system on which PPM Center is installed.
Sub-product	Leave **All Sub Products** selected.

3. Provide any optional criteria you want to specify, and then click **Search.**

   The **Title** section of the Patches search page lists the service packs that match your search criteria.

4. Click the link for the service pack to download.

   The download page for the selected service pack lists detailed information.

5. In the **Should I download?** section, click **DOWNLOAD PATCH.**

6. In the File Download message window, click **Open.**
7. Copy the deployment JAR file to the `<PPM_Home>` directory.

   Deployments are in the following format:

   `ppm-<Ver>-<ID>[.#].jar`

   where

`<Ver>`	represents the PPM Center version for which the service pack was created
`<ID>`	represents the unique identifier for service pack
`[.#]`	represents an optional revision number for the deployment, and may not be included in the deployment name.

If multiple languages are deployed on your PPM Center instance, after you install a service pack, you must redeploy the language packs to restore the MLU. For more information, see the *Multilingual User Interface Guide.*

Example

To install Service Pack 1:

1. Back up your database.
2. Stop the PPM Server.

   You cannot install the service pack on an active server. For information about how to start and stop the server, see *Starting and Stopping the PPM Server* on page 95.

3. Run the following command:

   ```
   sh ./kDeploy.sh -i SP1
   ```

4. As the script runs, respond to the prompts.

5. Start the PPM Server.

To obtain a list of all service packs applied to your PPM Center instance, run the command `sh ./kDeploy.sh -l`.

For more information about the `kDeploy.sh` script, see *kDeploy.sh* on page 489.

Contacting Support

If problems occur during service pack installation, go to the HP Software Support Web site (hp.com/go/hpsoftwaresupport).

Handling Backup Files Related to Service Pack Installation

During a service pack installation, the installer backs up all of the existing files that are to be replaced. After multiple service pack installations, the backup files can take up significant space.

Eventually, the backed up files can consume so much space that service pack installation fails. To prevent this from occurring, do one of the following:

- Use the `kDeploy.sh` script to install service packs without creating backup files.

  Example

  To install Service Pack 3 without creating a backup, run the script as follows:

  ```
  sh ./kDeploy.sh -i SP3 -B
  ```

- Specify that backed up files are deleted after service pack installation. To do this, run the `kDeploy.sh` script, as follows:

  ```
  sh ./kDeploy.sh -tidy
  ```

Verifying PPM Center Installation

To verify the installation:

1. Check the logs produced during installation. (Located in the `<PPM_Home>/install_910/logs` directory)
2. Complete the PPM Center configuration and perform all post-installation tasks (covered in later chapters).
3. Start the PPM Server.
4. Log on to PPM Center.

   All PPM Center clients use the same base URL, which is the Web location (top directory name) of the PPM Server. To obtain the URL, open the `server.conf` file, which is located in the `<PPM_Home>` directory. The URL is the value specified for the `BASE_URL` parameter.

5. Start the PPM Workbench.
6. Run a report. (For instructions, see *Running Server Reports from the Admin Tools Window* on page 263.)
7. Create a request and test the graphical view of the request. (For instructions, see the *HP Demand Management User's Guide*.)
8. Add a portlet to a PPM Dashboard page and export the page in PDF format. (For instructions, see the *Getting Started* guide.)

   Before you can export a page in PDF format, you must enable that functionality. For information, see *Enabling Export to PDF* on page 114.)

9. Create a project and a work plan. (For instructions, see the *HP Project Management User's Guide*.)
10. Create a staffing profile. (For instructions, see the *HP Resource Management User's Guide*.)
11. Create a time sheet. (For instructions, see the *HP Time Management User's Guide*.)

Optional Installations

This section provides descriptions of additional products that you can install and set up to work with PPM Center.

Installing HP Project and Portfolio Management Best Practices

HP PPM Best Practices provides customers with experience-derived information and advice about configuring and using HP Portfolio Management and HP Program Management. Best Practices installation places various workflows and request types on your system to help optimize your use of HP Program Management and HP Portfolio Management.

For more information, see *About PPM Center Best Practices Installation* on page 48. Before you start to install Best Practices, make sure that *all* of the conditions described in *Requirements for Installing Best Practices* on page 48 have been met.

To install Best Practices:

1. Start the PPM Server from the command line.
2. Set your server to RESTRICTED mode.

   Although setting your server to RESTRICTED mode is optional, HP recommends that you do so. In RESTRICTED mode, the PPM Server enables only users with Administrator access granted to log on.

   You can use the `setServerMode.sh` script to set the server to RESTRICTED mode. (See *Setting the Server Mode* on page 95.)

3. Run the `kDeploy.sh` script, as follows.

```
sh ./kDeploy.sh -best-practices
```

For more information about the `kDeploy.sh` script, see *kDeploy.sh* on page 489.

Verifying HP Project and Portfolio Management Best Practices Installation

To verify that Best Practices is successfully installed, run the `kDeploy.sh` script, as follows.

```
sh ./kDeploy.sh -l
```

This returns a list of the deployed bundles in an instance.

Installing HP Deployment Management Extensions

If you plan to install an HP Deployment Management Extension, you must do so after you install and configure PPM Center, and before you use PPM Center for processing.

You are not required to stop the PPM Server(s) before you install an Extension. However, HP recommends that you install the Extension when no users are logged on to the system. Consider placing the server in RESTRICTED mode before you install.

Although setting your server to RESTRICTED mode is optional, HP recommends that you do so. In RESTRICTED mode, the PPM Server enables only users with Administrator access granted to log on.

You can use the `setServerMode.sh` script to set the server to RESTRICTED mode. (See *Setting the Server Mode* on page 95.)

For specific information on how to install an HP Deployment Management Extension, see the documentation for the Extension you purchased.

To install an Extension successfully, you must make sure that you have the required system privileges. For information about these privileges, and how to grant them, see *Key Decisions* on page 46.

What to Do Next

After you have successfully installed PPM Center, delete all subdirectories of the `install_910` directory, except for the `logs` subdirectory.

Proceed to Chapter 4, *Configuring the System*, on page 93.

4 Configuring the System

About this Chapter

This chapter provides detailed information about how to configure the basic components of the PPM Center system and to start and stop the PPM Server. It also includes information that PPM Center users need to know in order to use the PPM Workbench.

For advanced PPM Center system configuration information, including how to configure an external Web server and PPM Server clusters, see Chapter 5, *Advanced System Configuration,* on page 135.

(UNIX only) Setting the ulimit Value

On UNIX systems, PPM Center (through the `kStart.sh` script) uses the ulimit utility to set the maximum number of open file descriptors to 1000. In a server cluster configuration, this setting may be too low, causing nodes to come down.

If the default ulimit setting does not meet your requirements, reset it as follows:

1. Navigate to the `<PPM_Home>/bin` directory and open the `kStart.sh` file in a text editor.
2. Locate the following text:

   ```
   if [ $HOST_TYPE = UNIX ]; then
       ulimit -n 1000
       umask 022
   ```

3. Change the ulimit value as follows:

   ```
   ulimit -n 1100
   ```

4. Repeat step 1 through step 3 for each additional node in the cluster.

   For example, if your server cluster consists of five nodes, then specifying the ulimit value of 1100 for each allocates 5500 open file descriptors for the PPM Server cluster deployment.

Starting and Stopping the PPM Server

To help ensure that your PPM Center system operates smoothly, HP recommends that you stop and restart the PPM Server once a month. This section provides information about how to start the PPM Server on a single-server system.

Unless otherwise indicated, “the server” refers to the PPM Server, and not the server machine.

For information about configuring and running a clustered configuration, see *Server Cluster Configurations (Recommended)* on page 28 and *Configuring a Server Cluster* on page 172.

Setting the Server Mode

PPM Center supports the following server modes:

- **NORMAL.** In NORMAL mode, all enabled users can log on, and all services are available, subject to restrictions set in `server.conf` parameters.
- **RESTRICTED.** In RESTRICTED mode, the server enables only users with Administrator access granted to log on. The server cannot run scheduled executions, notifications, or the concurrent request manager while in this mode.

  Before you can install an HP Deployment Management Extension, you must set the server to RESTRICTED mode.
- **DISDABLED.** DISABLED mode prevents server startup. A server enters disabled mode only after a PPM Center upgrade exits before the upgrade is completed.

Setting the Server Mode with setServerMode.sh

The `setServerMode.sh` script, located in the `<PPM_Home>/bin` directory, sets the server mode in situations where you want to obtain exclusive access to a running server.

To set the server mode using the `setServerMode.sh` script:

1. From the command line, change to the `<PPM_Home>/bin` directory, and run the `setServerMode.sh` script.

   The Run dialog box opens.

2. In the **Open** field, type the following:

   ```
   sh ./setServerMode.sh <Mode_Name>
   ```

   where `<Mode_Name>` represents the `NORMAL, RESTRICTED,` or `DISABLED` server mode.

   For example, to set the server to restricted mode, in the **Open** field, type:

   ```
   sh ./setServerMode.sh RESTRICTED
   ```

For more information about the `setServerMode.sh` script, see *Setting the Server Mode* on page 95. For more information about the `kConfig.sh` script, see *kConfig.sh* on page 486.

Setting the Server Mode Using kConfig

To set the server mode using the `kConfig.sh` script:

1. Run `sh ./kConfig.sh` (located in the `<PPM_Home>/bin` directory).
2. Select **Set Server Mode.**
3. In the list, select **Restricted Mode.**
4. Click **Finish.**
5. Run the `kUpdateHtml.sh` script.

Starting and Stopping the PPM Server on Windows

To start the server on a Windows system.

If your PPM Center instance includes multiple nodes in a cluster configuration, you must start these nodes one at a time. Make sure that you wait until each node is fully started before you start the next node.

1. Open the Control Panel.
2. Double-click **Administrative Tools.**
3. Double-click **Services.**
4. Right-click the name of the PPM Center service, and then click **Start** on the shortcut menu.

   The service name starts with “HP PPM.”

If you prefer to use the Windows shell command line to start servers instead of using Windows Services, you can use the `kStart.sh` script. For information about the `kStart.sh` script, see Appendix B, *Server Directory Structure and Server Tools,* on page 481

To stop the server on a Windows system:

1. Open the Control Panel.
2. Double-click **Administrative Tools.**
3. In the Administrative Tools window, double-click **Services.**
4. In the Services window, right-click the name of the PPM Center service, and then click **Stop** on the shortcut menu.

   The service name begins with “HP PPM.”

Using the Windows Shell Command Line to Stop PPM Servers

If you prefer to use the Windows shell command line to stop servers instead of using Windows Services, you can use the `kStop.sh` script. For information about the `kStop.sh` script, see Appendix B, *Server Directory Structure and Server Tools,* on page 481.

If the `REMOTE_ADMIN_REQUIRE_AUTH` parameter is set to `true,` users running `kStop.sh` to shut down the PPM Server must supply a valid PPM Center user name and password. If the parameter is set to `false,` any user with access to the `kStop.sh` script can shut down the server. For information about the `REMOTE_ADMIN_REQUIRE_AUTH` parameter, see *REMOTE_ADMIN_REQUIRE_AUTH* on page 445.

Starting and Stopping the PPM Server on UNIX

To start the server on UNIX.

If your PPM Center instance includes multiple nodes in a cluster configuration, you must start these nodes one at a time. Make sure that you wait until each node is fully started before you start the next node.

1. Change to the `<PPM_Home>/bin` directory.
2. Run the `kStart.sh` script, as follows.

```
sh ./kStart.sh
```

For more information about `kStart.sh`, see *kStart.sh* on page 502. For information about how to start servers in a cluster, see *Starting and Stopping the PPM Server* on page 95.

To stop the server on UNIX.

If the REMOTE_ADMIN_REQUIRE_AUTH parameter is set to `true`, users running `kStop.sh` to shut down the PPM Server must supply a valid PPM Center user name and password. If the parameter is set to `false`, any user with access to the `kStop.sh` script can shut down the server. For information about the REMOTE_ADMIN_REQUIRE_AUTH parameter, see *REMOTE_ADMIN_REQUIRE_AUTH* on page 445.

1. Navigate to the `<PPM_Home>/bin` directory.
2. Run the `kStop.sh` script as follows.

   ```
   sh ./kStop.sh -now -user <User_Name>
   ```

   Make sure that you type a valid user name that has Administrator privileges.

For more information about `kStop.sh`, see *kStop.sh* on page 502. For information about how to stop servers in a cluster, see *Starting and Stopping Servers in a Cluster* on page 194.

Configuring or Reconfiguring the PPM Server

If you configured the PPM Server during installation, it is probably not necessary to reconfigure it unless your environment or requirements have changed. If you did not configure the server during installation, configure it now.

You can perform most of the configuration using the procedure described in the next section, *Standard Configuration*. In some cases, however, configuration requires custom parameters. For information about when and how to configure the server using custom parameters, see *Defining Custom and Special Parameters* on page 102.

The server configuration tool runs in both console and graphical modes. To run in graphical mode in a UNIX environment, the tool requires an X Window session.

Standard Configuration

This section provides the steps for standard PPM Server configuration and all of the settings required for a typical PPM Center installation.

To configure the PPM Server:

1. From a DOS or UNIX command line, run the `kConfig.sh` script (located in the `<PPM_Home>/bin` directory) as follows.

   - To run the script in graphical mode, type:

     ```
     sh ./kConfig.sh -i swing
     ```

     (UNIX only) Run this utility in an X Window session.

   - To run the script in console mode, type:

     ```
     sh ./kConfig.sh -i console
     ```

2. Follow the configuration wizard prompts to complete the configuration.

   Specify a value for every parameter required for your system environment. To determine the correct value to provide for a parameter, move your cursor over the parameter name and display the tooltip text. For more information, see Appendix A, *PPM Center Configuration Parameters*, on page 389.

   All confidential information (such as passwords) is hidden and encrypted before it is stored.

   Do not change default values unless you are sure that the default value does not meet the requirements of your organization.

   Always use forward slashes (/) as a path separator, regardless of your operating system environment. PPM Center automatically uses the correct path separators when communicating with Windows, but expects to read only forward slashes on the configuration file.

   Specify any required parameters on the Custom Parameters page.

If you have installed multiple stand-alone PPM Center instances on a single machine, you must set the parameters described in the following table for each instance.

Parameter	Description
APP_SERVER_NAMING_SERVICE_RMI_PORT	JBoss Naming Service RMI port. For a stand-alone PPM Server, specify a port that is unique for the instance. For a node in a server cluster, specify a port that is unique for the node in the cluster.
APP_SERVER_NAMING_SERVICE_BINDING_PORT	JBoss Naming Service Binding port. For a stand-alone PPM Server, specify a port that is unique for the instance. For a node in a server cluster, specify a port that is unique for the node in the cluster.
APP_SERVER_JRMP_INVOKER_RMI_PORT	JBoss JRMP Invoker RMI port to which RMI clients connect when communicating through the proxy interface. For a stand-alone PPM Server, specify a port that is unique for the instance. For a node in a server cluster, specify a port that is unique for the node in the cluster.
APP_SERVER_UIL2_BINDING_PORT	Protocol listening port for the JBoss Application Server UIL2 service. For a node in a server cluster, specify a port that is unique for the node in the cluster.

3. If you have no custom parameters to add, leave **Custom Parameters** empty. If you require custom parameters, see *Defining Custom and Special Parameters* for instructions on how to specify them.

   The configuration wizard writes the configuration parameters to the `server.conf` file and generates other files that the PPM Server requires.

4. Stop, and then restart the server.

   For information about how to stop and start the server, see *Starting and Stopping the PPM Server* on page 95.

You can also modify parameters directly in the server configuration file, which is described in Appendix A, *PPM Center Configuration Parameters,* on page 389.

If you modify parameters directly, be sure to run the script `kUpdateHtml.sh` after you make your changes.

Defining Custom and Special Parameters

In addition to the standard parameters that HP supplies, PPM Center supports two additional kinds of server parameters:

- **Custom parameters.** You can define your own custom parameters.

  Custom parameter names must have the prefix `com.kintana.core.server`.

  Example

  To add a custom parameter named `NEW_PARAMETER`, in the **`Key`** field, type the following:

  `com.kintana.core.server.NEW_PARAMETER`

  Parameters that you add to the custom parameters list are accessible as tokens from within the application. These tokens are in the format `[AS.parameter_name]`.

- **Special Parameters.** HP has created configuration parameters that you can use in special situations after you add them to the custom parameters folder. *Table 4-1* lists these special parameters.

If you edit the `server.conf` file directly, you must then run the `kUpdateHtml.sh` script (located in the `<`*`PPM_Home`*`>/bin` directory) to rebuild the startup files. To implement your changes, you must stop, and then restart, the PPM Server. After you restart the server, you can run the Server Configuration Report to view the new or modified parameter values in the `server.conf` file.

For information about the `kConfig.sh` script, see *kConfig.sh* on page 486. For information about the `kUpdateHtml.sh` script, see *kUpdateHtml.sh* on page 505.

Table 4-1. Special configuration parameters

Parameter Name[a]	Description	Sample Value
DB_ CONNECTION_ STRING	If the `JDBC_URL` parameter is specified, then the security identifier (SID) of the database on which the PPM Center schema resides is requested. It is assumed that the connect string for this database is the same as the SID. However, this is not always the case. If the connect string (for connecting to the database using SQL*Plus from the server machine) is different than the database SID, add this parameter and supply the correct connect string.	PROD
NON_DOMAIN_ FTP_ SERVICES	**Windows environment only:** To open an FTP session, FTP servers on Windows typically require the Windows domain name and user name (in the form `Domain\User name`). By default, PPM Center includes the domain name and user name in an FTP session to a Windows computer. If you use an FTP server that does not require the domain name, you can use this parameter to override the default functionality. For more information, contact HP Software Support Web site (hp.com/go/ hpsoftwaresupport).	WAR-FTPD
TEMP_DIR	This parameter defines a PPM Center temporary directory. This defaults to a `temp` subdirectory of the `logs` directory. If you use this parameter, make sure that you provide the full directory path.	`C:/ppm/ logs/temp`

a. The parameter names listed in the table are shortened versions of the actual names, all of which start with the string `com.kintana.core.server.` For example, the full name of the `TEMP_DIR` parameter is `com.kintana.core.server.TEMP_DIR.`

Enabling Secure RMI (Optional)

To enable SRMI (RMI over SSL):

1. Create a keystore for SSL to use.

   You can use the Java keytool application to create a keystore. For information about the keytool application, see the Oracle documentation online.

   Use the keystore password that you use to run keytool to define the `KEY_STORE_PASSWORD` (see step 2).

2. In the `server.conf` file, specify values for the three parameters:

   - `RMI_URL`
   - Set the `KEY_STORE_FILE` parameter to point to the keystore file.
   - Set the `KEY_STORE_PASSWORD` to the keystore password you created in step 1. This password can be encrypted.

Example

If you ran keytool to create the file `security/keystore` relative to the *<PPM_Home>* directory, and you used the password "welcome," ran on host "caboose," and listened on port 1099, your `server.conf` parameters would look as follows:

```
com.kintana.core.server.RMI_URL=rmis://caboose:1099/
KintanaServer
com.kintana.core.server.KEY_STORE_FILE=security/keystore
com.kintana.core.server.KEY_STORE_PASSWORD=welcome
```

You can create a self-signed certificate.

Configuring Private Key Authentication with Secure Shell

This section provides information on how to configure private key authentication with secure shell (SSH). The procedure is based on the following assumptions:

- SSH is installed.
- The SSH server is configured for private key authorization.
- The ssh-keygen utility is part of the Cygwin installation. (To get this utility, you must enable the Open SSH components during Cygwin installation.)

Before you configure private key authentication, do the following:

- Verify that the PPM Center user account can be used to log on to the remote host through the SSH session.
- Add the RSA certificate information of the remote host to the ssh `known_hosts` file, which is located in the *<PPM_Home>* directory.

To add the RSA certificate of the remote SSH host to the PPM Server SSH `known_hosts` file:

1. Log on to the PPM Server as the PPM Center user.
2. From the command line, run the following:

   ssh *<User_ID>@<Remote_Host>*

   The first time you run this command, you are prompted to indicate whether you want to continue.
3. Type **yes.**
4. Terminate the SSH connection with the remote host.

To set up private key authentication with SSH:

1. Generate the private/public key pair on the PPM Server.
2. Add the generated public key to the remote `SSH Authorized_Key` file.
3. Configure the PPM Server.

The following sections provide the steps required to perform each of these tasks.

Generating the Private and Public Keys

To generate the private/public key pair on the PPM Server:

1. Log on to the PPM Server machine as the PPM Center user.
2. Change directory to the home directory defined for the PPM Center user on the operating system.
3. Run the following SSH utility.

   ```
   ssh-keygen -t rsa -b 1024
   ```

   PPM Center only supports the RSA key type, and not the DSA key type.

   Do not provide the “passphrase.”

4. Press **`Enter`** twice.
5. Verify that the `<PPM_Home>/<PPM_User>/.ssh` directory now contains the `id_rsa` (the private key) and `id_rsa.pub` (the public key) files.

Adding the Public Key to the SSH authorized_keys File on the Remote Host

To append the public key to the remote SSH `authorized_keys` file (remote hosts):

1. Transfer the `id_rsa.pub` file to the remote SSH host machine, in the `/<PPM_User_Home_Directory>/.ssh` directory as `ppm_id_rsa.pub`.

   On the remote UNIX host, the `.ssh` directory is in the `/home/<PPM_User>/` directory. On Windows, the location depends on the user home directory defined during Cygwin installation.

2. Log on to the remote host with the user ID that the PPM Server is to use to connect.

3. Change directory to the `<PPM_Home>/<User_ID>/.ssh` directory and locate the `authorized_keys` file.

   If the `authorized_keys` file does not exist, create it.

4. Append the contents of the `itg_id_rsa.pub` file to the `authorized_keys` file, by running the command:

   ```
   cat ppm_id_rsa.pub > authorized_keys
   ```

5. Repeat these steps on the PPM Server to enable public key authentication from the PPM Server back to itself.

Configuring the PPM Server

To configure the PPM Server:

1. Open the `server.conf` file in a text editor.
2. Add the following server directive to the file.

   ```
   com.kintana.core.server.SSH_PRIVATE_IDENTITY_FILE=/<PPM_Home>/<PPM_User>/.ssh/id_rsa
   ```

3. Change to the `<PPM_Home>/bin` directory.
4. To update the required startup files, run the `kUpdateHtml.sh` script.
5. Restart the PPM Server.

Verifying Server Configuration

To verify the configuration:

1. Open a command-line window outside of the PPM Server.
2. Log on to the PPM Server machine as the PPM Center user, as follows.

   ```
   ssh <User_ID>@<Remote_Host>
   ```

   You should not be prompted for the password. It should log on to the remote host using the RSA key file.

3. On the PPM Server, log on to PPM Center.
4. From the menu bar, select **Administration > Open Workbench.**

   The PPM Workbench opens.

5. From the shortcut bar, select **Environments > Environments.**

   The Environment Workbench page opens.

6. Click **New Environment.**

   The Environment: Untitled window opens.

7. In the **Environment Name** field, type the name of the remote host.

8. In the **Server** section, do the following:

   a. In the **Name** field, type the remote server name.

   b. In the **Type** list, select the operating system type on the remote server.

   c. In the **Username** field, type the user ID you provided in step 2.

   d. In the **Password** field, click the Password (C) button.

      The Enter or Change Password dialog box opens.

      The PPM Workbench requires that you provide a password, regardless of whether the authentication uses RSA.

   e. In the **Enter New Password** and **Confirm New Password** fields, type the password for the user ID you provided in step 2.

   f. Click **OK.**

   g. In the **Base Path** field, type the base path.

   h. In the **Connection Protocol** list, select **SSH2.**

   i. In the **Transfer Protocol** list, select **Secure Copy 2.**

9. Clear the **Enable Client** and **Enable Database** checkboxes.

   The user name specifies the user ID to be used to log on to the destination SSH server. The Environment Checker requires the password. Package line uses the public key file for authentication.

10. Click **Save.**

11. At the bottom left of the window, click **Check.**

    The Check Environment window opens.

12. In the left pane, expand the **Server** folder, and then click **SSH2 Server.**

13. Click **Check.**

In the left pane, an icon to the left of the selected server indicates whether the check succeeded or failed. The right pane displays the details.

Configuring Secure Web Logon (Optional)

This section provides instructions on how to use the built-in JBoss server and HTTPS to configure secure logon on the PPM Center logon page, the Administration Console, and the Change Password page.

To configure your instance to use HTTPS using the JBoss server:

1. Import your SSL certificate or, to create a simple self-signed certificate for testing, run the following command:

   ```
   keytool -genkey -alias <Your_Host> -keystore <Full_Keystore_
   File_Path> -storepass <Store_Password> -keypass <Key_
   Password>
   ```

2. Open the `server.conf` file (located in the *<PPM_Home>* directory) and set the `ENABLE_SSL_LOGIN` server configuration parameter to `true`.

3. Add the following server configuration parameters to the `server.conf` file and set values for each of them:

   `HTTPS_PORT` (see *HTTPS_PORT* on page 421)

   The `HTTPS_PORT` value must be the `HTTP_PORT` number plus 363.

   `HTTPS_WEB_THREAD_MIN` (see *HTTPS_WEB_THREAD_MIN* on page 421)
   `HTTPS_WEB_THREAD_MAX` (see *HTTPS_WEB_THREAD_MAX* on page 421)
   `HTTPS_KEYSTORE_LOCATION` (see *HTTPS_KEYSTORE_LOCATION* on page 421)
   `HTTPS_KEYPASSWORD` (see *HTTPS_KEYPASSWORD* on page 420)

   To get the encrypted password to copy and paste into the `server.conf` file, run the following command:

   ```
   sh kEncrypt.sh –t <Keystore_Password>
   ```

   For information about setting server configuration parameters, see Appendix A, *PPM Center Configuration Parameters*, on page 389.

4. Run the `kUpdateHtml.sh` script (located in the *<PPM_Home>*`/bin` directory), and then restart the servers.

   For information about how to stop and start PPM Servers, see *Starting and Stopping the PPM Server* on page 95.

Additional Considerations for Configuring Secure Web Logon

This section describes additional steps required to set up secure Web logon if your users access PPM Center using Internet Explorer (IE) 6.0 or 7.0, or if you have PPM Servers running on AIX.

Enabling Transport Layer Security on Internet Explorer 6.0 and 7.0

Make sure that users who access PPM Center using Internet Explorer (IE) 6.0 or 7.0 enable Transport Layer Security (TLS) on their browsers. (On the **Advanced** tab of the Internet Options dialog box, select the **Use TLS 1.0** checkbox.)

TLS is disabled by default in IE 6.0, and is enabled by default in IE 7.0.

Editing the server.xml File (AIX Only)

If you have PPM Servers running on AIX, open the `server.xml` file, and add `algorithm="IbmX509"` to it, as follows:

```
<Connector clientAuth="false" secure="true" scheme="https"
debug="0" acceptCount="10" enableLookups="true"
sslProtocol="TLS" address="${jboss.bind.address}" port="22383"
minSpareThreads="5" maxSpareThreads="5" maxThreads="75"
keystoreFile="/<PPM_Home>/key.store" keystorePass="iamnotroot"
algorithm="IbmX509" />
```

Generating Password Security (Optional)

For password security, PPM Center uses a client/server encryption model based on the ElGamal algorithm, which generates a public/private key pair. Passwords are encrypted using the server's public key. Only the server can decrypt the data using the private key. The client application does not have access to decrypted data.

The public and private keys, which are generated during PPM Center installation, reside in `<PPM_Home>/security`. Generate the key pair only once, unless you think that server security has been breached. In that case, regenerate the key pair and reencrypt all passwords.

To regenerate the private and public key pair:

1. From a DOS or UNIX prompt, run the `kKeygen.sh` script, which is located in the `<PPM_Home>/bin` directory.

   ```
   sh ./kKeygen.sh
   ```

2. If information is not available in `server.conf`, you are prompted for the following information:

   - `JDBC_URL` (the server uses this to communicate with the database)

     Example

     ```
     jdbc:oracle:thin: @DBhost.domain.com: 1521:SID
     ```

   - `DB_USERNAME` (username for the PPM Center database schema)

   - `DB_PASSWORD` (password for the PPM Center database schema)

     If you generate new public or private keys, users cannot log on. The old passwords stored in the database are encrypted using the old key. All of the passwords encrypted using the new keys do not match those stored in the database.

As the script run completes, the following two key files are placed in the `<PPM_Home>/security` directory:

- `public_key.txt`
- `private_key.txt`

On a Windows system, anyone can read these files. As the system administrator, make sure that non-trusted users do not have read privilege to the files. On UNIX, the files are read-only for the user running the script. If the user running the script is not the user who started the server, the server cannot read the keys and cannot start.

For more information about the `kKeygen.sh` script, see *kKeygen.sh* on page 495.

Configuring Solaris and Linux Environments to Use HP Deployment Management

PPM Center can connect to a machine on which the environment variable TERM is set to dumb. To enable HP Deployment Management to work in Solaris and Linux environments, you must set this environment variable.

To set the TERM value on Solaris, run:

```
.login:
if ("$TERM" == "dumb") ksh
```

To set the TERM value on Linux, run:

```
.profile:
if [ "$TERM" = "dumb" ]
then
   EDITOR=null
   SHELL=/bin/ksh
   export EDITOR
   VISUAL=null
   export VISUAL
   stty erase '^H'
fi
```

To set the TERM value on Linux 2.1, run:

```
.cshrc:
if ("$TERM" == "dumb") sh
```

Enabling Export to PDF

The PPM Dashboard supports exporting PPM Center portlet content in PDF format in supported languages. To enable this capability you must do the following:

- (Required) Provide the PPM Dashboard with access to Unicode fonts.
- If your PPM Center instance is to be integrated with an external Web server, you must set the `PDF-URL` parameter for the PPM Dashboard as follows:

  `PDF-URL=<Local_Host>`

- If your PPM Center instance is to be integrated with an external Web server *and* the client and application server are to communicate using HTTPS, you must set the `Non-SSL-Port` parameter for the PPM Dashboard as follows:

  `Non-SSL-Port=35000`

  For information about the PDF-URL and Non-SSL-Port parameters for the PPM Dashboard, see *Server Configuration Parameters Related to the PPM Dashboard* on page 466.

- If your PPM Center instance is *not* to be integrated with an external Web server, then set the `Non-SSL-Port` parameter for the PPM Dashboard to the same value set for the `HTTP_PORT` parameter.

Instructions on how to give the PPM Dashboard access to Unicode fonts are provided the following section.

Installing Unicode Fonts for Export to PDF

The Unicode character encoding standard enables the sharing of messages and other items in a multilingual environment when the languages involved span multiple code pages. This means that translated portlet content is exported to PDF files in multiple languages, in one string, and in different locales.

Some operating systems, such as Windows, provide Unicode fonts. If your PPM Center instance runs on an operating system that does not provide

Unicode, you must install a Unicode font on the machine that hosts the PPM Server, and then specify the font location by setting the `com.kintana.core.server.dashboard.PDF-Unicode-Font-File-Path` dashboard server configuration parameter. You can use any Unicode font (for example, Arial Unicode MS or Code2000). You can set additional font directory paths by setting the `com.kintana.core.server.dashboard.Fonts-Directory-Path` dashboard server configuration parameter in the `server.conf` file.

The PPM Dashboard looks for a Unicode font in the standard font locations for the operating system. The following table lists the operating system-specific fonts directories.

Unicode is the default mode that the PPM Dashboard uses. However, if it cannot locate a Unicode font, it switches to regular mode.

Operating System	Fonts Location
UNIX	• /usr/openwin/lib/X11/fonts/TrueType • /usr/X11/lib/X11/fonts/TrueType • /usr/X11/lib/X11/fonts/Type1
HPUX	• /usr/contrib/xf86/xterm/fonts • /usr/lib/X11/fonts/ms.st/typefaces
Linux	• /usr/share/fonts/truetype • /usr/share/fonts/local
Windows	• C:\\WINDOWS\\\Fonts • C:\\WINNT\\\Fonts
AIX	/usr/lpp/Acrobat3/Fonts

For information about how to install fonts, see the documentation for your operating system. For information about how to set server configuration parameters, see Appendix A, *PPM Center Configuration Parameters,* on page 389.

Verifying Client Access to the PPM Server

To verify client access to the PPM Server after installation, log on to a client machine as administrator.

To log on to PPM Center as administrator:

1. On a client machine, start a supported browser, and then specify the URL for your PPM Center site.

   All PPM Center clients use the same base URL, which is the Web location (top directory name) of the PPM Server. To obtain the URL, open the `server.conf` file, which is located in the <*PPM_Home*> directory. The URL is the value specified for the BASE_URL parameter.

   The PPM Center logon screen opens.

2. In the **Username** field, type the user name. (Unless you have changed the default username, as recommended, type **`admin`**.)

3. In the **Password** field, type the password. (Unless you have changed the default password, as recommended, type **`admin`**).

   PPM Center provides this default account for logging on the first time. HP recommends that you disable the admin account or change the password after you generate accounts for all of your users.

4. Click **Submit.**

The PPM Center standard interface opens.

For more information about how to configure licenses and user access, see the *Security Model Guide and Reference.*

Accessing the JBoss JMX Console

The JMX console is an application that ships with JBoss and provides visibility into the JMX MBeans that make up the JBoss application server. You can use the console to view all registered services that are active in the application server and that can be accessed either through the JMX console or programmatically from Java code.

If your PPM Center instance is installed on an AIX system, see *Accessing the JBoss JMX Console on AIX* for information on how to access the JMX console.

To access the JMX console, go to the following URL:

`http://<Server>:<Port>/admin-jmx`

The user name and password required to access the JMX console are encrypted to prevent unauthorized access to the information that the JMX console makes available. They are both stored as SHA-1 hash output in the `jmx-console-users.properties` file, which is located in the `<PPM_Home>/conf/props` directory.

The default user name for the JMX console is “admin.” The default password is “mercitg.”

To change the user name:

1. Open the `jmx-console-users.properties` file.
2. Change the value of the `ADMIN` parameter.
3. Save and close the file.

To obtain the hashed password:

- Run the `kHash.sh` script, (located in the `<PPM_Home>/bin` directory):

  ```
  sh ./kHash.sh -t <Password_Text>
  ```

You cannot change the JMX console password.

Accessing the JBoss JMX Console on AIX

Because of a conflict between Bouncy Castle and IBM security providers, you must do the following to ensure that the JMX console login works correctly:

1. Stop the PPM Server.
2. Navigate to the `<PPM_Home>/server/<PPM_Server_Name>/deployed/itg.war/WEB-INF/lib` directory and locate the `bcpprov-jdk13-133.jar` file.
3. Copy the `bcpprov-jdk13-133.jar` file to the `<JAVA_Home>/jre/lib/ext` directory.
4. Navigate to the `<JAVA_Home>/jre/lib/security` directory and open the `java.security` file.
5. Add the following to the security provider list.

   ```
   security.provider.5=org.bouncycastle.jce.provider.
   BouncyCastleProvider
   ```

6. Navigate to the `<PPM_Home>/server/<PPM_Server_Name>/deployed/itg.war/WEB-INF/lib` directory and delete the `bcpprov-jdk13-133.jar` file.
7. Start the PPM Server.

Configuring or Reconfiguring the Database

The settings described in this section are intended to serve as starting values only. Monitor the database and analyze performance data to fine-tune the settings for your system environment. Tuning an Oracle database involves an Oracle SYS DBA.

The recommendations provided in this section are based on the assumption that PPM Center is the only application that uses the database instance. If other applications share the database, adjust the recommended parameter values accordingly.

For more recommendations on improving PPM Center performance, see the *Deployment Best Practices Guide.*

Database Parameters

This section describes the key Oracle database parameters that can affect PPM Center system performance. It also provides parameter settings recommended for the PPM Center environment.

For detailed information about the Oracle parameters described in the following sections, see your Oracle database documentation.

CURSOR_SHARING

The `CURSOR_SHARING` parameter determines what kind of SQL statements can share the same cursors. Although this optional parameter accepts the following three values, PPM Center supports only the values `SIMILAR` and `EXACT`. HP recommends that you set `CURSOR_SHARING` to `SIMILAR`. `FORCE` is not supported.

- `FORCE`. This value forces statements that may differ in some literals, but are otherwise identical, to share a cursor, unless the literals affect the meaning of the statement.

Setting `CURSOR_SHARING=FORCE` results in errors during PPM Center project creation.

- `SIMILAR`. This (recommended) setting causes statements that may differ in some literals, but are otherwise identical, to share a cursor, unless the literals affect either the meaning of the statement or the degree to which the plan is optimized.
- `EXACT`. This value only enables statements that have identical text to share a cursor. Although this value is supported, HP recommends that you specify `SIMILAR` for the `CURSOR_SHARING` parameter.

DB_BLOCK_SIZE

The `DB_BLOCK_SIZE` parameter is used to specify the size (in bytes) of Oracle database blocks. After the database is created, you cannot change this parameter.

Recommended Setting

Set the `DB_BLOCK_SIZE` parameter value to 8192 (8 KB).

DB_CACHE_SIZE

The `DB_CACHE_SIZE` parameter value specifies the size (in `KB` or `MB`) of the default buffer pool for buffers with the primary block size (the block size defined by the `DB_BLOCK_SIZE` parameter).

Recommended Setting

Specify a `DB_CACHE_SIZE` parameter value of at least 500 (expressed in MB).

HP recommends that you set a value for this parameter, even if `SGA_TARGET` is set.

GLOBAL_NAMES

The `GLOBAL_NAMES` parameter value determines whether a database link must have the same name as the database to which it connects.

Recommended Setting

Set `GLOBAL_NAMES` to `false`. If you set the value to `true`, loopback database link creation fails.

If multiple PPM Center test instances use the same database instance, you must set `GLOBAL_NAMES` to `false`.

To create a loopback database link with this parameter set to `true`:

```
create database link <User_Name.Oracle_SID.Domain_Name> connect
to <User_Name> identified by <Password> using <Oracle_SID>
```

Example 1

```
create database link kinadm.dlngrd02.world connect to kinadm
identified by <Password> using 'dlngrd02'
```

To use the database link you created:

```
select * from <Table_Name>@<Oracle_SID>
```

Example 2

```
select * from clis_users@dlngrd02
```

LIKE_WITH_BIND_AS_EQUALITY

In situations in which the `LIKE` pattern is expected to match very few rows, you can set the hidden parameter `LIKE_WITH_BIND_AS_EQUALITY` to `true`. The optimizer treats the predicate as though it were `COLUMN = :BIND`, and uses column density as the selectivity instead of a fixed five percent selectivity factor. The optimizer treats expressions in the format `[indexed-column like :b1]` in the same way it treats expressions in the format `[index-column = :b1]`.

Oracle uses some defaults to estimate column selectivity for the `LIKE` operator, but most of the time this estimate is not precise and can cause an index path access to be rejected.

Default selectivity varies between releases. For Oracle releases earlier than 9.2.x, the default selectivity is 25 percent, whereas in 9.2.x and later releases, default selectivity is five percent.

As of Oracle 9i, the `LIKE_WITH_BIND_AS_EQUALITY` parameter also enables equality costing for expressions in the following format.

```
function(column) LIKE function(:bind)
```

Recommended Setting

Set the parameter value to `true`.

LOG_BUFFER

The `LOG_BUFFER` parameter value determines the size (in bytes) of the memory area used to save transaction change information. When data is committed, the log buffer is flushed to disk. Small log buffers cause more frequent flushes to disk.

Recommended Setting

For systems with 50 or more concurrent users, set the parameter value to 25 (expressed in MB).

NLS_LENGTH_SEMANTICS

The initialization parameter `NLS_LENGTH_SEMANTICS` lets you create `CHAR` and `VARCHAR2` columns using either byte- or character-length semantics.

Recommended Setting

You must set the `NLS_LENGTH_SEMANTICS` parameter to `CHAR`. After you do, the `VARCHAR2` columns in tables use character-length semantics. This means that if, for example, you declare a column as `VARCHAR2(30)`, the column stores 30 characters, and not 30 bytes. In a multibyte character set, this ensures that adequate space is available.

If you are using a single-byte character set, setting `NLS_LENGTH_SEMANTICS` to `CHAR` makes it easier to transition to a multibyte character set later.

OPEN_CURSORS

Oracle uses cursors to handle updates, inserts, deletes, and result sets that queries return. The `OPEN_CURSORS` parameter value determines the number of cursors one session can hold open at a given time.

Recommended Setting

Set the `OPEN_CURSORS` parameter value to 1000 or higher.

OPEN_LINKS

The `OPEN_LINKS` parameter setting affects only HP Deployment Management. It determines the number of open database link connections to other databases that can be active at a given time.

Recommended Setting

Set the `OPEN_LINKS` parameter value to 20.

OPEN_LINKS_PER_INSTANCE

The `OPEN_LINKS_PER_INSTANCE` parameter determines the maximum number of migrateable open connections globally for each database instance.

Recommended Setting

Set the `OPEN_LINKS_PER_INSTANCE` parameter value to 4.

OPTIMIZER_INDEX_CACHING

The `OPTIMIZER_INDEX_CACHING` parameter gives the optimizer an estimate of the percentage of indexes cached in the buffer cache. The default value is 0. At this setting, Oracle does not “expect” any index to be cached while deciding the best access plan for a query. If you set this parameter to a higher value, Oracle favors using an index in the ‘IN-list iterator’ and nested loop joins.

Recommended Setting

For most PPM Center deployments, set the `OPTIMIZER_INDEX_CACHING` parameter value to 75.

OPTIMIZER_INDEX_COST_ADJ

The OPTIMIZER_INDEX_COST_ADJ parameter is used to tune optimizer behavior for access path selection by making the optimizer more or less prone to selecting an index access path over a full table scan. The lower this parameter value, the likelier it is that the optimizer chooses index scan over table scan.

The default for this parameter is 100%. With this default value, the optimizer evaluates index access paths at regular cost. With any other value, the optimizer evaluates the access path at that percentage of the regular cost. For example, a setting of 50 makes the index access path look half as expensive as with the default value.

Recommended Setting

Although there is no perfect OPTIMIZER_INDEX_COST_ADJ value for every PPM Center deployment, HP recommends that you first set this parameter to between 25 and 40, test the performance of the application at this setting, and, based on the results, adjust the value.

PGA_AGGREGATE_TARGET

Automatic Program Global Area (PGA) memory management is enabled by default (unless PGA_AGGREGATE_TARGET is explicitly set to 0 or WORKAREA_SIZE_POLICY is explicitly set to MANUAL). PGA_AGGREGATE_TARGET defaults to 20 percent of the size of the SGA, unless explicitly set.

The PGA_AGGREGATE_TARGET parameter value determines the aggregate Program Global Area (PGA) memory available to all PPM Server processes attached to the instance. This parameter enables the automatic sizing of SQL working areas used by memory-intensive SQL operators such as sort, group-by, hash-join, bitmap merge, and bitmap create.

PGA_AGGREGATE_TARGET replaces the traditional SORT_AREA_SIZE parameter. Use it with the WORKAREA_SIZE_POLICY parameter set to AUTO.

Recommended Setting

Set the PGA_AGGREGATE_TARGET parameter value to a minimum of 450 MB. For very large databases, you can set the parameter to 1 GB.

PROCESSES

The PROCESSES parameter value determines the maximum number of operating system user processes that can simultaneously connect to the Oracle database. PPM Center uses a pool of database connections. When database activity is required, connections are picked from the pool and the database activity is performed on this existing connection. This process saves the overhead of creating and cleaning up database connections.

Recommended Setting

Set the PROCESSES parameter value to 20 plus the number of total connections that might be used (MAX_DB_CONNECTIONS), times the number of nodes in your server cluster.

Although concurrent usage and usage nature are factors used to determine the number of connections used, a PPM Server rarely uses more than 60 database connections. If a PPM Server cluster configuration is used, each PPM Server might use 60 database connections.

For single-server configurations, set the parameter value to 80 (the default). For a PPM Server cluster configuration with three nodes, set the parameter value to (3 x 60) + (3 x 20) = 240.

SGA_TARGET

The SGA_TARGET parameter value determines the maximum size of all System Global Area (SGA) components combined in the instance. If you specify SGA_TARGET, it is not necessary to specify individual values for SGA components such as SHARED_POOL_SIZE, JAVA_POOL_SIZE, LARGE_POOL_SIZE, and DB_CACHE_SIZE.

Recommended Setting

Set the SGA_TARGET parameter value to 1.66 GB. If you also set the SGA_MAX_SIZE parameter, its value must be higher than the value set for SGA_TARGET.

SHARED_POOL_RESERVED_SIZE

The SHARED_POOL_RESERVED_SIZE parameter helps to ensure that a portion of the shared pool (determined by the SHARED_POOL_SIZE parameter) is set aside for large objects. Reserving an area for large objects helps to ensure that requests for a large number of bytes do not fail as a result of shared pool fragmentation.

If you want to place an object in the reserved area, make sure that the object is larger than the SHARED_POOL_RESERVED_MIN_ALLOC value. HP recommends that you use the default value for the SHARED_POOL_RESERVED_MIN_ALLOC parameter.

Recommended Setting

Set the SHARED_POOL_RESERVED_SIZE parameter value to 10 percent of the shared pool (as determined by the *SHARED_POOL_SIZE* parameter).

SHARED_POOL_SIZE

The shared pool contains shared cursors and stored procedures. The SHARED_POOL_SIZE parameter value determines the size (in bytes) of the shared pool. Larger values can improve performance in multiuser systems, but they use more memory. Smaller values use less memory, but they can degrade the performance of multiuser systems.

Recommended Setting

Set the SHARED_POOL_SIZE parameter value to at least 350 MB.

WORKAREA_SIZE_POLICY

The WORKAREA_SIZE_POLICY parameter value determines whether work areas operate in automatic or manual mode. If the value is set to AUTO, work areas used by memory-intense operators are sized automatically based on the PGA memory that the system uses and the target PGA memory set for the PGA_AGGREGATE_TARGET parameter. If the value is set to MANUAL, work areas are set manually and based on the value of the *_AREA_SIZE parameter.

Recommended Setting

Set the parameter value to AUTO.

Granting Select Privileges to v_$session

If you want PPM Center to keep track of the open database sessions it uses, make sure that a public grant exists on the v_$session dynamic performance table. To do this, connect as SYS to the database that contains the PPM Center database schema, and then issue the following SQL statement.

```
grant select on v_$session to public
```

You typically assign this grant during PPM Center installation or upgrade.

Generating Database Links (Oracle Object Migration)

PPM Center can use database links to communicate with other databases. Usually a database link created and associated with a particular environment in PPM Center can be used in situations such as AutoCompleteSQL.

The following are examples of situations in which database links are used:

- Custom object types designed to provide parameter value lists directly from a source or destination database during HP Deployment Management activities
- Some HP Deployment Management Extensions, such as the Extension for Oracle E-Business Suite, to facilitate HP Deployment Management activities

You can define database links on an as-needed basis. For each database link you require (this probably includes a link to the PPM Center database), issue an SQL statement similar to the following in the PPM Center database schema.

```
create database link DEV_LINK
connect to APPS identified by APPS
using 'DEV'
```

For more information about database links, see:

- *HP Deployment Management Extension for Oracle E-Business Suite Guide*
- *HP Object Migrator Guide*
- *HP GL Migrator Guide*
- Oracle's reference document on the SQL language

Verifying PPM Center Installation

To verify the PPM Center installation:

1. Check the logs produced during installation. (Located in the `<PPM_Home>/install_910/logs` directory)
2. Complete the PPM Center configuration and perform all post-installation tasks.
3. Start the PPM Servers (one at a time).
4. Log on to PPM Center.

   All PPM Center clients use the same base URL, which is the Web location (top directory name) of the PPM Server. To obtain the URL, open the `server.conf` file, which is located in the `<PPM_Home>` directory. The URL is the value specified for the `BASE_URL` parameter.

5. Start the PPM Workbench.
6. Run a report.
7. Create a request.
8. Test the graphical view of the request.

If you encounter a problem that you cannot solve, contact HP Software Support Web site (hp.com/go/hpsoftwaresupport).

Configuring the PPM Workbench to Run as a Java Applet

This section provides the steps to follow to perform the following tasks:

- Enable the SOCKS proxy feature in PPM Center
- Run the PPM Workbench with secure RMI in place
- Provide users on client machines with the required version of the Java plug-in

(Optional) Enabling SOCKS Proxy

Using the SOCKS proxy feature in PPM Center improves security. With SOCKS proxy enabled, all RMI connections are routed through a central server so that each and every PPM Workbench is not required to contact the application server directly. The SOCKS proxy feature also makes it easier to monitor RMI traffic.

To enable the SOCKS proxy feature in PPM Center:

1. Open the `server.conf` file in a text editor.
2. Set the following two parameters:
   - `com.kintana.core.server.SOCKS_PROXY_HOST`
   - `com.kintana.core.server.SOCKS_PROXY_PORT`

   For the `com.kintana.core.server.SOCKS_PROXY_HOST` value, provide the hostname of the SOCKS proxy server.

   For the `com.kintana.core.server.SOCKS_PROXY_PORT` value, specify the port on the SOCKS proxy host that accepts proxy connections.

The PPM Server passes the SOCKS proxy configuration forward to the client applet launcher. Users are not required to configure anything.

To specify a different JRE version in the `server.conf` file, reset the `com.kintana.core.server.WORKBENCH_PLUGIN_VERSION` parameter.

Example

```
com.kintana.core.server.WORKBENCH_PLUGIN_VERSION=1.6.0_07
```

Running the PPM Workbench with Secure RMI (Optional)

To run the PPM Workbench as a Java applet with secure RMI:

- Specify the complete RMI URL, in the following format, when you start the PPM Workbench:

  ```
  java com.kintana.core.gui.LogonApplet rmis://<Host>:<RMI_Port>/<KintanaServer>
  ```

  You can type the RMI URL at the command line or, on Windows, specify it in a shortcut.

Providing Users with the Java Plug-In

The Java plug-in is required to access the PPM Workbench interface. When a user starts the PPM Workbench, the system checks the client browser for the Java plug-in, and then determines whether the correct version is installed.

The supported Java plug-in version is specified by the `WORKBENCH_PLUGIN_VERSION` parameter in the `server.conf` file. If the system cannot find the required version, it directs the user to the Oracle site where the user can download the plug-in and follow the installer wizard prompts to install it.

HP recommends that you leave the `WORKBENCH_PLUGIN_VERSION` parameter default value.

If users who access the PPM Workbench from client machines cannot access the Oracle Web site to download and install the Java plug-in, you must download the plug-in and make it available to users from within the firewall. You can obtain the plug-in directly from the Oracle Software Download site.

Consider restricting PPM Workbench access to users who must perform the kind of configuration and administration tasks performed through the PPM Workbench.

Using the PPM Workbench: What Users Need to Know

This section provides the information that users require to start the PPM Workbench on client machines. It also includes information on how to address JVM-related problems that can arise on client machines. For information on how to set up your Web browser to access the PPM Workbench, see the *Getting Started* guide.

For information on how to access this and other PPM Center documents, see *Accessing PPM Center Documentation* on page 22. For more information about the PPM Workbench, see the *Getting Started* guide.

Installing and Configuring the Java Plug-In on Client Machines

The `server.conf` contains one parameter that is associated with the Java plug-in. The `JAVA_PLUGIN_XPI_PATH` parameter specifies the Web location for downloading the cross-platform Java plug-in installer for Firefox browsers. The default setting for this parameter is `java.com/en/download/windows_xpi.jsp`.

For information about the Java plug-in supported for the current PPM Center version, see the *System Requirements and Compatibility Matrix*. For information about server configuration parameters, see Appendix A, *PPM Center Configuration Parameters*, on page 389.

Setting the Default Web Browser

To run the PPM Workbench interface as an application, users must specify the default browser setting in their user profiles.

To set the default browser setting:

1. From the shortcut bar in the PPM Workbench, select **Edit > User Profiles.**
2. On the **General** tab, in the **Default Browser** field, type the full path of the default Web browser.

If access to a URL is required, the PPM Workbench uses the default Web browser.

Starting the PPM Workbench on a Client Machine

To start the PPM Workbench from the PPM Center standard (HTML) interface:

- On the menu bar, select **Open > Administration > Open Workbench.**

➤ If a pop-up blocker is installed and enabled on the Web browser, the PPM Workbench cannot open. The user can configure the blocker to enable pop-ups from PPM Center.

Troubleshooting Default JVM Problems on Client Machines

If the Java plug-in sets itself as the default JVM for the browser, users can encounter the following problems in the PPM Workbench:

- The PPM Workbench displays a “class not found” exception error.
- Problems occur because other applications you are using require different versions of the Java plug-in.

To resolve these issues, make sure that an installed Java plug-in is not specified as the default.

To remove the default browser association to the Java plug-in:

1. Open the Windows control panel.
2. Double-click the **Java Plug-in** icon.

   The Java Plug-in Control Panel window opens.
3. Click the **About** tab.

   This tab lists the Java plug-in that PPM Center uses, as well as any other Java plug-ins installed.
4. Click the **Browser** tab.
5. Under **Settings,** deselect the checkbox (or checkboxes) for the installed browser (or browsers).

6. Click **Apply.**

   The Java Control Panel displays a message to indicate that you must restart the browser(s) to apply your changes.

After you make this change, other applications can use the Java plug-in version they require, and the PPM Workbench functions correctly.

If you encounter issues while downloading the PPM Workbench, try refreshing the Applet cache and deleting the temporary internet files.

What to Do Next

If you plan to perform any of the optional installations described in *Optional Installations* on page 90 (for example, if you plan to install an HP Deployment Management Extension), perform them now. If you have completed your installation tasks, test your system. As you do, be sure you understand the system maintenance tasks you must perform periodically. Those tasks are described in Chapter 7, *Maintaining the System,* on page 229.

5 Advanced System Configuration

Overview of Additional Installations and Advanced System Configuration

This chapter provides information about installations, integrations, and configurations in addition to the standard PPM Center setup. The following sections also include information about how to install optional products such as HP Deployment Management Extensions and the service packs that are delivered after the main PPM Center version.

For information on how to integrate the HP PPM Center product Web service component with external single sign-on systems, see Chapter 6, *Implementing User Authentication,* on page 203.

Configuring an External Web Server

The following sections provide information about how to configure an external Web server to work with a PPM Center Server cluster.

PPM Server can run with external Web servers such as Sun Java System Web Server, Microsoft IIS, Apache HTTP Server, HP-UX Apache-based Web Server, and IBM HTTP Server (IHS). For detailed information about which Web server versions PPM Center supports, and on which platforms, see the *System Requirements and Compatibility Matrix.*

(Windows) Using an External Web Server for Multiple Stand-Alone Instances of PPM Center

Because of Windows registry limitations, you cannot use just one Web server for multiple stand-alone PPM Center instances on a machine running Windows. Integration with an external Web server involves specifying the worker_file registry directive that points to the `workers.properties` file. The `workers.properties` file tells the redirector (`isapi_redirect.dll`) where to forward the request. Redirecting to two different instances does not work because each instance requires different workers properties. However, a single Windows registry points to only one `workers.properties` file.

If you must use an external Web server for multiple stand-alone PPM Server instances, HP recommends that you either use a UNIX machine to host the Web server, or use a hardware load balancer.

Overview of External Web Server Configuration

PPM Server can run with any of several external Web servers, including Sun Java System Web Server, Microsoft IIS, Apache HTTP Server, HP-UX Apache-based Web Server, and IBM HTTP Server (IHS).

For detailed information about which Web server versions PPM Center supports, and on which platforms, see the *System Requirements and Compatibility Matrix.*

To configure an external Web server, perform the following tasks:

1. Choose an external Web server.
2. Choose an external Web port.
3. Configure a `workers.properties` file.
4. (Microsoft IIS and Apache-based servers only) Configure a `uriworkermap.properties` file.
5. Configure the external Web server.
6. Integrate the external Web server with the PPM Server.
7. (Optional) Enable cookie logging on the external Web server.

The following sections provide details about these tasks.

Choosing an External Web Port

Choose the port through which the external Web server and the PPM Server(s) are to communicate. Select a port that is not in use on the machine running PPM Center. Later, you identify this port in the PPM Center `server.conf` file and your `workers.properties` file.

If you are integrating with an external Web server, you must set the `EXTERNAL_WEB_PORT` parameter on the PPM Server. This port number is then specified in the `workers.properties` file that is used by the jakarta 1 redirector.

Configuring the Workers Properties File

The `workers.properties` file stores information about the PPM Server(s), including the machine name, ports, and load balance setting. The external Web server uses this information to direct traffic to PPM Center applications, as required.

The following sections describe how to configure the `workers.properties` file for:

- Sun Java System Web Server
- Microsoft IIS
- Apache-based servers (Apache HTTP Server, HP-UX Apache-based Web Server, and IBM HTTP Server).

For information on the Web server versions supported, see the *System Requirements and Compatibility Matrix.*

Configuring the workers.properties File for a Single Server

The *Sample File* on page 139 shows the contents of a sample `workers.properties` file for a single-server configuration. Information that pertains to a clustered configuration is commented out.

As you edit the `workers.properties` file, keep the following two requirements in mind:

- The worker name must match the name of PPM Center instance defined for the `KINTANA_SERVER_NAME` parameter in the `server.conf` file.
- For Web servers such as Sun Java System Web Server, you must specify `connection_pool_size`, `connection_pool_minsize` and `connection_pool_timeout` (see comments in the following sample file).

Sample File

```
# JK 1.2.X configuration file. This file tells the external Web
# server how to connect to the PPM Servers.
# Place this file in the location you indicated in your Web
# server configuration.
# List of workers for handling incoming requests.
worker.list=load_balancer
# If "status" worker is defined (see below), then add it to the
# list of workers.
# worker.list=load_balancer,jkstatus
# Defines the PPM Server instances. The
# worker name is the value between the first and second period
# (server1, in this case). Copy this block for each additional
# server in the server cluster. Make sure the port number
# matches the port defined in the EXTERNAL_WEB_PORT parameter
# of the server.conf file, and that the worker name matches the
# PPM Center instance name defined by the
# KINTANA_SERVER_NAME parameter of the server.conf file. Please
# note that, for a server cluster setup, each HP PPM Center node
# has its own KINTANA_SERVER_NAME parameter.
worker.server1.host=localhost
worker.server1.port=8009
worker.server1.type=ajp13
worker.server1.lbfactor=1
# The following three parameters are required for
# Netscape-based Web servers such as Microsoft IIS.
# For Netscape-based Web servers, set the
# connection_pool_size equal to RqThrottle parameter in the Web
# server's magnus.conf file. Keep connection_pool_minsize at 1
# and connection_pool_timeout at 600. For Microsoft IIS, set
# the connection_pool_size parameter to 512 or higher, as
# necessary, to accomodate the load.
```

```
# HP recommends that you not use these parameters with
# Apache-based servers, including IBM HTTP Server, HP Web
# Server, and Apache itself.
#worker.server1.connection_pool_size=128
#worker.server1.connection_pool_minsize=1
#worker.server1.connection_pool_timeout=600
# Clustered configurations only.
# Defines a second PPM Server instance.#
# worker.server2.host=localhost
# worker.server2.port=8010
# worker.server2.type=ajp13
# worker.server2.lbfactor=1
#See comments above regarding setting the following three
# parameters.
#worker.server2.connection_pool_size=128
#worker.server2.connection_pool_minsize=1
#worker.server2.connection_pool_timeout=600
# Defines the load balancer. Be sure to list all servers in the
# PPM cluster in the balance_workers group.
worker.load_balancer.type=lb
worker.load_balancer.balance_workers=server1
# Optional. Define a special "status" worker. It enables
# monitoring of jk plugin status. If enabled, add it to the list
# of available workers (see above).
#worker.jkstatus.type=status
```

For more information about how to configure a server cluster, see *Configuring a Server Cluster* on page 172.

Configuring the workers.properties File

To configure a `workers.properties` file:

1. Navigate to the `<PPM_Home>/integration/webserverplugins/configuration` directory and open the `workers.properties` file in a text editor.
2. Set the `worker.list` parameter to `load_balancer`.
3. For the single server (or for each node in a cluster), configure the following values:
   a. Set `<Worker_Name>` to the name of PPM Center instance to which this worker connects. This is the name defined by the `KINTANA_SERVER_NAME` server configuration parameter in the `server.conf` file.

      In a clustered setup, each server has its own `KINTANA_SERVER_NAME` parameter.

   b. Set the `worker.server#.host` parameter to the network address of the machine on which PPM Center is installed.

      If the PPM Center instance runs on the same machine as the Web server, you can use `localhost`.

   c. Set the `worker.server#.port` parameter to the external Web port (`EXTERNAL_WEB_PORT` parameter) to use.
   d. Set the `worker.server#.type` parameter to `ajp13`, which is the protocol used to connect to the remote server.
   e. Set the `worker.server#.lbfactor` parameter to the load balancing factor used to distribute load to the PPM Servers.

      If all servers can handle approximately the same load, assign “1” to each server. If a server can handle twice as much load as another server, assign “2” to that more robust server and “1” to the other server.
4. Set the `worker.load_balancer.type` parameter to `lb`.

5. Set the `worker.load_balancer.balance_workers` parameter to a comma-delimited list of all servers in the cluster (as configured in step 3).

   Example:

   `worker.load_balancer.balance_workers=worker1,worker2,worker3`

6. (Optional) To enable the JK status page, add a worker of special type "status" (`worker.jkstatus.type=status`), and then add this worker to the list of workers (`worker.list`).

Configuring the uriworkermap.properties File on Microsoft IIS and Apache-Based Servers

The `uriworkermap.properties` file is used to specify mappings between a given URL (or URL pattern) and worker name. The following shows the contents of a sample `uriworkermap.properties` file.

```
# /itg/* must be mapped to one of the workers
/itg/*=load_balancer
/dashboard/*=load_balancer
# You can access the JK status page at
# http://web_server_host:web_server_port/jkmanager.
# If you want to enable the JK status page, uncomment the
# following line.
#/jkmanager=jkstatus
```

Each line of `uriworkermap.properties` file represent a single mapping in the format *<URL_Pattern> = <Worker_Name>*. If the Web server processes a URL that matches *<URL_Pattern>*, then *<Worker_Name>* is used to serve this request. *<Worker_Name>* must be defined in the `workers.properties` file.

Configuring the External Web Server

This section provides information about how to set up the following external PPM Center-supported Web servers:

- Sun Java System Web Server
- Microsoft IIS
- Apache HTTP Server
- HP-UX Apache-based Web Server
- IBM HTTP Server

For a list of supported versions, see the *System Requirements and Compatibility Matrix* document.

Configuring the Sun Java System Web Server

To configure the Sun Java System Web Server to run as the external Web server for the PPM Server:

1. Connect to the Sun Java System administration server and create a new server named "PPM."

   This creates the `https-PPM` directory. The `https-PPM` directory contains two files: `magnus.conf` and `obj.conf`.

2. Stop the PPM Server. (For instructions, see *Starting and Stopping the PPM Server* on page 95.)

3. Place the configured `workers.properties` file (see *Configuring the Workers Properties File* on page 138) in the `<Sun_Home>/https-<Web_Server_Name>/config` directory.

4. Copy the `nsapi_redirector.so` plug-in to any directory on the machine that runs the Sun Java System Web Server.

   The Web server must have permissions to read and execute this file.

5. Add the following two lines to the `magnus.conf` file (the text can wrap, but each "`init fn=`" must be a continuous line with no spaces):

```
Init fn="load-modules" shlib="<Path_To_NSAPI_Redirector>/
nsapi_redirector.so" funcs="jk_init,jk_service"

Init fn="jk_init" worker_file="<Sun_Home>/https-<Web_Server_
Name>/config/workers.properties" log_level="error"
log_file=<Path_To_Log_Files>/ppm_server.log
```

6. If you are using the 64-bit version of Sun Java System Web Server, do the following:

   a. Set the LD_LIBRARY_PATH_64 environment variable to `/usr/local/lib/sparcv9`.

   b. Navigate to the `/webserver/config` directory, open the `magnus.conf` file, and then add the following lines to the file:

```
Init fn="load-modules" shlib="/sun/webserver7/
https-https-ppm1/config/nsapi_redirector.so" funcs=
"jk_init,jk_service"

Init fn="jk_init" worker_file="<Sun_Home>/https-
<Web_Server_Name>/config/workers.properties"
log_level="debug" log_file="/sun/webserver7/
https-https-ppm1/logs/itg_server.log" shm_file="/sun/
webserver7/https-https-ppm1/logs/jk_shm"
```

7. In the `obj.conf` file, do the following:

   a. Add the following line at the beginning of the "Object" section (that is, after `<object name=default>`).

```
NameTrans fn="assign-name" from="/itg/*" name=
<PPM_Servlet>
NameTrans fn="assign-name" from="/dashboard/*" name=
<PPM_Servlet>
```

   b. Place the following text after the `</Object>` section:

```
<Object name="ppm_servlet">
Service fn="jk_service" worker=<Load_Balancer>
</Object>
```

The `<PPM_Servlet>` strings must match.

The `worker` attribute specifies the name of the JK worker used to serve requests with URLs that match the `path` attribute, which is `/itg/*` in this case.

Check the start and end of each line in the `magnus.conf` and `obj.conf` files to make sure that there are no extra spaces in either of these files.

8. Enable content compression.

   For information on how to enable dynamic content compression, see *Enabling Dynamic Compression On an External Web Server* on page 165.

(Optional) Enabling Cookie Logging on the Sun Java System Web Server

To enable cookie logging:

1. Stop the Sun Java System Web Server.
2. In the `magnus.conf` file, find the line that initializes flex. The line begins with the following text.

   ```
   Init fn=flex-init
   ```

3. Append the following string to the end of this line:

   ```
   %Req->headers.cookie.JSESSIONID%
   ```

   The resulting modified line is:

   ```
   Init fn=flex-init access="$accesslog" format.access=
   "%Ses->client.ip% - %Req->vars.auth-user%[%SYSDATE%]
   \"%Req->reqpb.clf-request%\" %Req->srvhdrs.clf-status%
   %Req->srvhdrs.content-length%"
   JSESSIONID=%Req->headers.cookie.JSESSIONID%
   ```

4. Restart the Web server.

Configuring the Microsoft Internet Information Services 6.0 Web Server on Windows Server 2003 Systems

To configure the Microsoft Internet Information Services (IIS) 6.0 Web server on a Windows Server 2003 system:

1. Create a virtual directory named `jakarta` that points to the IIS scripts directory, as follows:

   a. Select **Start > Control Panel > Administrative Tools > Internet Information Services (IIS) Manager.**

   b. Copy the `workers.properties` file, the `uriworkermap.properties` file, and `<PPM_Home>/integration/webserverplugins/iis/windows/x86-32/isapi_redirect.dll` file either to the `c:\inetpub\Adminscripts` directory or to the `c:\inetpub\Wwwroot` directory.

   c. In the IIS Services Manager, right-click the Web site you plan to integrate with the PPM Server, and then select **New > Virtual Directory** from the shortcut menu.

   d. On the first page of the Virtual Directory Creation Wizard, click **Next.**

   e. On the Virtual Directory Alias page, under **Alias,** type **`jakarta`.**

   f. Click **Next.**

   g. On the Web Site Content Directory page, under **Directory,** type the full path to the `isapi_redirect.dll` file (`c:\inetpub\Adminscripts\isapi_redirect.dll` or `c:\inetpub\Wwwroot\isapi_redirect.dll`).

      Depending on the IIS root directory configuration, the drive and directory may vary. This directory must have run permission.

   h. On the Access Permission page, select **Read, Run scripts (such as ASP),** and **Execute (such as ISAPI application or cgi).**

2. Change directory security to set up anonymous access, as follows:

   Anonymous access enables anyone to visit the public areas of your Web site but prevents unauthorized users from gaining access to your Web server's critical administrative features and restricted information.

   If you do not change directory security, you may encounter an "Access Denied" error message.

   a. From the Control Panel, open **Administrator Tools,** and then open the Windows Internet Information Services Manager.

   b. In IIS Manager, click **Web Sites,** right-click the Web site instance you are integrating with PPM Server, and then select **Properties** from the shortcut menu.

   c. Click the **Directory Security** tab.

   d. In the **Authentication and Access Control** section, click **Edit.**

   e. Select the **Enable anonymous access** check box, and then browse for IUSR_<HOST>.

   f. Clear everything selected in the **Authenticated access** section.

3. Configure a `workers.properties` file and a `uriworkermap.properties` file, as described in *Configuring the Workers Properties File* on page 138 and *Configuring the uriworkermap.properties File on Microsoft IIS and Apache-Based Servers* on page 142.

4. Configure IIS to load `isapi_redirect.dll` as a filter, as follows:

   a. To define registry values for IIS with Apache Jakarta Tomcat Connector (JK):

      i. Add the following registry key.

      ```
      HKEY_LOCAL_MACHINE\SOFTWARE\Apache Software
      Foundation\Jakarta Isapi Redirector\1.0
      ```

      ii. Add a string value with the name `extension_uri` and set `/jakarta/isapi_redirect.dll` as its value.

      The `extension_uri` key in the Windows registry must point to the full path of the `isapi_redirect.dll` file.

iii. Add a string value with the name `worker_file` and set its value as the full path to the `workers.properties` file. That is, `<`*`ISAPI_Redirector_Home`*`>\workers.properties`

Example

`c:\inetpub\Adminscripts\workers.properties`

iv. Add a string value with the name `log_level` and set `ERROR` as its value.

For more verbose logging, use `DEBUG` or `INFO`.

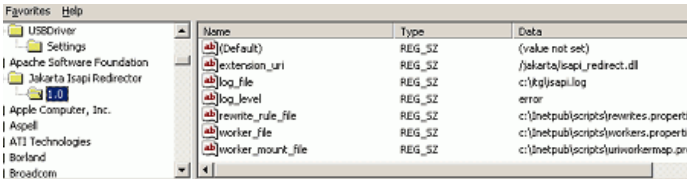
v. Add a string value with the name `log_file` and specify the directory in which you want the log file to reside. (Include the log file name. For example, `c:\inetpub\Adminscripts\isapi.log`.)

vi. Add a string value with the name `worker_mount_file` and a value that is the full path to your `uriworkermap.properties` file (`c:\inetpub\Adminscripts\uriworkermap.properties` or `c:\inetpub\Wwwroot\uriworkermap.properties`).

vii. Create an empty file named `rewrites.properties`, and save it either to the `c:\inetpub\Adminscripts` directory or to the `c:\inetpub\Wwwroot` directory. Add a string value with the name `rewrite_rule_file` and assign it a value that is the full path to the `rewrites.properties` file (`c:\inetpub\Adminscripts\rewrites.properties` or `c:\inetpub\Wwwroot\rewrites.properties`).

The previous step is required as the result of a known issue in JK 1.2.18. For detailed information about this issue, go to the Web site issues.apache.org/bugzilla/show_bug.cgi?id=40384.

The following figure shows a correctly configured registry.



b. Restart your Windows system.

c. Select **Start > Control Panel > Administrative Tools > Internet Information Services Manager.**

Perform the following steps at the Web-sites level.

d. Right-click the Web site name, and then click **Properties** from the shortcut menu.

   The Properties dialog box opens.

e. Click the **ISAPI Filter** tab.

f. Click **Add.**

   The Filter Properties window opens.

g. In the **Filter Name** field, type **`jakarta.`**

h. In the **Executable** field, type the full path to the `isapi_redirect.dll` file (`c:\inetpub\Adminscripts\isapi_redirect.dll` or `c:\inetpub\Wwwroot\isapi_redirect.dll`).

5. Enable Tomcat's redirector DLL in Web service extensions, as follows:

   a. In the Windows management console, click **Web Services Extensions.**

   b. Select **Add a new Web service extension.**

   c. Type the extension name.

      Example

      `Jakarta-Tomcat`

   d. Type the path to the `isapi_redirect.dll` file (`c:\inetpub\Adminscripts\isapi_redirect.dll` or `c:\inetpub\Wwwroot\isapi_redirect.dll`).

   e. Select **Set extension status to Allowed.**

   f. Click **Add.**

6. Restart the IIS service. (Restarting the Web site is not enough. You must restart World Wide Web Publishing Service from the Services management console.)

   After you restart the IIS service, the ISAPI filter does not load immediately. The IIS service may require a few minutes to establish a connection with PPM Center. Before the connection is established, your browser may display the error message “HTTP Error 404 - File or directory not found. Internet Information Services (IIS)”.

7. Start the PPM Server(s).

8. Enable dynamic compression of assets. For instructions, see *Enabling Dynamic Compression On an External Web Server* on page 165.

(Optional) Enabling Cookie Logging on Microsoft IIS 6.0

To enable cookie logging on IIS 6.0:

1. Open IIS.
2. Select a Web or FTP site and open its property sheets.
3. Select **Enable Logging.**
4. Click **Properties.**
5. On the **Extended Properties** page, select **Cookies.**
6. Click **Apply.**

Configuring the Microsoft Internet Information Services 7.0 Web Server on a Windows Server 2008 System

Because running IIS 7.0 in 64-bit mode is not supported on Windows 2008 systems, you must configure it to run in 32-bit mode on Windows 2008.

To configure the IIS 7.0 Web server to run in 32-bit mode on a Windows Server 2008 system (64-bit machine):

1. Stop the IIS 7.0 Web Server.
2. Open Server Manager on the Microsoft IIS Web server host and make sure that the ISAPI Filters and ISAPI Extensions role services are installed.

   If you must install the ISAPI Filters and ISAPI Extensions, make sure that you restart after you install these services.

3. Start the IIS Manager.
4. In the **Advanced Settings** section for the Default Application Pools, set **Enable 32 bit Application** to **True.**
5. Make sure that you configure the Windows registry for the Tomcat connector (isapi_redirect) plug-in only under the branch Wow6432Node branch, as follows:

   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Apache Software
   Foundation\Jakarta Isapi Redirector\1.0
   ```

6. Create a virtual directory named `jakarta` that points to the IIS scripts directory, as follows:

   a. Select **Start > Control Panel > Administrative Tools > Internet Information Services (IIS) Manager.**

   b. Under your IIS Web server, create a new (or identify an existing) Web site to integrate with the PPM Server.

c. In your file system, create a new (or select an existing) directory in which to store integration-related files. In this procedure, this directory is referred to as `<ISAPI_REDIRECTOR_HOME>`.

The `<ISAPI_REDIRECTOR_HOME>` directory must have run permission.

If you choose to use the Windows Registry instead of using `isapi_redirect.properties`, then you *must* use the branch Wow6432Node for running IIS 7.0 on Windows 2008.

d. Copy the `workers.properties` file, `uriworkermap.properties` file and `<PPM_Home>/integration/webserverplugins/iis/windows/x86-32/isapi_redirect.dll` file to the `<ISAPI_REDIRECTOR_HOME>` directory you created (or selected) in step c.

e. Right-click the Web site you created (or identified) in step b, and then select **Add Virtual Directory** from the shortcut menu.

f. In the first Add Virtual Directory window, do the following:

i. In the **Alias** box, type the alias name (for example, **`Jakarta`**).

ii. Use the **Physical path** multiselect to navigate to and select the `<ISAPI_REDIRECTOR_HOME>` directory path.

An example of this directory is `c:\inetpub\scripts`. The drive and directory depend on the IIS root directory configuration. This directory must have run permission.

7. If you are using NTLM for user authentication, then do the following:

a. Start the Internet Information Services (IIS) Manager and access the Error Pages feature.

b. Right-click error 500, and then select **Edit Feature Settings** from the shortcut menu.

c. Set Error Responses to **Detailed errors.**

8. Configure both a `workers.properties` file and a `uriworkermap.properties` file.

9. Configure IIS to load `isapi_redirect.dll` as a filter, as follows:

   a. To define registry values for IIS with Apache Jakarta Tomcat Connector (JK):

      i. Add the following registry key:

      ```
      HKEY_LOCAL_MACHINE\SOFTWARE\Apache Software
      Foundation\Jakarta Isapi Redirector\1.0
      ```

      ii. Add a string named `extension_uri` and set its value to `/jakarta/isapi_redirect.dll`.

      iii. Add a string named `worker_file` and set its value to the full directory path for the `workers.properties` file. That is, *<ISAPI_REDIRECTOR_HOME>*`\workers.properties`.

      Example:

      ```
      C:\inetpub\scripts\workers.properties
      ```

      iv. Add a string with the name `log_level` and set its value to `ERROR`. (For more verbose logging, options include `DEBUG` and `INFO`.)

      v. Add a string with the name `log_file` and set its value to the directory in which you want to save your log file. Include the log file name in the directory path (for example, `C:\PPM\isapi.log`).

      vi. Add a string named `worker_mount_file` and set its value to the full directory path for the `uriworkermap.properties` file. That is, *<ISAPI_REDIRECTOR_HOME>*`\uriworkermap.properties`.

      Example:

      ```
      C:\inetpub\scripts\uriworkermap.properties
      ```

      vii. Create a file named `rewrites.properties` in the *<ISAPI_REDIRECTOR_HOME>* directory

viii. Add a string named `rewrite_rule_file` to the `rewrites.properties` file and set its value to the full directory path for the new `rewrites.properties` file. That is, `<`*`ISAPI_REDIRECTOR_HOME`*`>\rewrites.properties.`

Example:

```
C:\inetpub\scripts\rewrites.properties
```

b. Open **Internet Information Services (IIS) Manager,** and then do the following:

i. Select the name of the Web site you created (or identified) in step b on page 156, and then, in the center panel, double-click **ISAPI Filters.**

ii. In the **Actions** pane, click **Add.**

The Add ISAPI Filter dialog box opens.

iii. In the **Filter name** box, type a name for the ISAPI filter.

iv. In the **Executable** box, click the ellipsis button (...), and then navigate to and select the folder that contains the `isapi_redirect.dll` file.

The **ISAPI Filters** list displays the filter name.

10. Enable the Tomcat redirector DLL in Web service extensions, as follows:

a. From the **Connections** panel of Internet Information Services (IIS) Manager, select the Web server name.

b. In **Features View,** double-click **ISAPI and CGI Restrictions.**

The ISAPI and CGI Restrictions window opens.

c. In the **Actions** pane, click **Add.**

The Add ISAPI or CGI Restriction dialog box opens.

d. Do the following:

   i. In the **ISAPI or CGI path** text box, provide the full directory path for the `isapi_redirect.dll` file.

   ii. In the **Description** text box, type a short description of the restriction.

   iii. Select the **Allow extension path to execute** check box.

e. In the **ISAPI and CGI Restrictions** list, select the restriction you added.

f. In the **Actions** pane, click **Allow.**

11. Enable execution of ISAPI filter, as follows:

    a. From the **Connections** panel of Internet Information Services (IIS) Manager, select the Web site (see step b on page 156).

    b. In **Features View,** double-click **Handler Mappings.**

       The Handler Mappings window opens.

    c. Right-click the ISAPI DLL item, and then select **Edit Feature Permissions** from the shortcut menu.

       The Edit Feature Permissions window opens.

    d. Select the **Read, Script,** and **Execute** check boxes.

12. Restart the IIS service. (Restarting the Web site is not enough. You must restart World Wide Web Publishing Service from the Services management console.)

    After you restart the IIS service, the ISAPI filter does not load immediately. The IIS service may require a few minutes to establish a connection with PPM Center. Before the connection is established, your browser may display the error message “HTTP Error 404 - File or directory not found. Internet Information Services (IIS)”.

13. Start the PPM Server(s).

    If your PPM Center instance includes multiple nodes in a cluster configuration, you must start these nodes one at a time. Make sure that you wait until each node is fully started before you start the next node.

Configuring the Microsoft Internet Information Services 7.5 Web Server on a Windows Server 2008 System

To configure the IIS 7.5 Web server on a Windows Server 2008 system:

1. Open Server Manager on the Microsoft IIS Web server host and make sure that the ISAPI Filters and ISAPI Extensions role services are installed.

   If you must install the ISAPI Filters and ISAPI Extensions, make sure that you restart after you install these services.

2. Create a virtual directory named `jakarta` that points to the IIS scripts directory, as follows:

   a. Select **Start > Control Panel > Administrative Tools > Internet Information Services (IIS) Manager.**

   b. Under your IIS Web server, create a new (or identify an existing) Web site to integrate with the PPM Server.

   c. In your file system, create a new (or select an existing) directory in which to store integration-related files. In this procedure, this directory is referred to as *<ISAPI_REDIRECTOR_HOME>*.

   The *<ISAPI_REDIRECTOR_HOME>* directory must have run permission.

   d. Copy the `workers.properties` file, `uriworkermap.properties` file and *<PPM_Home>*`/integration/webserverplugins/iis/windows/x86-64/isapi_redirect.dll` file to the *<ISAPI_REDIRECTOR_HOME>* directory you created (or selected) in step c.

   e. Right-click the Web site you created (or identified) in step b, and then select **Add Virtual Directory** from the shortcut menu.

f. In the first Add Virtual Directory window, do the following:

   i. In the **Alias** box, type the alias name (for example, **`Jakarta`**).

   ii. Use the **Physical path** multiselect to navigate to and select the *`<ISAPI_REDIRECTOR_HOME>`* directory path.

   An example of this directory is `c:\inetpub\scripts.` The drive and directory depend on the IIS root directory configuration. This directory must have run permission.

3. If you are using NTLM for user authentication, then do the following:

   a. Start the Internet Information Services (IIS) Manager and access the Error Pages feature.

   b. Right-click error 500, and then select **Edit Feature Settings** from the shortcut menu.

   c. Set Error Responses to **Detailed errors.**

4. Configure both a `workers.properties` file and a `uriworkermap.properties` file.

5. Configure IIS to load `isapi_redirect.dll` as a filter, as follows:

   a. To define registry values for IIS with Apache Jakarta Tomcat Connector (JK):

      i. Add the following registry key:

      ```
      HKEY_LOCAL_MACHINE\SOFTWARE\Apache Software
      Foundation\Jakarta Isapi Redirector\1.0
      ```

      ii. Add a string named `extension_uri` and set its value to `/jakarta/isapi_redirect.dll.`

      iii. Add a string named `worker_file` and set its value to the full directory path for the `workers.properties` file. That is, *`<ISAPI_REDIRECTOR_HOME>`*`\workers.properties.`

      Example:

      `C:\inetpub\scripts\workers.properties`

iv. Add a string with the name `log_level` and set its value to `ERROR`. (For more verbose logging, options include `DEBUG` and `INFO`.)

v. Add a string with the name `log_file` and set its value to the directory in which you want to save your log file. Include the log file name in the directory path (for example, `C:\PPM\isapi.log`).

vi. Add a string named `worker_mount_file` and set its value to the full directory path for the `uriworkermap.properties` file. That is, `<ISAPI_REDIRECTOR_HOME>\uriworkermap.properties`.

Example:

```
C:\inetpub\scripts\uriworkermap.properties
```

vii. Create a file named `rewrites.properties` in the `<ISAPI_REDIRECTOR_HOME>` directory

viii. Add a string named `rewrite_rule_file` to the `rewrites.properties` file and set its value to the full directory path for the new `rewrites.properties` file. That is, `<ISAPI_REDIRECTOR_HOME>\rewrites.properties`.

Example:

```
C:\inetpub\scripts\rew rites.properties
```

b. Open **Internet Information Services (IIS) Manager,** and then do the following:

i. Select the name of the Web site you created (or identified) in step b on page 156, and then, in the center panel, double-click **ISAPI Filters.**

ii. In the **Actions** pane, click **Add.**

The Add ISAPI Filter dialog box opens.

iii. In the **Filter name** box, type a name for the ISAPI filter.

iv. In the **Executable** box, click the ellipsis button (...), and then navigate to and select the folder that contains the `isapi_redirect.dll` file.

The **ISAPI Filters** list displays the filter name.

6. Enable the Tomcat redirector DLL in Web service extensions, as follows:

   a. From the **Connections** panel of Internet Information Services (IIS) Manager, select the Web server name.

   b. In **Features View,** double-click **ISAPI and CGI Restrictions.**

      The ISAPI and CGI Restrictions window opens.

   c. In the **Actions** pane, click **Add.**

      The Add ISAPI or CGI Restriction dialog box opens.

   d. Do the following:

      i. In the **ISAPI or CGI path** text box, provide the full directory path for the `isapi_redirect.dll` file.

      ii. In the **Description** text box, type a short description of the restriction.

      iii. Select the **Allow extension path to execute** check box.

   e. In the **ISAPI and CGI Restrictions** list, select the restriction you added.

   f. In the **Actions** pane, click **Allow.**

7. Enable execution of ISAPI filter, as follows:

   a. From the **Connections** panel of Internet Information Services (IIS) Manager, select the Web site (see step b on page 156).

   b. In **Features View,** double-click **Handler Mappings.**

      The Handler Mappings window opens.

   c. Right-click the ISAPI DLL item, and then select **Edit Feature Permissions** from the shortcut menu.

      The Edit Feature Permissions window opens.

   d. Select the **Read, Script,** and **Execute** check boxes.

8. Restart the IIS service.(Restarting the Web site is not enough. You must restart World Wide Web Publishing Service from the Services management console.)

After you restart the IIS service, the ISAPI filter does not load immediately. The IIS service may require a few minutes to establish a connection with PPM Center. Before the connection is established, your browser may display the error message “HTTP Error 404 - File or directory not found. Internet Information Services (IIS)”.

9. Start the PPM Server(s).

If your PPM Center instance includes multiple nodes in a cluster configuration, you must start these nodes one at a time. Make sure that you wait until each node is fully started before you start the next node.

Configuring an Apache-Based Web Server

This section provides the steps you use to:

- Compile a binary of JK. (Do this if, and only if, a precompiled binary does not work on your system.)
- Configure Apache HTTP Server

This information applies to Apache HTTP Server, HP-UX Apache-based Web Server, and IBM HTTP Server.

Compiling a Binary JK Module

Configuring an Apache-based Web server on UNIX requires a dynamically linkable JK module binary named `mod_jk.so`. In most cases, the `<PPM_Home>/integration/webserverplugins/<Web_Server_Name>` directory contains precompiled binaries of JK for several operating systems.

Before you try to compile the JK module, check this directory to determine whether it contains the binaries required for your system. Select the mod_jk plug-in based on your operating system, Web server and CPU type (32 or 64 bit).

If a precompiled binary is unavailable, perform the following steps.

To compile a binary JK module:

1. Download and unpack a source code bundle from the following Web site.

   tomcat.apache.org/connectors-doc/index.html

2. Change to the following directory:

   `tomcat-connectors-<Version>-src/native`

3. Run the configuration script, as follows:

   ```
   ./configure --with-apxs=/<Path_To_Apache_Bin>/apxs
   ```

   The configuration script generates the `make` files for the current machine environment. The `make` files are required to run the `make` command, as described in the next step.

4. Run the `make` command to build the Apache module that forwards requests from the Apache HTTP Server to the PPM Server using the AJP13 protocol.

For more details on how to recompile the connector, go to the following Web site.
tomcat.apache.org/connectors-doc/index.html

Enabling Content Compression

Configuring an Apache-based Web server also involves enabling dynamic content compression. For information on how to enable content compression, see *Enabling Dynamic Compression On an External Web Server* on page 165.

(Optional) Enabling Cookie Logging on Apache 2.0

To enable cookie logging on Apache 2.0:

1. Open the Apache `httpd.conf` file in a text editor. The `httpd.conf` file is located in `<IHS_Home>/conf` (where `<IHS_Home>` is the IBM HTTP Server installation directory).

2. Find the line of text that begins with the following string.

   ```
   LogFormat "%h %l %u %t \"%r\"%>s %b
   ```

3. After "`%b,`" type the following:

   `%{Cookie}i"`

   The log format and custom log lines now look as follows.

   ```
   LogFormat "%h %l %u %t \"%r\"%>s %b %{Cookie}i" common
   CustomLog logs/access_log common
   ```

4. Save and close the `httpd.conf` file.

For information about how to configure an Apache-based Web server to work with a PPM Server cluster, see the *System Administration Guide and Reference* for PPM Center.

Configuring IBM HTTP Server Version 6.1.0

This section provides the procedure for configuring IBM HTTP Server (IHS) to work with PPM Center.

To configure IHS for PPM Center:

1. Navigate to `<PPM_Home>/integration/webserverplugins/ibmihs/aix/powerpc-32/mod_jk.so`, and then copy `mod_jk.so` to the IHS module directory (usually `<IHS_Home>/modules`).

   The Jakarta Tomcat Connector (`mod_jk`) is used to connect IHS and PPM Center. The Ajp13 protocol keeps an open socket and controls the communications between PPM Center (its built-in Tomcat component) and IHS.

2. Instruct IHS to load the Jakarta Tomcat Connector (`mod_jk`). You can use the IHS LoadModule configuration directives in the `httpd.conf` file, which is located in `<IHS_Home>/conf` (where `<IHS_Home>` is the IHS installation directory).

3. Add the following lines of text to the `httpd.conf` file:

```
LoadModule jk_module <Relative_Modules_Path>/mod_jk.so
JkWorkersFile <Relative_Conf_Path>/workers.properties
JkMountFile <Relative_Conf_Path>/uriworkermap.properties
JkLogFile <Relative_Logs_Path>/jk.log
JkMountCopy On
JKLogLevel ERROR
```

   If you plan to enable SSL on IHS, then you must also add the “JkMountCopy On” virtual host directive.

4. Navigate to the `<PPM_Home>/integration/webserverplugins/configuration` directory, and then copy the `workers.properties` and `uriworkermap.properties` to the IHS configuration directory (usually `<IHS_Home>/conf`).

5. Configure the `workers.properties` file. (For detailed information and instructions, see *Configuring the Workers Properties File* on page 138.)

6. Configuring the `uriworkermap.properties` file to specify mappings between a given URL (or URL pattern) and worker name. (For detailed information and instructions, see *Configuring the uriworkermap.properties File on Microsoft IIS and Apache-Based Servers* on page 142

   Make sure that the name of the worker mapped to `/itg/*` pattern in the `uriworkermap.properties` file matches the name of the worker defined in the `workers.properties` file. This worker must also be listed in the `worker.list` directive of the `workers.properties` file.

7. Restart your IBM HTTP Server to see whether your configuration works.

Enabling Secure Sockets Layer on an External Web Server

To enable Secure Sockets Layer (SSL) on the Web server:

1. Generate a certificate signing request (CSR) for the server on which you plan to install the SSL certificate.

   To do this, use the software that your external Web server provides. If you do not know what software your server uses, contact the Web server vendor for that information.

2. Submit the CSR to a certificate authority (such as VeriSign).

   It may take several days for the certificate authority to validate the company.

3. After you obtain the SSL certificate, install it on your Web server.

4. Contact your Web server administrator or Web server vendor to help you enable SSL on the Web server.

5. If your external Web server or hardware load balancer uses SSL, open the `server.conf` file and change the server configuration parameter `BASE_URL` to `https://<Web_Server>`.

   By default, the HTTPS typically runs on port 443 on the Web server. If you use a port other than 443, you must specify the port number in the `BASE_URL` (`https://<Web_Server>:<Web_Server_Port>`).

6. Restart the Web server.

If you enable SSL on IBM HTTP Web Server, the “JKMountCopy On” virtual host directive must be included in the `httpd.conf` file. For more information, see *Configuring IBM HTTP Server Version 6.1.0* on page 163.

Enabling Dynamic Compression On an External Web Server

Wide area networks (WANs) often have both low bandwidth and high latency (delays in network data processing), which significantly degrade network performance. Users who access applications over a WAN experience poorer response times than users who access the same applications through a local area network (LAN).

HP Project and Portfolio Management Center leverages application content compression to minimize the performance overhead imposed by operating in a WAN environment. Rather than compress content within the application code, PPM Center uses the compression capabilities of both the Tomcat Web container that runs in JBoss and the compression capabilities in third-party Web servers (Microsoft Internet Information Services 6.0, Apache-based Web server, or Sun Java System Web Server).

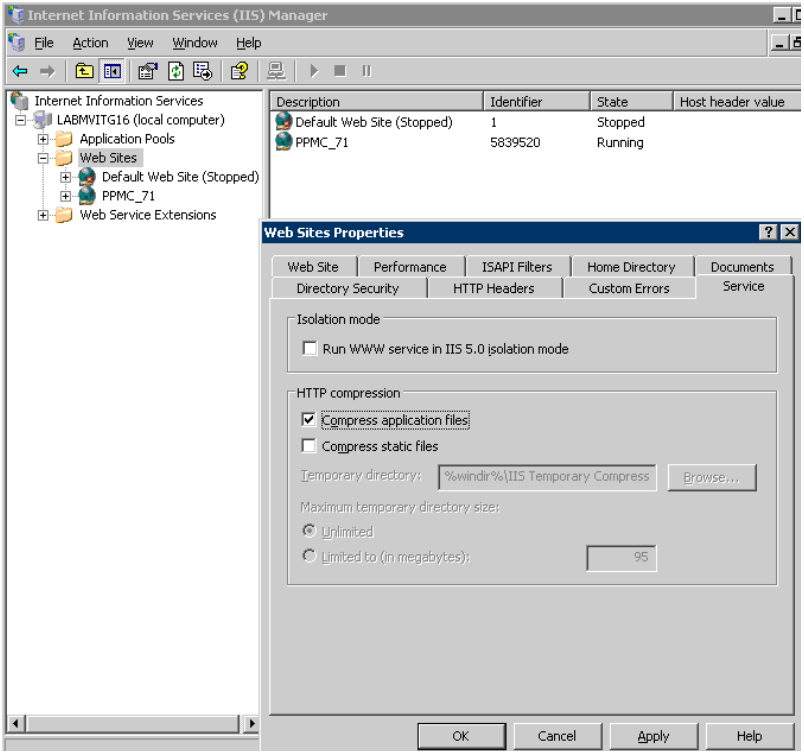
If you deploy PPM Center without an external Web server, the application content is compressed by default, and no additional configuration is required. If, however, you deploy an external Web server as the Web tier, then you must enable compression for that Web server. Otherwise, application content is delivered uncompressed, which results in poor response times for users over the WAN.

Enabling Dynamic Content Compression on Microsoft Internet Information Services 6.x

To enable HTTP compression through the Microsoft Internet Information Services (IIS) 6.x Manager console:

1. Open the Internet Information Services (IIS) Manager window.
2. Under the name of the computer that hosts the IIS site, right-click the **Web Sites** folder, and then select **Properties** on the shortcut menu.

   The Web Sites Properties window opens.
3. On the **Service** tab, select the **Compress application files** check box.



4. From either the services Control Panel or the command line, stop and then restart the World Wide Web Publishing Service.

If you need to enable compression for a subset of the Web sites hosted by the server, see *Enabling HTTP Compression (IIS 6.0)*, which is available on the Microsoft TechNet Web site.

Enabling Dynamic Content Compression on Apache-Based Web Servers

This section provides information on how to enable dynamic compression on an Apache-based Web server that either has been compiled with the `mod_deflate` module enabled or that can load the dynamic module. Apache Web server installation documentation provides instructions on how to enable modules within the application server. If `mod_deflate` is not loaded in Apache, the following steps cannot enable content compression.

To enable compression of dynamic assets on an Apache-based Web server:

1. Navigate to the `<Apache_Home>/conf` directory and open the `httpd.conf` file in a text editor.
2. Add the following to the `httpd.conf` file.

```
# gzip config begin
LoadModule deflate_module modules/mod_deflate.so
<Location/itg>
SetOutputFilter DEFLATE
BrowserMatch ^Mozilla/4 gzip-only-text/html
BrowserMatch ^Mozilla/4\.0[678] no-gzip
BrowserMatch \bMSI[E] !no-gzip !gzip-only-text/html
SetEnvIfNoCase Request_URI \
   \.(?:gif|jpe?g|png)$ no-gzip dont-vary
Header append Vary User-Agent env=!dont-vary
</Location>
# gzip config end
```

3. Save and close the `httpd.conf` file.

Enabling Dynamic Content Compression on Sun Java System Web Server

To enable compression of dynamic assets on Sun Java System Web Server:

1. On the machine running the Sun Java System Web Server, navigate to the `<Sun_Home>/https-<Web_Server_Name>/config` directory, and open the `obj.conf` file.

   During the initial Sun Java System Web Server configuration, installation of jk_service required that the following text be added to the `obj.conf` file (after `</Object>`).

   ```
   <Object name="ppm_servlet">
   Service fn="jk_service" worker=<Load_Balancer>
   </Object>
   ```

2. Modify that text, as follows.

   ```
   <Object name="ppm_servlet">
   Service fn="jk_service" worker=<Load_Balancer>
   Output fn="insert-filter" filter="http-compression"
   vary="off" compression-level="6"
   </Object>
   ```

Integrating an External Web Server with a PPM Server

To integrate an external Web server with the PPM Server, perform the following tasks:

1. Stop the PPM Server.

   For information about how to do this, see *Starting and Stopping the PPM Server* on page 95.

2. Set the server configuration parameter values.

3. Validate the integration.

The following sections provide the steps you use to set the `server.conf` parameters and verify the integration.

Setting the External Web Port

To set the external Web port:

1. Back up the `<PPM_Home>/server.conf` file.
2. Open the `server.conf` file in a text editor.
3. Add `com.kintana.core.server.EXTERNAL_WEB_PORT,` and set it to the port number in the `workers.properties` file.
4. Change `BASE_URL` to the base URL of the external Web server.

   If your external Web server or hardware load balancer uses Secure Sockets Layer (SSL), you must change the BASE_URL parameter value to `https://<Web_Server>`.

   By default, the HTTPS runs on port 443 on the Web server. If you use a port other than 443, you must specify that port number in the BASE_URL (`https://<Web_Server>:<Web_Server_Port>`).

5. Save and close the `server.conf` file.
6. Run the `kUpdateHtml.sh` script.

For more information about the `BASE_URL` parameter, see Appendix A, *PPM Center Configuration Parameters*, on page 389. For more information about the `kUpdateHtml.sh` script, see *kUpdateHtml.sh* on page 505.

Verifying the Integration

To verify the integration between the external Web server and the PPM Server:

1. Start the external Web server and check for errors.
2. Start the PPM Server and check for errors.
3. In a supported browser, open the page `<Host>:<Port>/itg/dashboard/app/portal/PageView.jsp`. (You must use the complete path. Specifying only `<Host>:<Port>/itg` does not work.)

   For information about how to start the PPM Server, see *Starting and Stopping the PPM Server* on page 95. For information about supported browsers, see the *System Requirements and Compatibility Matrix.*

Troubleshooting External Web Server Configuration

If HTTP errors occur when you try to log on to PPM Center, do the following:

1. Gather `DEBUG` level logging information for the Web server.
2. Gather `DEBUG` level mod_jk logs.
3. Check to make sure that Tomcat connector the plug-in is being loaded:
   - Review the plug-in log file.
   - Review the operating system logs (for example, Event Viewer on Windows systems).
   - Enable debug level logging for the plug-in.
4. If plug-in logs are not generated, check to make sure that the HTTP listener (SSL) port is configured correctly. (You can check this configuration from the Administration Console.)
5. Review PPM Center logs to see if the PPM Server is receiving requests.
6. (Sun Java System Web Server only) Check to make sure that Java is disabled in the Sun Java System Web Server console.

   By default, Sun Java System Web Server is configured to process JSP files. Because of this, HTTP requests are not redirected to PPM Center.

7. If PPM Center does not receive requests from the Web server, try to access PPM Center directly using the HTTP port to isolate the issue. (The HTTP port number is the value assigned to the `HTTP_PORT` parameter in the `server.conf` file.)

Troubleshooting: Exporting PPM Dashboard Pages in PDF Format

If, after you integrate an external Web server with your PPM Center instance, you find that you cannot export PPM Dashboard pages in PDF format, do the following:

1. From the PPM Center standard interface, open the Administration Console and specify values for the following PPM Dashboard-related parameters:

   - `dashboard.Non-SSL-Port`
   - `dashboard.PDF-URL`

   The `dashboard.Non-SSL-Port` parameter specifies non-SSL port number for the PPM Dashboard to use. The `dashboard.PDF-URL` parameter specifies the PPM Dashboard URL for PDF files. (You can configure this as localhost.)

   For information about how to open the Administration Console, see *Opening the Administration Console* on page 242.

   For information about how to set parameter values in the Administration Console, see *Modifying Parameters from the Administration Console* on page 247.

   For information about dashboard configuration parameters, see *Server Configuration Parameters Related to the PPM Dashboard* on page 466.

2. Add a server configuration parameter named `WEB_SERVER` (the full name would be `com.kintana.core.server.WEB_SERVER`) to the `server.conf` file, and set its value to either `Apache` or `IIS`.

   For information about how to add a server configuration parameter to your PPM Center instance, see *Modifying Server Configuration Parameters Not Listed in the server.conf File* on page 173.

Configuring a Server Cluster

This section provides the following information about server clustering in the PPM Center environment:

- Server clustering overview
- Creating a shred folder
- Required configuration parameter settings
- Server clustering configuration on a single machine and multiple machines
- Starting and stopping servers in a cluster
- Validating the cluster configuration

Overview of Server Clustering

Before you begin to set up a PPM Server cluster, review the information provided in Chapter 2, *System Overview*, on page 23, particularly *Server Cluster Configurations (Recommended)* on page 28. The concepts described in that section are key to understanding configuring server clusters.

KINTANA_SERVER_NAME and the *<PPM_Home>*/server directory

A PPM Server consists of the common code located in the `<PPM_Home>` directory, as well as the directory of files that make up the actual PPM Server. These are separate directories in the `<PPM_Home>/server` directory.

Nodes are the individual PPM Servers that comprise a server cluster. Each node in a cluster requires a separate directory in the `<PPM_Home>/server` directory. The directory names are the server names, and you configure these in the `server.conf` file with the `KINTANA_SERVER_NAME` parameter. Each server directory in `<PPM_Home>/server` must have a corresponding `KINTANA_SERVER_NAME` defined in `server.conf`, all with the same assigned value.

Server directories cannot contain spaces, commas, or other non-alphanumeric characters, except for hyphens (-) or underscores (_). For example, `server1_1` is a valid name, but `server 1,1` is not.

@node Directive in the server.conf File

The `@node` directive in the `server.conf` file (that is, `@node` alone on a line) tells the PPM Server that the server configuration parameters listed after an `@node` are specific to one node in the cluster. You must specify one `@node` directive for each server in your cluster. Parameters displayed before the first `@node` are common to all servers in the cluster.

If you plan to deploy multiple nodes as a cluster on a single host machine, make sure that each node has its own dedicated ports (HTTP, RMI, SRMI, and so on) that do not conflict.

Modifying Server Configuration Parameters Not Listed in the server.conf File

The KNTA_SERVER_PARAM_DEF_NLS table contains all of the server configuration parameters and their default values. *Table A-1* on page 393 provides descriptions of all of the parameters in the KNTA_SERVER_PARAM_DEF_NLS table.

The `server.conf` file contains a subset of the server configuration parameters in the KNTA_SERVER_PARAM_DEF_NLS table. If a configuration parameter exists in the `server.conf` file, the value specified for it there supersedes the default value for the parameter in the KNTA_SERVER_PARAM_DEF_NLS table.

If a server configuration parameter exists in the KNTA_SERVER_PARAM_DEF_NLS table but not in the `server.conf` file, and you want to change the value of that parameter, you must add it to the `server.conf` file.

To change the value of a server configuration parameter that exists in the KNTA_SERVER_PARAM_DEF_NLS table, but is not in the `server.conf` file:

1. Stop all the nodes in the cluster.
2. Navigate to the shared folder that contains the `server.conf` file, and open the file in a text editor.

3. Do one of the following:

   - To add a parameter that is to be common to all nodes in the server cluster, type the parameter name and value before the first `@node` directive.

   - To add a parameter that is to be specific to one node, type the parameter name and value under the `@node` directive for that node.

   Use the parameter name as it is listed in *Table A-1* on page 393. Make sure that you include the prefix "`com.kintana.core.server`" in the parameter name. For example, `com.kintana.core.server.CLIENT_TIMEOUT`.

4. Save and close the `server.conf` file.

5. Run the `kUpdateHtml.sh` script on each machine.

If the servers in a cluster are running on multiple machines, then each `@node` section requires the `SERVER_NAME=<Host_Name> server.conf` directive.

Synchronizing Clocks on Machines Participating in the Server Cluster

Check to make sure that the clocks on all machines that host the nodes included in your server cluster are synchronized to within one second. If the clocks on different machines are not synchronized, use a time-synch service to synchronize them. For instructions on how to do this, go to the NIST Internet Time Service (ITS) page of the National Institute of Standards and Technology (NIST) web site (http://www.nist.gov/index.html).

Server Parameters Required for Server Clustering

Table 5-1 lists some of the server configuration parameters that you must define *for each node* in a server cluster, based on the type of clustering used. For more information about these parameters, see Appendix A, *PPM Center Configuration Parameters*, on page 389.

Table 5-1. Server configuration parameters affected by clustering (page 1 of 2)

Parameter Name [a]	External Web Server, Single Machine	External Web Server, Multiple Machines	Hardware Load Balancer, Multiple Machines
KINTANA_SERVER_NAME	X	X	
ATTACHMENT_DIRNAME		X	X
BASE_PATH		X The BASE_PATH specified for the core server is inherited by all of the @node sections. Specify this in an individual @node only if the value is different for that specific instance.	X
ORACLE_HOME		X	X
BASE_URL	X	X	X
BASE_LOG_DIR		X	
HTTP_PORT	X	X	X
EXTERNAL_WEB_PORT	X	X	
RMI_URL	X	X	X
TRANSFER_PATH		X	X
PACKAGE_LOG_DIR		X	X

Table 5-1. Server configuration parameters affected by clustering (page 2 of 2)

Parameter Name [a]	External Web Server, Single Machine	External Web Server, Multiple Machines	Hardware Load Balancer, Multiple Machines
REPORT_DIR		X	X
REQUEST_LOG_DIR		X	X

a. The parameter names listed in the table are shortened versions of the actual names, all of which start with the string com.kintana.core.server. For example, the full name of the BASE_PATH parameter is `com.kintana.core.server.BASE_PATH`.

PPM Center uses JBoss clustering technology, which enables you to set up a PPM Server cluster in various configurations. For example, you can have multiple nodes on the same host (server machine) and cluster them together. Or, you can have one or more nodes on one host and other nodes on a different host, all participating in the same server cluster.

In addition to the server configuration parameters listed in *Table 5-1*, successful server cluster setup requires that you define additional node-specific parameters to specify ports (*Table 5-3*) and cluster-specific (*Table 5-2*) server configuration parameters.

For the cluster-specific server configuration parameters listed in *Table 5-2*, you must set the same values for all nodes in the cluster.

Table 5-2. Required cluster-specific parameters

Parameter	Description
APP_SERVER_ MULTICAST_ PORT	Port used by JBoss's HAPartition service to coordinate nodes in a cluster. All nodes in the cluster must use the same value. Specify a value that is different than the value set for the MULTICAST_PORT parameter or the value for the mcast_port parameter in the `cache.conf` file (which is hardcoded to 46545).
MULTICAST_IP	IP address used for exchange of heartbeat messages, cache synchronization, and cluster communication. This must be between 224.0.0.0 and 239.255.255.255. The IP address you specify for MULTICAST_IP must *not* include the text string "http://".
MULTICAST_ PORT	Multicast port used by PPM Center's cluster monitor. You can specify any unused port number that does not conflict with other multicast ports.
MULTICAST_ CLUSTER_NAME	Used by JBoss to uniquely identify a cluster of nodes. Also used by the PPM Server to monitor the status of all nodes in a cluster. Example `server.mydomain.com/ppm` **Note:** Do not configure two clusters with the same name running on the same subnet. **Warning:** The value you specify for MULTICAST_CLUSTER_NAME must *not* include the text string `"http://"`.
mcast_port	Port used by PPM Center through the JGroups channel to synchronize cache messages across nodes in a cluster. This parameter is in the `cache.conf` file and is hardcoded to 46545.

JBoss clustering requires that you define a specific set of ports for each node in the cluster. If the cluster consists of multiple nodes on same host, you must specify unique port values for each node to prevent port collisions. *Table 5-3* lists the server configuration parameters you use to specify these ports.

Table 5-3. Required node-specific parameters for multiple nodes on a single host (page 1 of 2)

Port Parameter	Description
APP_SERVER_ NAMING_SERVICE_ RMI_PORT	JBoss Naming Service RMI port. For a stand-alone PPM Server, specify a port that is unique for the instance. For a PPM Server in a server cluster, specify a port that is unique for the node in the cluster.
APP_SERVER_ NAMING_SERVICE_ BINDING_PORT	JBoss Naming Service Binding port. For a stand-alone PPM Server, specify a port that is unique for the instance. For a PPM Server in a server cluster, specify a port that is unique for the node in the cluster.
APP_SERVER_ WEBSERVICE_ PORT	Port on which the Web service is listening. This value must be unique for each node in the server cluster.
APP_SERVER_ JRMP_INVOKER_ RMI_PORT	JBoss JRMP Invoker RMI port to which RMI clients connect when communicating through the proxy interface. For a stand-alone PPM Server, specify a port that is unique for the instance. For a PPM Server in a server cluster, specify a port that is unique for the node in the cluster.
APP_SERVER_ POOLED_INVOKER_ BINDING_PORT	Port to which the JBoss PooledInvoker is bound. This value must be unique for each node in the server cluster.
APP_SERVER_ HAJNDI_RMI_PORT	RMI port of the JBoss High Availability JNDI (HAJNDI) service. This value must be unique for each node in the server cluster.
APP_SERVER_ HAJNDI_BINDING_ PORT	Port to which the JBoss High Availability JNDI (HAJNDI) service is bound. This value must be unique for each node in the server cluster.

Table 5-3. Required node-specific parameters for multiple nodes on a single host (page 2 of 2)

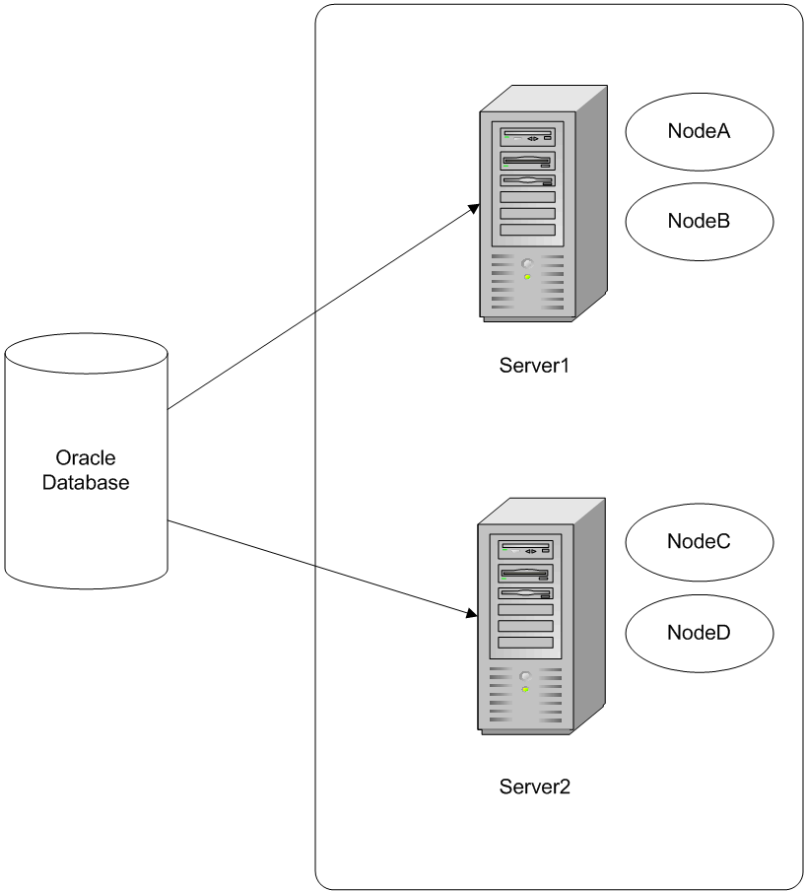
Port Parameter	Description
APP_SERVER_POOLEDHA_BINDING_PORT	Port of the JBoss Pooled High Availability JNDI (HAJNDI) service for multiple servers on the same machine. This value must be unique for each node in the server cluster.
APP_SERVER_JMX_RMI_PORT	JBoss JMX RMI port. This value must be unique for each node in the server cluster.
APP_SERVER_UIL2_BINDING_PORT	Protocol listening port for the JBoss Application Server UIL2 service. For a PPM Server in a server cluster, specify a port that is unique for the node in the cluster.

Note that if that two nodes in the same server cluster are on separate machines, and they have the same port settings, no port conflicts occur.

Server Cluster Example

Figure 5-1 shows an example of a server cluster that includes two host machines, Server1 and Server2. Server1 hosts NodeA and NodeB. Server2 hosts NodeC and NodeD. You want to create a server cluster between Server1 and Server2 and include all the nodes (NodeA, NodeB, NodeC, and NodeD) on both servers. As long as the ports assigned to each PPM Server do not overlap with any port set assigned to another PPM Server on the same host, no port conflicts occur.

Figure 5-1. Example server cluster configuration



The `server.conf` file used for this server cluster might look as follows:

```
# Common Area
..
# PPM Cluster Cluster-Specific Configuration
com.kintana.core.server.MULTICAST_IP=225.39.39.2
com.kintana.core.server.MULTICAST_PORT=9101
com.kintana.core.server.APP_SERVER_MULTICAST_PORT=9201
com.kintana.core.server.MULTICAST_NAME=APP_SERVER

com.kintana.core.server.KINTANA_SERVER_NAME=NodeA
com.kintana.core.server.HTTP_PORT=9000
com.kintana.core.server.RMI_URL=rmi://<IP of NodeA>:9001/
KintanaServer
com.kintana.core.server.EXTERNAL_WEB_PORT=9002
# PPM Cluster Node-specific ports - Using Port Set A
com.kintana.core.server.APP_SERVER_NAMING_SERVICE_RMI_PORT=1198
com.kintana.core.server.APP_SERVER_NAMING_SERVICE_BINDING_
PORT=1199
com.kintana.core.server.APP_SERVER_WEBSERVICE_PORT=8083
com.kintana.core.server.APP_SERVER_JRMP_INVOKER_RMI_PORT=4444
com.kintana.core.server.APP_SERVER_POOLED_INVOKER_BINDING_
PORT=4445
com.kintana.core.server.APP_SERVER_HAJNDI_RMI_PORT=1101
com.kintana.core.server.APP_SERVER_HAJNDI_BINDING_PORT=1100
com.kintana.core.server.APP_SERVER_POOLEDHA_BINDING_PORT=4446
com.kintana.core.server.APP_SERVER_JMX_RMI_PORT=19001
com.kintana.core.server.APP_SERVER_UIL2_BINDING_PORT=8093

@node
com.kintana.core.server.KINTANA_SERVER_NAME=NodeB
com.kintana.core.server.HTTP_PORT=10000
com.kintana.core.server.RMI_URL=rmi://<IP of NodeB>:10001/
KintanaServer
com.kintana.core.server.EXTERNAL_WEB_PORT=10002
# PPM Cluster Node-specific ports - Using Port Set B
com.kintana.core.server.APP_SERVER_NAMING_SERVICE_RMI_PORT=1298
com.kintana.core.server.APP_SERVER_NAMING_SERVICE_BINDING_
PORT=1299
com.kintana.core.server.APP_SERVER_WEBSERVICE_PORT=8183
com.kintana.core.server.APP_SERVER_JRMP_INVOKER_RMI_PORT=4544
com.kintana.core.server.APP_SERVER_POOLED_INVOKER_BINDING_
PORT=4545
com.kintana.core.server.APP_SERVER_HAJNDI_RMI_PORT=1201
com.kintana.core.server.APP_SERVER_HAJNDI_BINDING_PORT=1200
com.kintana.core.server.APP_SERVER_POOLEDHA_BINDING_PORT=4546
com.kintana.core.server.APP_SERVER_JMX_RMI_PORT=19101
com.kintana.core.server.APP_SERVER_UIL2_BINDING_PORT=8193
```

```
@node
com.kintana.core.server.KINTANA_SERVER_NAME=NodeC
com.kintana.core.server.HTTP_PORT=11000
com.kintana.core.server.RMI_URL=rmi://<IP of NodeC>:11001/
KintanaServer
com.kintana.core.server.EXTERNAL_WEB_PORT=11002
# PPM Cluster Node-specific ports - Using Port Set C
com.kintana.core.server.APP_SERVER_NAMING_SERVICE_RMI_PORT=1398
com.kintana.core.server.APP_SERVER_NAMING_SERVICE_BINDING_
PORT=1399
com.kintana.core.server.APP_SERVER_WEBSERVICE_PORT=8283
com.kintana.core.server.APP_SERVER_JRMP_INVOKER_RMI_PORT=4644
com.kintana.core.server.APP_SERVER_POOLED_INVOKER_BINDING_
PORT=4645
com.kintana.core.server.APP_SERVER_HAJNDI_RMI_PORT=1301
com.kintana.core.server.APP_SERVER_HAJNDI_BINDING_PORT=1300
com.kintana.core.server.APP_SERVER_POOLEDHA_BINDING_PORT=4646
com.kintana.core.server.APP_SERVER_JMX_RMI_PORT=19201
com.kintana.core.server.APP_SERVER_UIL2_BINDING_PORT=8293

@node
com.kintana.core.server.KINTANA_SERVER_NAME=NodeD
com.kintana.core.server.HTTP_PORT=12000
com.kintana.core.server.RMI_URL=rmi://<IP of NodeD>:12001/
KintanaServer
com.kintana.core.server.EXTERNAL_WEB_PORT=12002
# PPM Cluster Node-specific ports - Using Port Set D
com.kintana.core.server.APP_SERVER_NAMING_SERVICE_RMI_PORT=1498
com.kintana.core.server.APP_SERVER_NAMING_SERVICE_BINDING_
PORT=1499
com.kintana.core.server.APP_SERVER_WEBSERVICE_PORT=8483
com.kintana.core.server.APP_SERVER_JRMP_INVOKER_RMI_PORT=4744
com.kintana.core.server.APP_SERVER_POOLED_INVOKER_BINDING_
PORT=4745
com.kintana.core.server.APP_SERVER_HAJNDI_RMI_PORT=1301
com.kintana.core.server.APP_SERVER_HAJNDI_BINDING_PORT=1400
com.kintana.core.server.APP_SERVER_POOLEDHA_BINDING_PORT=4746
com.kintana.core.server.APP_SERVER_JMX_RMI_PORT=19301
com.kintana.core.server.APP_SERVER_UIL2_BINDING_PORT=8393
```

Creating a Shared Folder for the server.conf File

To implement a server cluster that includes nodes hosted on different machines you must have a shared folder for the server configuration file (`server.conf`). In addition to giving all nodes in a cluster access to the same `server.conf` file, the shared folder simplifies maintenance of the `server.conf` file.

The shared folder described in this section is also required to give users access to the Administration Console interface after your PPM Center instance is deployed. For information about the Administration Console, see *Administration Tools in the Administration Console* on page 241.

If you plan to configure the server cluster on multiple machines, keep in mind that the nodes in the cluster must all run on the same operating system. Shared access to the `server.conf` file does not support mixed operating systems.

The following sections provide instructions on how to prepare the shared folder on both Windows and UNIX systems.

Preparing a Shared Folder for server.conf on a Windows System

To prepare a shared folder for the `server.conf` file on Windows:

1. Create a shared folder on a file server.
2. Attach the shared folder to each machine that is to host PPM Center.
3. If you plan to host multiple PPM Server clusters (instances) under the same account on a single machine, do the following. Otherwise, proceed to step 4.
   a. Using a text editor, create a file named “ppm_server_conf.env”, and add to it the following text:

      ```
      export PPM_SERVER_CONF_DIR=//<IP_Address>/shared_folder
      ```

   b. Save the `ppm_server_conf.env` file in the *<PPM_Home>* directory and close the file.
4. Open the Control Panel and define an environment variable named `PPM_SERVER_CONF_DIR` for an account that is to run PPM Center nodes on Windows. The value of the environment variable is the location of the shared folder.

   Make sure that you use Universal Naming Convention (UNC) notation (*//<IP_Address>/<Shared_Folder>* or *<File_Server_Name>/<Shared_Folder>*) to specify the location of your shared folder.

Preparing a Shared Folder for server.conf on a UNIX System

To prepare a shared folder for the `server.conf` file on UNIX:

1. Create a shared folder on a file server.
2. Mount the shared folder to each machine that is to host PPM Center.
3. If you plan to host multiple PPM Server clusters under the same account on a single machine, do the following. Otherwise, proceed to step 4.
   a. Using a text editor, create a file named “ppm_server_conf.env”, and add to it the following text:

      ```
      export PPM_SERVER_CONF_DIR=//<IP_Address>/shared_folder
      ```

   b. Save the file to the `<PPM_Home> directory` and close the file.
4. In the `$HOME/.profile` file of the account that is to run PPM Center, add the following line:

   ```
   export PPM_SERVER_CONF_DIR=<Mount_Point>/<Shared_Folder>
   ```

High-Level Steps for Server Cluster Configuration

Server clusters are configured using the `server.conf` file. You can define all of the nodes in a cluster (cluster-specific and node-specific configuration) in one `server.conf` file, regardless of whether the nodes are running on a single machine, or distributed on different machines. Using a single `server.conf` file ensures that each node reflects the correct setting. It also enables scripts such as `kStatus.sh` to gather information from all the nodes in the server cluster, and not just the nodes that reside on the machine from which you run the script.

You can use the same `server.conf` file on different machines that host nodes that participate in the same cluster. If you do, keep in mind that you must change the machine-specific parameters settings in the file.

To configure a server cluster:

1. Stop the PPM Server. (See *Starting and Stopping the PPM Server* on page 95.
2. If your cluster is to include nodes hosted on different machines, make sure that you have set up a shared folder. (See *Creating a Shared Folder for the server.conf File* on page 183
3. If you are using an external Web server, do the following:
   a. Stop the external Web server.
   b. Configure the `workers.properties` file to include information for the multiple cluster nodes. Each node requires an external Web port defined (using the `EXTERNAL_WEB_PORT` configuration parameter).

      For information about how to configure the `workers.properties` file, see *Configuring the Workers Properties File* on page 138.
4. Configure the server nodes on the file system.
5. Manually configure the server nodes in the `server.conf` file.

The next sections provide the steps you use to configure the following server cluster setups (*Table 5-1*):

- External Web server, single machine
- External Web server, multiple machines
- Hardware load balancer, multiple machines

External Web Server, Single Machine

To set up a cluster with an external Web server on a single machine:

1. Stop the PPM Server. (See *Starting and Stopping the PPM Server* on page 95).
2. Stop the external Web server.
3. Modify the `workers.properties` file to include relevant information about the nodes in the cluster. (See *Configuring the Workers Properties File* on page 138.)
4. Create the `<PPM_Home>/server` directory.

   Make a copy of the first server directory (the entire directory) at the same level as the first.

   Example:

   ```
   <PPM_Home>
   + server
         + node1
         + node2
   ```

   Use the value specified for the `KINTANA_SERVER_NAME` parameter in the `server.conf` file that corresponds to the subdirectory node for that system.

5. Open the `server.conf` file in a text editor and add an `@node` directive for each node.
6. Before the first `@node` directive, add the cluster-specific parameters listed in *Table 5-2* on page 177.

7. After each `@node` directive, do the following:

   a. Set values for the parameters listed in *Table 5-1* on page 175 (External Web Server, Single Machine column). The values should be the same for all nodes in the cluster.

   b. Add and specify unique values for the parameters described in *Table 5-3* on page 178.

      For your convenience, *Table 5-4* on page 193 lists port set values that you can use for up to five separate nodes in a cluster. (These are simply here only for your convenience. You can use any available port numbers you want.)

8. To apply the changes to all the servers in the cluster, from `<PPM_Home>/bin`, run `kUpdateHtml.sh`.

9. If the PPM Server is running in a Windows environment, start it using the Windows service called "`PPM Server_name`," where `<Server_Name>` is the value of the `KINTANA_SERVER_NAME` parameter for the node in the cluster.

10. Generate a new service for the new node, as follows:

    a. From `<PPM_Home>/bin`, run `kConfig.sh`.

       The configuration wizard starts up.

    b. Select **Configure Windows Services.**

    c. Follow the wizard prompts to create the service.

11. To validate the cluster, use the procedure provided in *Verifying Successful Cluster Configuration* on page 195.

External Web Server, Multiple Machines

In a server cluster, a `<PPM_Home>` directory must reside on each machine, each with a server running against the same database.

To set up a cluster with an external Web server on multiple machines:

1. Install the PPM Server on the first machine in the cluster and configure it so that it is integrated with an external Web server.

   For information about how to configure a machine for integration with an external Web server, see *Configuring an External Web Server* on page 135. For information on how to integrate the PPM Server with an external Web server, see.*Integrating an External Web Server with a PPM Server* on page 168.

2. Stop the PPM Server. (See *Starting and Stopping the PPM Server* on page 95.)

3. Stop the external Web server.

4. Make sure that the common directories that the servers use (`<PPM_Home>/logs`, `<PPM_Home>/reports`, `<PPM_Home>/attachments`, and `<PPM_Home>/transfers`) are shared.

   Set the permissions for the shared directories so that users of each machine in the cluster can read from and write to them.

5. Modify the `workers.properties` file to include relevant information about the nodes in the cluster. (See *Configuring the Workers Properties File* on page 138.)

6. Modify the `server.conf` file to include an `@node` directive for each node in the cluster, including those hosted on different machines.

7. If the nodes in the cluster are running on different machines, specify the `SERVER_NAME=<Host_Name>` server configuration directive for each `@node` section.

   You must specify the `BASE_LOG_DIR, REPORT_DIR, ATTACHMENT_DIRNAME,` and `TRANSFER_PATH.` The rest of the log directories are derived from these four directories. Consider specifying these before the first `@node` so that you do not have to specify them in each and every `@node` section.

   On a Windows system, you must use the UNC format. You cannot use the local shared drive letter. Use forward slashes.

   Example

   `//<Host_Name>/<Drive_Letter><Path>)`

   To enable a node to share these directories, you must start the PPM Windows services using the PPM Center user account that has read and write permission on the shared host.

   Example

   `//com.kintana.core.server.TRANSFER_PATH=//kiwi/e$/PPM_Prod/transferpath`

   On a UNIX system, you must NFS-mount the shared directories locally with the same directory structure.

8. In the `server.conf` file, before the first `@node` directive, add the cluster-specific parameters listed in *Table 5-2* on page 177.

9. After each `@node` directive, do the following:

   a. Set values for the parameters listed in *Table 5-1* on page 175 (External Web Server, Multiple Machines column). The values should be the same for all nodes in the cluster.

   b. Add and specify unique values for the parameters described in *Table 5-3* on page 178. (For your convenience, HP provides port set values that you can use for up to five separate nodes in a cluster. These port sets are listed in *Table 5-4* on page 193.)

10. To apply the changes to all nodes on the machine that are part of the cluster, from `<PPM_Home>/bin`, run `kUpdateHtml.sh.`

11. After you configure the first server to include all additional nodes, copy the entire `<PPM_Home>/server` directory from machine1 to machine2, to the `BASE_PATH` defined in the `@node` directive.

12. Zip the file, send it using FTP, and then unzip it at the destination.

13. After you copy the file, change the directory to `<PPM_Home>/server` on the new machine, and then rename the `node1` directory to `node2`.

    The server name must match the value set for the `KINTANA_SERVER_NAME` parameter.

    Example

    The directories on machine1 could be:

    ```
    <PPM_Home>
       + server/
          + node1
    ```

    The directories on machine2 could be:

    ```
    <PPM_Home>
       + server/
          + node2
    ```

14. Put a new license on machine2, as required by the new IP address.

15. Run `kUpdateHtml.sh` on both host machines to apply the `server.conf` changes.

16. Start the PPM Server using the Windows service.

    In a multiple-machine configuration, you must generate the services on all machines running Windows.

17. Generate a new service for the new node, as follows:

    a. From `<PPM_Home>/bin`, run `kConfig.sh`.

       The configuration wizard starts up.

    b. Select **Configure Windows Services.**

    c. Follow the prompts to create the service.

The keys in the security directory are required to read encrypted values in `server.conf` and the database. The same keys must be present on all nodes in the cluster.

Hardware Load Balancer, Multiple Machines

You can use a hardware load balancer as the front end of a PPM Server cluster configuration. A hardware load balancer is similar to an HTTP reverse-proxy server and forwards HTTP requests.

All PPM Servers in a server cluster must listen for HTTP requests on a unique port. Each server in the cluster must have its `HTTP_PORT` parameter set to a unique value that does not conflict with other external applications. You specify this parameter value for all servers in a cluster in the `@node` section of the `server.conf` file.

Sticky sessions are required for hardware load balancing in the PPM Center environment.

Sample Port Sets

Table 5-4 lists five port sets that you can assign to the nodes in the `server.conf` file. These are listed here only for your convenience. You can use any available port numbers.

Table 5-4. HP-supplied port sets

Port Name[a]	Node A	Node B	Node C	Node D	Node E
APP_SERVER_NAMING_SERVICE_RMI_PORT	1198	1298	1398	1498	1598
APP_SERVER_NAMING_SERVICE_BINDING_PORT	1199	1299	1399	1499	1599
APP_SERVER_WEBSERVICE_PORT	8083	8183	8283	8383	8483
APP_SERVER_JRMP_INVOKER_RMI_PORT	4444	4544	4644	4744	4844
APP_SERVER_POOLED_INVOKER_BINDING_PORT	4445	4545	4645	4745	4845
APP_SERVER_HAJNDI_RMI_PORT	1101	1201	1301	1401	1501
APP_SERVER_HAJNDI_BINDING_PORT	1100	1200	1300	1400	1500
APP_SERVER_POOLEDHA_BINDING_PORT	4446	4546	4646	4746	4846
APP_SERVER_JMX_RMI_PORT	19001	19101	19201	19301	19401
APP_SERVER_UIL2_BINDING_PORT	8093	8193	8293	8393	8493

a. A PPM Server in a single-server configuration is assigned the Node A port configuration by default.

For a PPM Server in a single-server configuration, only a subset of these port definitions is required.

Starting and Stopping Servers in a Cluster

The procedures used to start and stop the primary node in a cluster are identical to the procedures used to start and stop the PPM Server in a single-server configuration. (For detailed information, see *Starting and Stopping the PPM Server* on page 95.)

If your PPM Center instance includes multiple nodes in a cluster configuration, you must start these nodes one at a time. Make sure that you wait until each node is fully started before you start the next node.

To start a secondary node, use the `-name server-name` argument in the `kStart.sh` script, as follows.

```
sh ./kStart.sh -name=<PPM_Server_Name>
```

To stop a secondary node, run the `kStop.sh` script, as follows:

```
sh ./kStop.sh -name=<PPM_Server_Name> -now -user <User_Name>
```

On Windows, there is one service (called "`HP PPM <PPM_Server_Name>`") per node. If you prefer to use the Windows shell command line to start nodes instead of using Windows Services, you can use the `kStart.sh` script.

If you do not have a script to stop all nodes in a cluster, you can write a script for this purpose. The following example script for the UNIX environment stops all three nodes in a cluster configuration (all nodes are on the same machine).

```
#!/bin/sh
./kStop.sh -name serv1 -now -user <User_Name>
./kStop.sh -name serv2 -now -user <User_Name>
./kStop.sh -name serv3 -now -user <User_Name>
```

A PPM Server cluster continues to operate as long as at least one node in the cluster is running. If a node stops, the HP PPM Web server module detects that the node is unavailable and stops sending it HTTP requests. When the node becomes available again, the HP PPM Web server module detects the node and sends the requests again.

If you make a change to the `server.conf` file that affects more than one node in a cluster, you must:

- Stop and restart (one at a time) all the nodes in the cluster.
- Run the `kUpdateHtml.sh` script on all machines.

Verifying Successful Cluster Configuration

To verify successful server cluster configuration:

1. If you are using an external Web server, start it and check for errors.

   If the server does not start, make sure that the values in the `workers.properties` file are correct. If you have already validated the external Web server configuration, the problem is likely in this file.

2. Start one of the nodes, and then try to connect to it.

   If you cannot connect to the node, check the `server.conf` file and correct any errors you find.

3. Start the remaining nodes in the cluster, one at a time.

   If your PPM Center instance includes multiple nodes in a cluster configuration, you must start these nodes one at a time. Make sure that you wait until each node is fully started before you start the next node.

4. Use the `kStatus.sh` script to confirm that all server nodes are running.

   If a node is not running, check the server log files in `<PPM_Home>/server/<PPM_Server_Name>/log` for errors.

In addition, make sure that:

- Multiple users logging on are automatically distributed to all nodes. Use server reports to verify which users are logged on to which nodes.
- If you shut down a node, users logged on to the other nodes can continue to work. Users logged on to the shut down node can log on again and continue to work.
- If more than one node in your cluster is dedicated to running services (recommended), and you shut down a services node, the services that were running on the node start on another services node. (For more information, see *Services Isolation* on page 35.)

If you have only one services node in a cluster, and it is shut down, the services will not run because only nodes that handle user traffic are up and running.

Multicast Settings for Server Cluster Configurations

Multicast must be enabled on network components such as network cards, switches, and routers. To avoid conflicts between cluster environments, consider the following points.

IP Address and Port Usage within a Server Cluster

The following apply to IP address and port usage in a server cluster:

- Each cluster deployment must have values specified for the `MULTICAST_CLUSTER_NAME`, `MULTICAST_IP`, `MULTICAST_PORT`, and `APP_SERVER_MULTICAST_PORT` parameters in the `server.conf` file. These values are shared by all of the nodes in a cluster.

  The IP address you specify for `MULTICAST_IP` must *not* include the text string "http://".

  To ensure that the multicast port for the server cluster does not conflict with the multicast port used by JGroups, make sure that the `MULTICAST_PORT` parameter in the `server.conf` file is *not* sent to port 46545.

- All nodes in a cluster must use *different* `APP_SERVER_MULTICAST_PORT` settings.

- All nodes in a cluster must use *different* `MULTICAST_CLUSTER_NAME` settings.

- Each cluster environment must have the `mcast_port` parameter value specified in the `cache.conf` file. This value is shared by all of the nodes in a cluster.

  The cache multicast port is hard-coded in the `cache.conf` file to 46545. Although there is no need to modify this value, be aware of it, and check to make sure that other multicast ports do not conflict with it.

IP Address and Port Usage Across Multiple Clusters within a Subnet

The following apply to IP address and port usage across multiple clusters within the same subnet:

- The `MULTICAST_CLUSTER_NAME` parameter value must be unique across clusters within a subnet.
- The combination of `MULTICAST_IP` with any one of the following ports must be unique across clusters within a network.
  - `MULTICAST_PORT`
  - `APP_SERVER_MULTICAST_PORT`
  - `mcast_port`
- If the `MULTICAST_IP` value is shared by multiple clusters within a network, then values for the `MULTICAST_PORT`, `APP_SERVER_MULTICAST_PORT`, and `mcast_port` parameters must be unique for multiple clusters within a subnet.
- If the `MULTICAST_IP` value is unique across clusters within a network, then values for the `MULTICAST_PORT`, `APP_SERVER_MULTICAST_PORT`, and `mcast_port` parameters can be duplicated across clusters within a subnet.
- All nodes in a cluster, such as a production cluster, must use the same `MULTICAST_IP`, `MULTICAST_PORT`, and `APP_SERVER_MULTICAST_PORT` settings.
- If clusters other than those related to PPM Center are set up, and these use the same multicast IP/port, the environment may also conflict.

Restricting Services to Nodes that Receive No User Traffic

HP recommends that, if you have a server cluster configured, you isolate the PPM Center background service execution on nodes that do not handle incoming user requests.

To isolate services execution to nodes that receive no user traffic:

1. Stop the nodes.

   For instructions on how to start and stop PPM Servers, see *Starting and Stopping the PPM Server* on page 95.

2. Navigate to the `<PPM_Home>` directory, and then open the `server.conf` file in a text editor.

3. Under `PPM_HOME`, do the following:

   a. To the nodes that are to receive user requests, add the following:

   ```
   com.kintana.core.server.SERVICES_ENABLED=false
   ```

   b. To the nodes that are not to receive user requests, add the following:

   ```
   com.kintana.core.server.SERVICES_ENABLED=true
   ```

   For information about adding parameters to nodes in a server cluster, see *@node Directive in the server.conf File* on page 173.

4. Navigate to the `<PPM_Home>/bin` directory, and then run the `kUpdateHtml.sh` script.

5. Start the nodes.

   You must start the nodes one at a time. Make sure that you wait until each node is fully started before you start the next node.

Switching Between Stand-Alone and Server Cluster Configurations

If you upgrade a stand-alone instance of PPM Center, and you later determine that a server cluster configuration better meets the needs of your organization, you can switch to a clustered server setup. Conversely, if you have configured a server cluster for a test or development instance and you determine that a stand-alone setup would be adequate for your immediate needs, you can transition to a stand-alone deployment. This section provides instructions for performing both of these transitions.

For information about server clustering, see *Server Cluster Configurations (Recommended)* on page 28.

Switching from Server Cluster to Stand-Alone Configuration

If you plan to migrate data from a Production instance to a Development, Test, or Sandbox instance, and you do not want to migrate all the cluster configurations, you can switch from a server cluster to a stand-alone deployment.

To switch from a server cluster configuration to a stand-alone configuration:

1. Stop all PPM Servers. (For instructions, see *Starting and Stopping the PPM Server* on page 95.)
2. Navigate to the `<PPM_Extract>/ppm910/archives` directory and locate the `deployStandalone.zip` file.
3. Extract the `deployStandalone.zip` file contents to the `<PPM_Home>/bin/` directory.
4. Run the following command:

   ```
   sh ./kStandaloneDeploy.sh
   ```

5. The script asks you to provide the .zip file name. Type the relative or absolute path to the `deployStandalone.zip` file.

The `kStandaloneDeploy.sh` script does the following:

- Removes any server cluster-related files
- Unzips the `deployStandalone.zip`
- Runs the `kUpdateHtml.sh` script.

6. After the script run is completed, run the `kStart.sh` script to start the PPM Server.

Switching from Stand-Alone to a Server Cluster Configuration

To switch from a stand-alone to a server cluster deployment:

1. Stop the PPM Server. (See *Starting and Stopping the PPM Server* on page 95.)
2. Navigate to the `<PPM_Extract>/ppm910/archives` directory and locate the `deployCluster.zip` file.
3. Navigate to the `<PPM_Home>/bin` directory, and open the `server.conf` file in a text editor.
4. Make sure that all of the following server cluster-related parameters are in the `server.conf` file (located in the `<PPM_Home>` directory), and that they are uncommented:
   - `APP_SERVER_NAMING_SERVICE_RMI_PORT`
   - `APP_SERVER_NAMING_SERVICE_BINDING_PORT`
   - `APP_SERVER_WEBSERVICE_PORT`
   - `APP_SERVER_JRMP_INVOKER_RMI_PORT`
   - `APP_SERVER_POOLED_INVOKER_BINDING_PORT`
   - `APP_SERVER_HAJNDI_RMI_PORT`
   - `APP_SERVER_HAJNDI_BINDING_PORT`
   - `APP_SERVER_POOLEDHA_BINDING_PORT`
   - `APP_SERVER_JMX_RMI_PORT`
   - `APP_SERVER_UIL2_BINDING_PORT`

5. Save and close the `server.conf` file.
6. From the `<PPM_Home>/bin` directory, run the following command:

   ```
   sh ./kClusterDeploy.sh
   ```

7. The script asks you to provide the .zip file name. Type the relative or absolute path to the `deployCluster.zip` file.
8. Complete the PPM Server cluster environment setup as described in *Configuring a Server Cluster* on page 172.
9. After you complete the server cluster setup, follow the steps described in *Verifying Successful Cluster Configuration* on page 195.

6 Implementing User Authentication

Overview of Implementing User Authentication

PPM Center uses a framework similar to Java Authentication and Authorization Service (JAAS) to integrate with pluggable authentication schemes. Integration of PPM Center with CA SiteMinder and LDAP is supported. This chapter provides information on how to integrate PPM Center with SiteMinder and LDAP, as well as instructions on how to implement either Web remote single sign-on or generic single sign-on (SSO) with PPM Center.

The following sections address different types of user authentication methods supported for use with PPM Center. They provide instructions on how to:

- Integrate PPM Center with an LDAP directory server
- Implement Web remote single sign-on with PPM Center
- Implement generic single sign-on with PPM Center
- Integrate PPM Center with SiteMinder

The user experience logging off of a PPM Center instance depends on the SSO plug-in implemented. If a user logs off (clicks **Sign Out**) of a PPM Center instance that is integrated with SiteMinder, he is logged out of both PPM Center and SiteMinder, and does not need to close the browser tab. If a user logs off (clicks **Sign Out**) of a PPM Center instance that is integrated with a plug-in that does not support log-off, users signing out from PPM Center are directed to close the browser window in order to log off.

Integrating with an LDAP Server

You can integrate PPM Center with any LDAP v3-compliant server such as Microsoft Windows Active Directory. Integrating with an LDAP server helps minimize the setup and maintenance costs associated with user account management. With an LDAP server, the PPM Server authenticates users directly to the LDAP directory server, and does not store passwords in the PPM Center database.

This section addresses LDAP directory server integration with a PPM Center. For information on how to import users from LDAP and on LDAP authentication, see the *Open Interface Guide and Reference.*

In an LDAP environment, the PPM Server authenticates users in the following way:

- The PPM Server binds to the LDAP server using the credentials supplied in the `KINTANA_LDAP_ID` and `KINTANA_LDAP_PASSWORD` server configuration parameters. If passwords are not supplied in the `server.conf` file, the PPM Server performs anonymous authentication.
- The PPM Server tries to obtain the user name by supplying a search filter to the LDAP server in the format `uid=user name`. The `uid` attribute can vary from one LDAP server to another, depending on the information supplied in the `server.conf` file.
- If the PPM Server obtains a name, it tries to rebind to the LDAP server using the name and the password supplied by the user.
- If more than one LDAP server has been specified in the `LDAP_URL` `server.conf` parameter, the PPM Server tries to authenticate against all LDAP servers until it succeeds. If the referral option is enabled, and the user is not logged on to the primary server, the PPM Server also checks the referral server for authentication.

Integrating PPM Center with an LDAP Server

To integrate PPM Center with an LDAP server:

1. Collect the following LDAP server information:
   - LDAP server URL (the default port is 389), in the following format.

     `Ldap://<LDAP_Server>:PORT`
   - LDAP base distinguished name (DN) for PPM Center users, in the following format:

     `CN=Users,DC=PPMAD,DC=com`
   - LDAP user account and password. (The PPM Server uses this information to look up users.)
   - If you are integrating with SSL-enabled LDAP, collect the following additional information.
     - Entire certificate chain. That is, `root_certificate_authority/intermediate_certificate/host_certificate`, in the BASE-64 encoded X509 (`.cer`) file format.
     - LDAP SSL port number (the default is typically 636).
2. From `<PPM_Home>/bin` on the PPM Server, run the `kConfig.sh` script.
3. Provide the information that you collected in step 1 for the following server configuration parameters in the `server.conf` file:
   - `AUTHENTICATION_MODE=ITG,LDAP`
   - `LDAP_URL`. Specify the comma-delimited list of LDAP URLs that the PPM Server queries (in the order queried). If you do not specify a port number, the server uses port number 389.

     Example

     `ldap://ldap.theurl.com:389`
   - `KINTANA_LDAP_PASSWORD`. Specify the PPM Center password on the LDAP server.

Example

```
#!#ghengis#!#
```

If you run the `kConfig.sh` script, the PPM Server configuration utility automatically encrypts this password. In this case, you must type the exact password string.

If you modify the `server.conf` file manually, you must encrypt the password string by enclosing it with the `#!!#` character delimiters (as shown in the example), and then set this encrypted string as the `KINTANA_LDAP_PASSWORD` parameter value.

- `KINTANA_LDAP_ID.` Specify the PPM Center account on the LDAP server. The PPM Server uses this to bind to the LDAP server.

  Examples

  - `KINTANA_LDAP_ID=kintana`
  - `\KINTANA_LDAP_ID=CN=kintana,CN=Users,DC=PPMAD,DC=com`

- `LDAP_BASE_DN.` Specify the base in the LDAP server from which the search is to start. If you do not specify a value, the server queries the LDAP server to determine the base.

  Example

  `LDAP_BASE_DN=CN=Users,DC=PPMAD,DC=com`

For an SSL-enabled LDAP server, provide the following additional information:

- `LDAP_SSL_PORT=636`
- `LDAP_KEYSTORE=<`*`JAVA_Home`*`>/jre/lib/security/cacerts`
- `LDAP_KEYSTORE_PASSWORD=changeit`

The script run makes the required changes to the `server.conf` file, encrypts the LDAP password, and updates the required PPM Center startup files.

4. On the PPM Server, back up the existing `LdapAttribute.conf` file, which is located in the `<PPM_Home>/integration/ldap` directory.

   The `LdapAttribute.conf` file is required for user importation and authentication. The `<PPM_Home>/integration/ldap` directory contains LDAP attribute configuration files for different types of LDAP servers.

5. Copy the appropriate `LdapAttribute_<Vendor_Name>.conf` file and overwrite the `LdapAttribute.conf` file in the same directory.

   If you are using Microsoft Active Directory, replace the `LdapAttribute.conf` file with the `<PPM_Home>/integration/ldap/LdapAttribute_AD.conf` file.

   If you are using a Sun Java System Active Server Pages LDAP server, replace the `LdapAttribute.conf` file with the `<PPM_Home>/integration/ldap/LdapAttribute_Netscape.conf` file.

6. If you are integrating with an SSL-enabled LDAP server, do the following:

   a. Get the entire trusted certificate chain of the LDAP server (`Root CA/Intermediate Certificate/host Certificate`, exported as Base-64 encoded `X509.cer` format) from your LDAP server administrator.

      If the certificate chain is not in the correct `X509.cer` format, you can import it to Internet Explorer, and then export it in the correct format.

   b. Use the JDK Keytool utility (from jdk 1.4.2 or later) to import the certificate into the `<JAVA_Home>/jre/lib/security/cacerts keystore` file.

      Your system administrator can help you use the JRE Keytool utility to import the LDAP server certificate chain into the JDK cacerts file.

c. Change to the `<JAVA_Home>/jre/lib/security` directory, and run the command:

```
keytool -import -trustcacerts -alias <SSL_LDAP_Host>
-file <SSL_LDAP_CERT.cer> -keystore cacerts
```

The default cacerts keystore password is “changeit.” For tighter security, you may want to change this password.

7. To enable entity ownership and security, do the following:

   a. Make sure that the PPM Server is running.

   b. Use the Import Users report to import the LDAP user into the KNTA_USERS table on the PPM Server.

      For instructions on how to run the Import Users report, see the *Open Interface Guide and Reference*.

      If you are running the Import Users report for the first time, edit the `LdapAttribute.conf` file and comment out the `MANAGER_USERNAME`, `LOCATION_MEANING`, and `DEPARTMENT_MEANING` parameters. If you do not make these changes, the import fails and an error message such as “Unknown Manager,” “Unknown Location,” or “Unknown Department” is displayed. The error occurs because the import tries to validate the data before the data is imported.

   c. For the **LDAP Import?** option, click **Yes.**

Authenticating Against Multiple LDAP Domains

PPM Center can handle multiple domains during LDAP authentication. To configure this feature, you add the server configuration parameter `LDAP_URL_FULL` to the `server.conf` file.

The values for the `LDAP_URL_FULL` parameter include a space-delimited (not comma-delimited) list of full LDAP URLs. Each LDAP URL must specify a base distinguished name (DN), which is used in place of the `LDAP_BASE_DN` server configuration parameter.

Example of how to set the `LDAP_URL_FULL` parameter:

```
com.kintana.core.server.LDAP_URL_FULL=ldap://<Host>.<Your_
Domain>.com/CN=Users,DC=<Your_Domain>,DC=com ldap://
<Host>.<Your_Domain>.com/OU=Users2,DC=<Your_Domain>,DC=com
```

Disabling the LDAP_URL parameter

If you add the `LDAP_URL_FULL` parameter to the `server.conf` file, make sure that you comment out the `LDAP_URL` parameter. The `LDAP_URL` parameter supersedes the `LDAP_URL_FULL` parameter so that, if both are specified in the `server.conf` file, PPM Center uses the value set for `LDAP_URL`.

If the URLs provided for `LDAP_URL_FULL` do not have a DN value, PPM Center uses the value set for `LDAP_BASE_DN`.

To specify a space character inside of a URL, use the URL-encoding scheme, and replace the space with `%20`. For example, if you have an organizational unit called My Org Unit, then specify `My%20Org%20Unit` in the LDAP URL.

For more information about server parameters related to LDAP integration, see *LDAP Attribute Parameters* on page 478.

Validating LDAP Parameters

You can use any of several available tools to validate and troubleshoot the LDAP configuration parameters. For example, Softerra provides Softerra LDAP Browser freeware, which you can download and install. You can then use the LDAP server information you collected in step 1 to create a new LDAP server profile. This confirms that the information is correct. On the LDAP browser windows at the top, blue line, you can view the DN for a specific resource. Use this to determine the base DN as well as the search filter for the Import Users report. (To download the Softerra LDAP Browser software, go to the Softerra LDAP Administrator site.)

Implementing Web Remote Single Sign-On with PPM Center

This section provides information on how to implement Web remote single sign-on with PPM Center. This implementation is based on NTLM authentication and requires that the PPM Server(s) be integrated with an external Web server running Microsoft IIS.

Web remote single sign-on works with PPM Center as follows:

1. A user logs in to a Windows desktop.
2. The user accesses PPM Center through the external (IIS) Web server.
3. The user is authenticated through the Windows user account to IIS and the user name is passed to the PPM Server by way of the `REMOTE_USER` HTTP header field.
4. If the user is a valid PPM Center user, the standard interface and PPM Dashboard open.

Requirements for Implementing Web Remote Single Sign-On

To implement Web remote single sign-on, your system must meet the following requirements:

- PPM Center must be set up with an external Microsoft IIS Web server. For information on how to do this, see *Integrating an External Web Server with a PPM Server* on page 168.
- To ensure that you have the required access rights, make sure that the system username you use to log on to PPM Center is same as the account username for the active directory.
- Clients must use Microsoft Internet Explorer to log on to PPM Center. Logon credentials are not automatically passed from Web browsers other than Internet Explorer (for example, Firefox) when connecting to IIS.

Setting Up Web Remote Single Sign-On with PPM Center

To configure Web remote single sign-on with PPM Center:

1. Integrate the external IIS Web server with the PPM Server(s).

   For information about how to integrate the external Web server with a PPM Server, see *Integrating an External Web Server with a PPM Server* on page 168.

2. On the PPM Server, do the following:

   a. Stop the PPM Server.

   b. Open the `server.conf` file in a text editor, and then add to it the following:

   ```
   com.kintana.core.server.SINGLE_SIGN_ON_PLUGIN
   =com.kintana.sc.security.auth.WebRemoteUserSingleSignOn
   ```

   For information on how to edit the `server.conf` file, see Appendix A, *PPM Center Configuration Parameters,* on page 389.

   c. Save and then close the `server.conf` file.

d. Run the `kUpdateHtml.sh` script.

For information about the `kUpdateHtml.sh` script, see *kUpdateHtml.sh* on page 505.

3. On the IIS external Web server, do the following:

   a. From IIS Microsoft Management Console, select the default Web site.

   b. Right-click the default Web site, and then click **Properties** on the shortcut menu.

   c. Click the **Directory Securities** tab.

   d. Under **Anonymous access,** click **Edit.**

   e. Deselect the **Anonymous Access** checkbox.

   f. Leave the **Integrated Windows authentication** checkbox selected.

   g. Stop, and then restart the IIS Windows service.

   Do not use Basic Authentication. If you do, the Web server does not force the browser to authenticate, and so does not result in a single sign-on solution.

4. Stop, and then restart the PPM Server.

For information on troubleshooting issues you may encounter with Web remote single sign-on, see *Troubleshooting Your Single Sign-On Implementation* on page 215.

Implementing Generic Single Sign-On with PPM Center

This section provides information on how to configure PPM Center to use the generic single sign-on module to integrate with third-party authentication servers.

Single sign-on works as follows:

1. A user logs on to a portal that has been configured to use a third-party authentication application.
2. The user accesses the PPM Center standard interface through an external Web Server integration that is part of the logged-in domain.
3. The PPM Server receives the user information through the HTTP header specified in the `sso.conf` file.
4. If the user is a valid PPM Center user, he is granted access to the PPM Center standard interface and PPM Dashboard.

Requirements for Implementing Generic Single Sign-On

To implement generic single sign-on with PPM Center, your PPM Center system be integrated with an external Web server (Sun Java System Web Server, an Apache-based server, or IIS).

Setting Up Generic Single Sign-On with PPM Center

To implement generic single sign-on:

1. Regarding the third-party authentication application you plan to use:
   a. To configure the third-party application, follow the instructions provided with the application.
   b. Verify that the PPM Center user is also a valid single sign-on user and can be authenticated.

2. External Web server:

   a. Integrate PPM Center with the external Web server.

      For information on how to integrate an external Web server, see *Integrating an External Web Server with a PPM Server* on page 168.

   b. Configure the external Web server to integrate with the third-party authentication application. For information on how to do this, see the documentation provided with the with third-party authentication application.

   c. Make sure that the authenticated user's HTTP request is forwarded to the PPM Server with the user ID inserted into the HTTP header specified in the `sso.conf` file.

      You can find the `sso.conf` file in the `<PPM_Home>/integration/sso` directory.

3. PPM Server configuration

   a. Verify that the `sso.conf` file has the following setting.

      ```
      USERNAME=<Authenticated_User_Header>
      ```

      where `<Authenticated_User_Header>` is the header your single sign-on system uses to store the user ID of the authenticated user. For example, CA SiteMinder uses `HTTP_SM_USER`.

   b. Add the following line to the `server.conf` file.

      ```
      com.kintana.core.server.SINGLE_SIGN_ON_
      PLUGIN=com.kintana.sc.security.auth.GenericSingleSignOn
      ```

   c. Run the `kUpdateHtml.sh` script, which is located in the `<PPM_Home>/bin` directory.

4. Stop, and then restart the PPM Server.

For information on troubleshooting issues you may encounter with single sign-on, see *Troubleshooting Your Single Sign-On Implementation*.

Troubleshooting Your Single Sign-On Implementation

Determine the header information that the single sign-on server is sending.

1. Check the timestamp as follows:

   a. Open the `server.conf` file in a text editor, and set the value of the `ENABLE_WEB_ACCESS_LOGGING` parameter to `true`.

   For information on how to edit the `server.conf` file, see Appendix A, *PPM Center Configuration Parameters,* on page 389.

   b. Run the `kUpdateHtml.sh` script.

   c. Restart the PPM Server.

   For details on how to stop and start the PPM Server, see *Starting and Stopping the PPM Server* on page 95.

   d. Log on to PPM Center.

   e. Check the timestamp on the PPM Server.

   f. Navigate to the `<PPM_Home>/server/<PPM_Server_Name>/log` directory.

   g. Open the `<Date>.access.log` file and check the timestamp.

2. Open the `logging.conf` file (located in the `<PPM_Home>/conf` directory) in a text editor, and add the following text.

```
com.kintana.core.logging.PRODUCT_FUNCTION_LOGGING_LEVEL =
com.kintana.web.filter.debug, DEBUG
com.kintana.core.logging.PRODUCT_FUNCTION_LOGGING_LEVEL =
com.kintana.sc.authentication, DEBUG
com.kintana.core.logging.SYSTEM_THRESHOLD = DEBUG
```

3. Restart the PPM Server by running the following:

```
sh ./kStart.sh -debug
```

   The information is written to the `<PPM_Home>/bin/serverLog_<Debug_Timestamp>.txt` file.

4. Enable logging on the single sign-on agent side, and then check the information passed back and forth. Check for any error messages reported.

After you check for problems and error messages, you can remove the debugging code you added to the `logging.conf` file in step 2.

Integrating PPM Center with CA SiteMinder

You can configure PPM Center to delegate user authentication to CA SiteMinder for both the standard (Web) and PPM Workbench interfaces. The configuration supports two authentication modes: mixed and Single Sign-On (SSO).

Mixed Mode

In the mixed mode configuration, PPM Center users can continue to log on using the PPM Center logon page. Within the PPM Server, the integrated SiteMinder Authentication Module routes the logon request to an existing SiteMinder Policy Server for authentication. This mode is referred to as mixed because you can configure PPM Center to use both SiteMinder and its own authentication simultaneously. In this case, the authentication mode to be used must be specified in each PPM Center user account.

Integration Architecture for Mixed Mode

In a mixed mode configuration, users log on to PPM Center, and the integrated SiteMinder Authentication Module passes logon information to the SiteMinder Policy Server for authentication.

To use mixed mode, you must configure the integrated SiteMinder Authentication Module correctly. An external Web server can be used, but is not required. For information about external Web servers supported, see the *System Requirements and Compatibility Matrix*.

For PPM Workbench clients, once the user provides a username and password in the logon page, the user authentication information is passed to the SiteMinder Policy Server for verification. Once verified, the information is passed to the PPM Workbench applet for automatic logon. After it starts, the applet communicates directly with the PPM Server.

Figure 6-1 shows a system diagram of the SiteMinder integration in mixed mode. *Figure 6-2* shows the integration architecture for mixed mode with the optional external Web server.

Figure 6-1. SiteMinder integration architecture for mixed mode

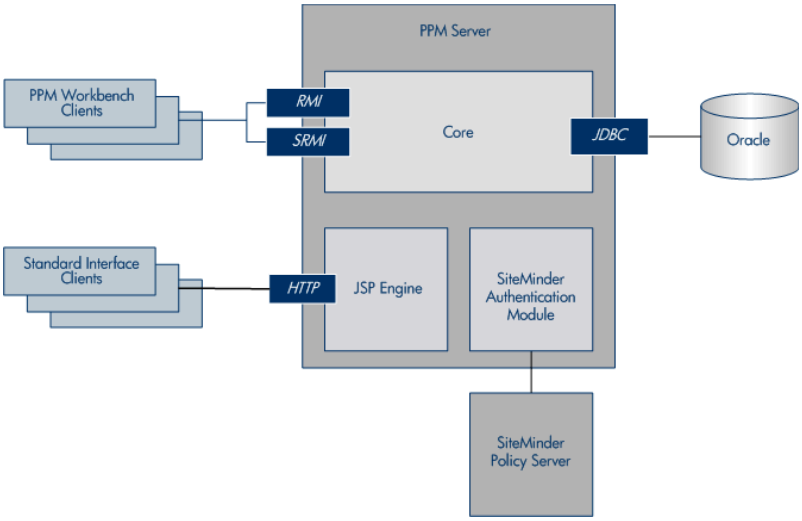
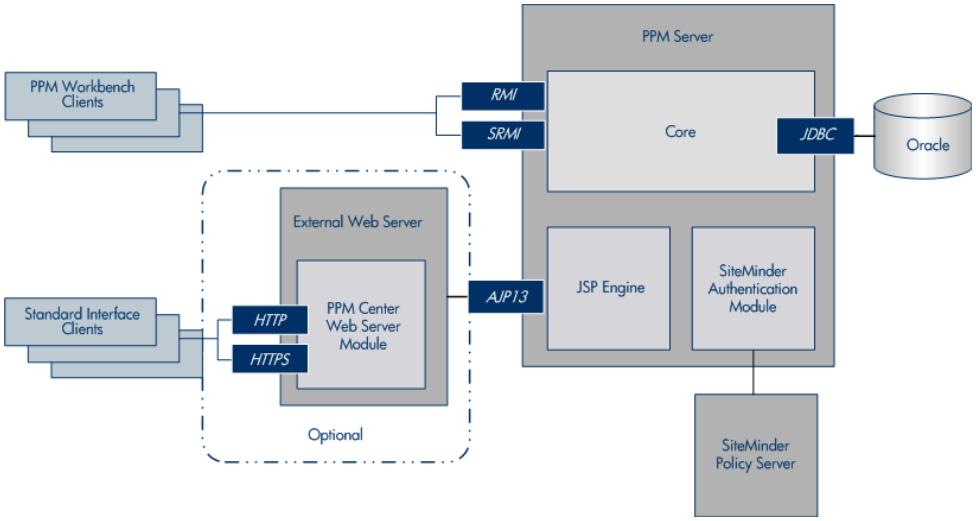


Figure 6-2. SiteMinder integration architecture for mixed mode with optional external Web server



Single Sign-on Mode

In the SSO mode configuration, Web requests are authenticated before being passed to PPM Center, bypassing the PPM Center logon page. To enable SSO mode, the SiteMinder Web Agent must be plugged into any third-party Web server software that PPM Center supports, and be configured to communicate with a SiteMinder Policy Server. The SiteMinder Web Agent intercepts Web requests and checks with the Policy Server to ensure they are authenticated before passing them to PPM Center.

Note that you cannot use SiteMinder to manage PPM Center application-level authorization for controlling access to various screens and functions. Application-level authorization is controlled by the PPM Center security model using security groups, access grants, product licensing, and so on. Therefore, user accounts must exist in both PPM Center and the SiteMinder Policy Server, but PPM Center does not have to maintain the associated passwords.

Integration Architecture for Single Sign-On Mode

Single sign-on configuration requires that PPM Center be integrated with an external Web server that has both the SiteMinder Web Agent and PPM Center Web Server Module installed. (The PPM Center internal Web server does not support SiteMinder SSO because there is no compatible Web agent or a suitable API to create one.)

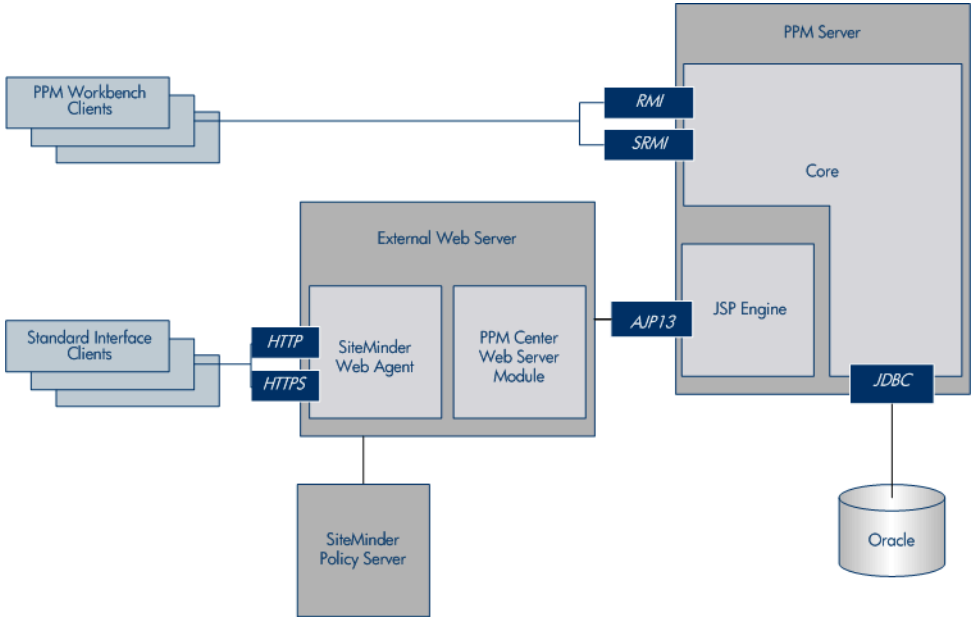
The SiteMinder Web Agent is the single access point for all Web clients. The SiteMinder Web Agent intercepts all incoming requests and ensures that they are authenticated before passing them to the PPM Center Web Server module. The requests then proceed to the PPM Server.

For PPM Workbench clients, the SiteMinder Web Agent protects access to the PPM Workbench logon page. After the user provides a username and password, the authentication information is passed to the PPM Workbench applet for automatic logon. Once started, the applet communicates directly with the PPM Server.

PPM Workbench does not support SSO mode if you start it from the menu bar (select **Administration > Open Workbench on Desktop**). However, if PPM Center is launched as an application, it uses SiteMinder to authenticate. See *Configuring the PPM Workbench to Run as a Java Applet* on page 129.

Figure 6-3 shows a system diagram of SiteMinder integration in SSO mode.

Figure 6-3. SiteMinder integration architecture for SSO mode



Requirements for Integrating with SiteMinder

The requirements for integrating PPM Center with SiteMinder are as follows:

- An External Web server (required for SSO mode, optional for mixed mode)
- PPM Center Web Server Module
- CA SiteMinder version 6.0 (for both SSO and mixed modes) is installed and functioning correctly

For information on how to install SiteMinder, see the product documentation.

- SiteMinder Java Agent API is installed (for mixed mode only)

Overview of Integrating PPM Center with SiteMinder

PPM Center integration with SiteMinder involves the following tasks:

1. Configure PPM Center for integration with SiteMinder
2. Configure SiteMinder for integration with PPM Center.

The configuration of SiteMinder for integration with PPM Center must be performed by a SiteMinder administrator.

The following sections provides instructions for performing these tasks.

Configuring PPM Center for Integration with SiteMinder

To configure PPM Center to integrate with SiteMinder:

1. Verify that your PPM Center installation is functioning correctly.
2. If you plan to use mixed authentication mode, do the following:
   a. Install the SiteMinder Java Agent API on the PPM Server:
      - On a Windows system, copy the `smjavaagentapi.jar` file to the `<PPM_Home>\server\<PPM_Server_Name>\deploy\itg.war\WEB-INF\lib` directory.
      - On a UNIX system, copy the `smjavaagentapi.jar` file to the `<PPM_Home>/server/<PPM_Server_Name>/deploy/itg.war/WEB-INF/lib` directory.

      These JAR and DLL files are available on the SiteMinder Developer SDK CD. You can also find these files in the SDK home directory. The PPM Server automatically includes the JAR file in its CLASSPATH upon server startup.

   b. Install the SiteMinder Agent native code, as follows:
      - On a Windows system navigate to the `C:\Program Files\netegrity\sdk\bin` folder, and then copy the following files to the `<PPM_Home>\integration\siteminder` directory:

        ```
        smagentapi.dll
        smerrlog.dll
        smjavaagentapi.dll
        ```

        (or, for SiteMinder 6.0 SP1, the `smjavaagentapi.jar` file) to the `<PPM_Home>\integration\siteminder` directory.

        Regardless of which directory you place the DLL files in, check to make make sure that you include the directory path in the PATH system environment variable.

      - On a UNIX system, set the CA SiteMinder SDK-related variables (such as `LD_LIBRARY_PATH`, `PATH`, `CLASSPATH`, `LIBPATH`, and `SHLIB_PATH`) so that the system can find the JNI support library. Next, navigate to the `/Program Files/netegrity/sdk/java`

directory, and then copy the `smjavaagentapi.jar` file to the `<PPM_Home>/integration/siteminder` directory.

For information about which variables to set for which platforms, and what values to set for them, see the guidelines provided in the CA SiteMinder SDK documentation.

3. (Mixed mode only) Open the `siteminder.conf` file (located in the `<PPM_Home>/integration/siteminder` directory), and make sure that the settings for the following SiteMinder parameters match the corresponding settings in the SiteMinder setup:

   - `SM_ACCOUNTING_PORT`
   - `SM_AGENT_NAME`
   - `SM_AUTHENTICATION_PORT`
   - `SM_AUTHORIZATION_PORT`
   - `SM_CONNECTION_MAX`
   - `SM_CONNECTION_MIN`
   - `SM_CONNECTION_STEP`
   - `SM_CONNECTION_TIMEOUT`
   - `SM_POLICY_SERVER`
   - `SM_PROTECTED_URL`
   - `SM_SHARED_SECRET`

   Pay particular attention to the value set for `SM_AGENT_NAME`.

   If any SiteMinder settings are modified later, you must update the `siteminder.conf` file to reflect these changes.

4. (Optional, but recommended) Create a backup copy of the PPM Server `server.conf` file.

5. For mixed mode authentication only:

   a. To enable selection of either SiteMinder or PPM authentication for PPM Center users, in the `server.conf` file, modify the authentication mode as follows:

   ```
   com.kintana.core.server.AUTHENTICATION_MODE
   =ITG,SiteMinder
   ```

b. Comment out the following parameter setting in the `server.conf` file.

```
com.kintana.core.server.SINGLE_SIGN_ON_PLUGIN
=com.kintana.sc.security.auth.SiteMinderSingleSignOn
```

c. Stop, and then restart the PPM Server.

d. From the User Workbench, (from the PPM Workbench shortcut bar, select **Sys Admin > Users**), change the users' authentication mode to SiteMinder.

You may want to set a few user accounts to use the PPM authentication mode to enable access to PPM Center in the event that the SiteMinder Policy Server is unavailable.

6. For SSO mode only:

a. To enable only SiteMinder authentication for PPM Center users, in the `server.conf` file, change the authentication mode as follows.

```
com.kintana.core.server.AUTHENTICATION_MODE=SiteMinder
```

b. In the `server.conf` file, specify the use of SSO as follows.

```
com.kintana.core.server.SINGLE_SIGN_ON_PLUGIN
=com.kintana.sc.security.auth.SiteMinderSingleSignOn
```

When both the SiteMinder Web Agent and PPM Center Web server module are installed on the external Web server, the SiteMinder Web Agent always takes precedence for requests in the form of `/itg/*`.

7. Stop, and then restart the PPM Server.

Configuring PPM Center Users

To configure PPM Center users to authenticate using SiteMinder, complete the following steps:

1. Make sure that the usernames for PPM Center users match those used by SiteMinder.
2. Make sure that PPM Center users are set up to use SiteMinder authentication.

   In SSO mode, users whose authentication mode is set to anything other than SiteMinder are forced to log on to SiteMinder. Users not set up correctly in SiteMinder are locked out of PPM Center. If this occurs, revert to the `server.conf` file you created in step 4 on page 223, and then make the necessary changes to the user accounts before resetting the authentication mode in the `server.conf` file.

Configuring SiteMinder for Integration with PPM Center

Before you configure SiteMinder for use with PPM Center, make sure that the Policy Server is working correctly and that the User Directory to be used for PPM Center authentication is correctly configured. The SiteMinder Test Tool is useful for verifying that the installation is functioning correctly.

Configuring SiteMinder for PPM Center is the same as configuring any other type of protected resource in SiteMinder. Use the SiteMinder Policy Server User Interface to update the SiteMinder configuration entities as necessary. For both mixed and SSO modes, four standard SiteMinder configurations should exist: Host Configuration Object, User Directory, Policy Domain, and Policy.

To configure SiteMinder for integration with PPM Center, perform the following steps.

These steps must be performed by a SiteMinder administrator.

1. Create a new Web agent.
2. (Mixed mode only) If you plan to use mixed-mode authentication, then after you create a new Web agent, do the following:
   a. Make sure that the 4.x compatibility flag is set.
   b. Specify the name of the PPM Server, and a secret password.
   c. In the `siteminder.conf` file, set the following parameters:
      - Set the `SM_AGENT_NAME` parameter value to the PPM Server name.
      - Set the `SM_SHARED_SECRET` parameter value to the secret password you specified.
3. Create a new Web Agent Conf object.
4. Double-click the new Agent Conf Object to open the Properties window.

5. Add the new property value LogOffUri to `/itg/web/knta/global/Logout.jsp`.

   PPM Center uses the `LogoffUri` property to log off users correctly when they log off of the PPM Center standard interface.

6. Create a realm for PPM Center to protect resource `/itg/*`, and specify the name of the agent you created in step 3 for this realm.

7. Configure and enable two rules for the realm (one to enable HTTP on GET, POST, and PUT actions, and another to enable OnAuthAccept action as the authentication event) with the following settings:

   - Rule 1. Set the **Name** field to **AllowHTTP,** the **Resource** field to **/itg/*,** and the **Action** field to **GET,POST,PUT.**
   - Rule 2. Set the **Name** field to **OnAuthAccept,** the **Resource** field to **/itg/*,** and the **Action** field to **OnAuthAccept.**

8. Specify URLs for the CookieDomain and CookieProvider parameters in the agent configuration object for the SiteMinder Web Agent that is to authenticate PPM Center Web requests.

   Cookies are used to track session and idle timeouts.

   The format used to specify the value for CookieProvider depends upon the external Web server you use:

   - For Microsoft IIS, Sun ONE, and Sun Java System Web servers, use the following format.

     `http://<Server_Domain>:<Port>/siteminderagent/SmMakeCookie.ccc`

     represents the host name or IP address where your PPM Center instance is accessed.

   - For Apache, use the following format.

     `http://<Server_Domain>:<Port>/SmMakeCookie.ccc`

It is important to understand that PPM Center reads the information that SiteMinder automatically injects into the HTTP Request header.

PPM Center relies on the following user attributes:

- `SM_USER`. For an authenticated user, this parameter specifies the user distinguished name (DN). For an unauthenticated user, this is the user ID as specified by the user at logon.
- `SM_SERVERSESSIONID`. This parameter specifies the session ID of a user who has already authenticated, or the session ID that is to be assigned to the user upon successful authentication.
- `SM_SERVERSESSIONSPEC`. This parameter specifies the user's session ticket.

For configuration details for these and other SiteMinder parameters, see the SiteMinder documentation.

7 Maintaining the System

Overview of System Maintenance

This chapter provides information on how to maintain your PPM Center instance. The initial sections include descriptions of tools available in the PPM Center standard interface, the Administration Console, and the PPM Workbench. Later sections address the maintenance tasks required to keep your PPM Center instance running smoothly.

This chapter includes the following sections:

Administration Tools in the Standard Interface

The following sections provide information about the administration tools you can access through the PPM Center standard interface:

- *Viewing Server Running Server Reports, Requests, and Packages*
- *Viewing Running Executions*
- *Viewing Interrupted Server Reports, Requests, and Package Executions*
- *PPM Center Background Services*

Viewing Server Running Server Reports, Requests, and Packages

To view running reports, requests, and packages:

1. Log on to PPM Center.
2. On the **Open** menu, click **Administration > Report Execution > View Running Reports.**

The View Running Reports page opens and lists any reports, requests, and packages currently running.

Viewing Running Executions

To view running executions:

1. Log on to PPM Center.
2. On the **Open** menu, click **Administration > Report Execution > View Running Executions.**

   The View Running Executions page opens, and the **Summary** section lists any distributions, server reports, requests, or packages that are running.
3. If any reports are listed as running, click **View Running Reports.**

Viewing Interrupted Server Reports, Requests, and Package Executions

To view interrupted server reports, requests, and package executions:

1. Log on to PPM Center.
2. On the **Open** menu, click **Administration > Report Execution > View Interrupted Executions.**

   The View Interrupted Executions page opens and lists interrupted executions (if any exist).
3. In the list below **View Interrupted Executions for a Server Startup,** select the date of the interrupted execution to view.
4. To view the details of the selected interrupted execution listed in the **Failed Executions** section, click **View.**

Return to *Administration Tools in the Standard Interface*.

PPM Center Background Services

This section provides information about the background services available in PPM Center, and instructions on how to enable and schedule them. It also provides guidelines for scheduling background services to optimize resource use and system performance.

Table 7-1 lists the PPM Center background services.

For information on background services monitoring in PPM Center, see *Monitoring Activity in PPM Center* on page 310.

Table 7-1. Background services in PPM Center (page 1 of 7)

Service Name	Description
ALM Startup	Ensures that the quartz scheduler that synchronizes PPM Center and Service Manager is running.
Applet Key Cleanup	Periodically removes old records from the database table KNTA_APPLET_KEYS. (These are temporary, system-generated keys used for one-time access to the system—for example, if a user wants to open the PPM Workbench.)
Commands Cleanup	Periodically removes old records from the prepared commands tables.
Concurrent Request Watch Dog	When HP Deployment Management submits a concurrent request (job) to Oracle Apps, this service polls Oracle to determine what state the job is in, and when it has completed.
Cost Rate Rule Update	After it checks for changes to cost rules and cost factors, this service: • Updates time sheet costs stored on the time sheet • Updates financial summaries that are synched to staffing profiles • Adds projects to the queue for Cost Rollup Service, which updates project cost in the workplan and in the financial summary.

Table 7-1. Background services in PPM Center (page 2 of 7)

Service Name	Description
Cost Rollup	Cost rollup service asynchronously rolls up planned and actual costs (entered manually or pulled from time sheets) from leaf tasks to root tasks in workplans, and then pushes the data to the financial summary. In addition, the cost rollup service rolls up actual costs from time sheets to financial summaries for proposals and assets.
Debug Messages Cleanup	Periodically removes old records from the KNTA_DEBUG_MESSAGES database table, which can collect a lot of temporary data.
Directory Cleanup	Cleans up files in the dynamic content directory. The PPM Server generates these files and writes temporarily to the dynamic content directory so that they can be accessed over the Web. After the scheduled number of days, the files are deleted because they are no longer necessary.
Document Cleanup	Periodically checks for documents that are no longer attached to a PPM Center entity, and removes those it finds from the PPM Center file system.
Evaluate TM Approvers	An HP Time Management service that periodically checks to determine whether the resource assigned to approve a timesheet has changed.
Exception Rule	Periodically checks to determine whether active projects are running on time. Determines if and when task exceptions are recalculated. For more information about this service, see the *HP Project Management Configuration Guide.*
Field Security Pending Denormalization	Because managing field-level security is computationally expensive, whenever the security settings at the field level are updated, this service performs calculations that ensure live security checks in performance.
Financial Metrics Update	Calculates net present value (NPV) and nominal return for HP Financial Management.

Table 7-1. Background services in PPM Center (page 3 of 7)

Service Name	Description
Financial Summary Rollup	Calculates rollups of financial information, including forecast and actual costs and benefits (monthly data) and approved budgets (annual data), for the following: • Rollups from proposals, projects, and assets to a program • Rollups from proposals, projects, assets, programs, and subportfolios to a portfolio, along with immediate rollups to all the successively higher levels in the portfolio hierarchy The following events, performed by manual entry or by another background service such as Cost Rollup or Web services, trigger this rollup service: • Addition or removal of items in the program or portfolio • A change to the financial summary of any item in the program • A change to the financial information of any item in the portfolio
FX Rate Update	Recalculates cost after financial exchange (FX) rates change.
Interface Tables Cleanup	Periodically removes old records from the database open interface tables.
Logon Attempts Cleanup	Periodically removes old records from the KNTA_LOGON_ATTEMPTS database table, which contains records of all logon attempts.
Mobility Access	Enables PPM Center users to process approval workflow steps from desktop email or a PDA device. Resources working outside of an office or without VPN access can act on approval workflow steps without having to first log on to PPM Center. For information about PPM Center Mobility Access, see the *HP Demand Management Configuration Guide.*

Table 7-1. Background services in PPM Center (page 4 of 7)

Service Name	Description
Notification Cleanup	Deletes rows (older than the current date minus the number of days set for the notification cleanup service) from the KNTA_NOTIF_TXN_PARENTS table in the database. The service then deletes all child rows from the KNTA_NOTIF_TXN_DETAILS, KNTA_NOTIF_TXN_COLUMNS, and KNTA_NOTIF_TXN_RECIPIENTS tables.
Notification	Enables the notification service. You can use this parameter to turn off notifications for copies of production instances being used for testing, and turn them on again when the system goes to production.
Oracle Statistics Calculation	Automatically collects statistics for the cost-based optimizer.
Pending Assignments Table Cleanup	Periodically checks for duplicate rows in the KNTA_PENDING_ASSIGNMENTS table. This parameter is related to the Work Item Pending Assignment service. If a work item is updated more than once between runs of the work item breakdown service, the KNTA_PENDING_ASSIGNMENTS table contains duplicate rows. This service removes the duplicates.
Pending Cost EV Update	Asynchronously applies external updates to the Pending Cost EV Updates service when updates cannot be made immediately.
Pending EV Updates Table Cleanup	Removes duplicate rows in the Pending EV Updates table.
Performance Log Cleanup	Deletes data from the Performance Log table (PPM_PERFORMANCE_LOG) in the database. The PERF_LOG_DAYS_TO_KEEP parameter determines how long records remain in the table. All records older than the number of days specified by this parameter are deleted from the table.
Project Health	Automatically updates project health indicators.

Table 7-1. Background services in PPM Center (page 5 of 7)

Service Name	Description
Project Planned Value Update	This service handles synchronization between requests (as cases of blocking predecessors) and between requests and tasks if a request is updated and the target entity is locked.
Reference Update	Automatically updates references between entities.
Request Status Export	Determines whether any request status values were changed since the service last ran. If status values have changed, and if the updated requests reference remote entities, then the status values for the referenced remote entities are updated.
Resource Pool Rollup	Performs resource pool rollup (between child and parent resource pools.)
RM Notification	Resource pool and staffing profile notification service. This service must be enabled in order to send notifications to staffing profile managers, resource pool managers and resources. For more information, see the *HP Resource Management User's Guide.*
Service to update the Projected Total values for Budgets and Staffing Profiles	Periodically updated the projected totals for budgets and staffing profiles.
Shared Lock Cleanup	Cleans up any entries left in the shared lock table after a PPM Server crash.
Staffing Profile Financial Summary Sync	Synchronizes staffing profile data with financial summary data at a configurable interval. You can schedule the service so that synchronization does not happen automatically whenever changes are made to a staffing profile or a budget. If updates are frequent, delaying synchronization can help preserve system performance.
Staffing Profile Linked Budget Sync Service	Synchronizes the budgets linked with the staffing profiles that are updated through the Web service. The Web service update creates an entry in the ITG_ PENDING_ROLLUPS table for a staffing profile when its positions are updated from the Web service.

Table 7-1. Background services in PPM Center (page 6 of 7)

Service Name	Description
Staffing Profile Period Sum Update	Rolls up actuals from time sheets and projects/tasks to staffing profiles. Whenever a time sheet or project/task is updated, the actuals are displayed on the linked staffing profile only after this service runs.
Synchronize Documentum Folder/ Security Group Name	The HP Document Management module uses PPM Center entity names (project names or request type names) to name the folders and security groups in the EMC Documentum repository. As those entity names change in PPM Center, this background service picks them up and applies the changes to associated items in the Documentum repository. For more information, see the *Document Management Guide and Reference*.
Task Actual Rollup	Determines if and how frequently periodic task actual roll-ups are calculated. Asynchronously rolls up actuals provided through HP Time Management or the My Tasks portlet. For more information about this service, see the *HP Project Management Configuration Guide.*
Task Scheduler	Determines if the work plan schedule health is recalculated and the frequency with which work plan schedule health is recalculated. For more information about this service, see the *HP Project Management Configuration Guide.*
Time Sheet Notifications	Enables notifications to be sent on time sheets.
TM-PM Sync	Synchronizes time sheet updates from HP Time Management to project work plan tasks in HP Project Management, at a configurable interval. Each time the service runs, it sends a message to the queue for each work plan that must be synchronized with time sheets. The service ensures that roll-ups for each work plan can be accumulated and updated once, if necessary, per work plan.

Table 7-1. Background services in PPM Center (page 7 of 7)

Service Name	Description
Work Item Pending Assignment	Periodically populates the KRSC_ WORK_ITEM_ ASSIGNMENTS table, which is used for resource work load information. The service retrieves the actuals information from the request.
Work Item Pending Update	On the PPM Server, periodically calls KRSC_ PROCESS_PENDING_UPDATES.Patrol to process updates to work items.
Workflow Timeout Reaper	Scans all active workflow steps to verify that they have timed out according to the settings for the step.

Return to *Administration Tools in the Standard Interface*.

Running Services on Multiple Nodes

You can run multiple instances of the same type of service concurrently to process different entity IDs on the same or different nodes in a server cluster. For recommendations on how to schedule and run background services, see *Minimizing the Performance Impact of Running Background Services* on page 296.

Enabling and Scheduling PPM Center Services

You can enable and schedule the PPM Center background services through the standard interface.

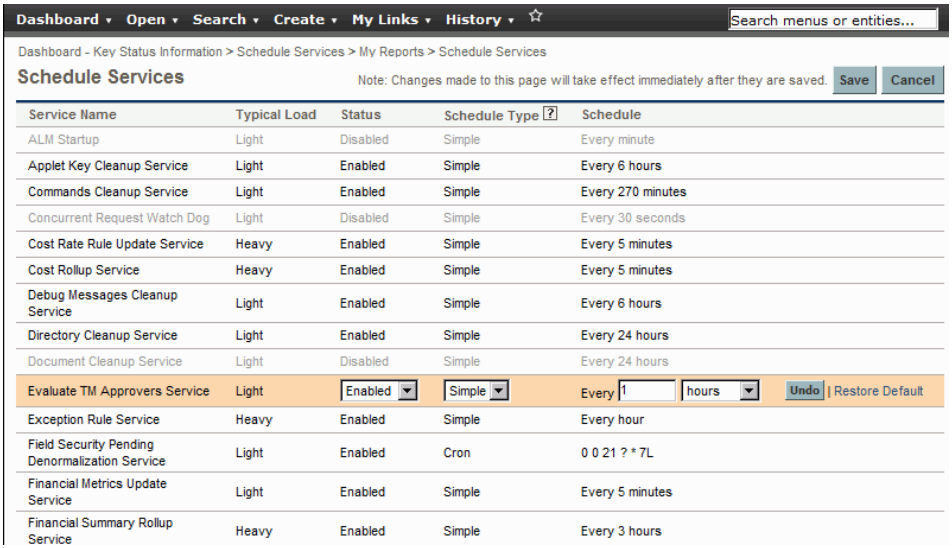
To enable and schedule PPM Center services:

1. Log on to PPM Center.
2. On the **Open** menu, click **Administration > Schedule Services.**

   The Schedule Services page lists all of the available services, and shows the typical load each service manages, whether the service is enabled, the type of expression used to schedule the service, and the current run schedule.

3. Click the table row that displays the service you want to enable, disable, or schedule.

   The editable fields for that service are enabled.



Service Name	Typical Load	Status	Schedule Type	Schedule
ALM Startup	Light	Disabled	Simple	Every minute
Applet Key Cleanup Service	Light	Enabled	Simple	Every 6 hours
Commands Cleanup Service	Light	Enabled	Simple	Every 270 minutes
Concurrent Request Watch Dog	Light	Disabled	Simple	Every 30 seconds
Cost Rate Rule Update Service	Heavy	Enabled	Simple	Every 5 minutes
Cost Rollup Service	Heavy	Enabled	Simple	Every 5 minutes
Debug Messages Cleanup Service	Light	Enabled	Simple	Every 6 hours
Directory Cleanup Service	Light	Enabled	Simple	Every 24 hours
Document Cleanup Service	Light	Disabled	Simple	Every 24 hours
Evaluate TM Approvers Service	Light	Enabled	Simple	Every 1 hours
Exception Rule Service	Heavy	Enabled	Simple	Every hour
Field Security Pending Denormalization Service	Light	Enabled	Cron	0 0 21 ? * 7L
Financial Metrics Update Service	Light	Enabled	Simple	Every 5 minutes
Financial Summary Rollup Service	Heavy	Enabled	Simple	Every 3 hours

The typical load values assigned to services are based extensive testing and feedback from the field. Light services are short-lived with low resource consumption. Heavy services take longer to run and are more resource-intensive than light services. You cannot modify these values.

4. To enable or disable the service, from the **Status** list, select **Enabled** or **Disabled.**

5. To select the type of expression to use to schedule the service, from the **Schedule Type** list, select either **Simple** or **Cron.**

   If you use a cron expression to schedule a service, keep in mind that cron expressions take into account the `TIME_ZONE` parameter setting for the PPM Server on which the service runs. In a server cluster environment, servers can be running on machines located in different time zones.

6. In the **Schedule** column, do one of the following:

   - To schedule the service using a simple expression, type a number in the first field and, from the list on the right, select the time unit (**seconds, minutes,** or **hours.**)
   - To schedule the service using a cron expression, type the expression in the text field. For detailed help on how to compose a cron expression, under the **Schedule Type** column heading, select the Help icon (?).

     If you use a cron expression to schedule a service, keep in mind that the value you type in the **Schedule** field cannot exceed 40 characters.

7. For each additional service you want configure, repeat step 3 through step 6.

8. After you have finished configuring services, click **Save.**

Your changes take effect immediately after you save them. There is no need to restart the PPM Server.

If a service misses one or more of its scheduled runs because, for example, the PPM Server is shut down, the service is run as soon as the server is restarted.

For information about how to view the current status of background services, see *Viewing the Services Audit Results Page* on page 320.

Return to *Administration Tools in the Standard Interface*.

Administration Tools in the Administration Console

The following sections provide information about the administration tools you can access through the Administration Console interface:

- *Opening the Administration Console*
- *Using the Administration Console to View PPM Server Status*
- *Using the Administration Console to View and Modify Parameters*
- *Working with Fiscal Periods from the Administration Console*
- *Changing the Display of Data in Administration Console Tables*

In order to access and use the Administration Console, you must:

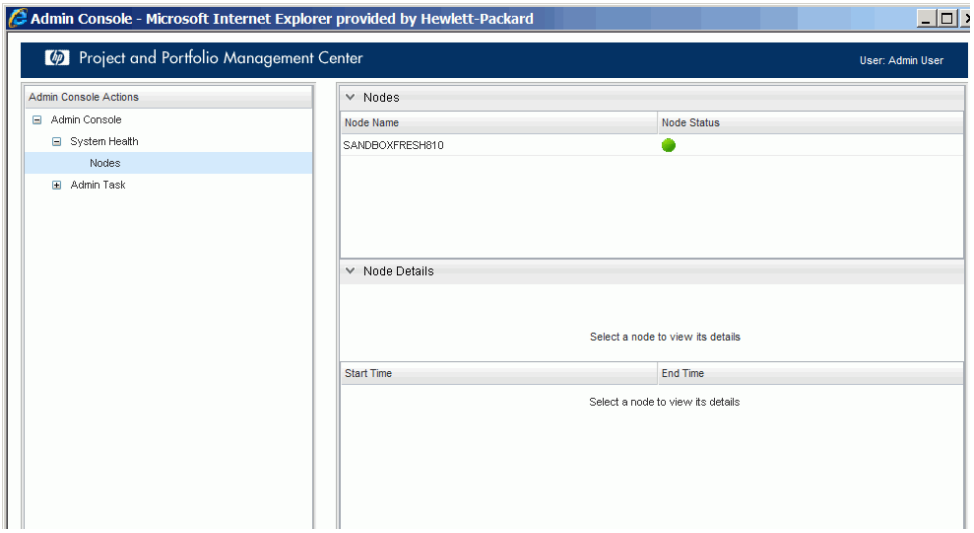
- Have the User Administration license
- Belong to a security group that has the Sys Admin Server Tools: Execute Admin Tools access grant.

Opening the Administration Console

You can open the Administration Console from the PPM Center standard interface.

To open the Administration Console from the standard interface:

1. Log on to PPM Center.
2. From the menu bar, select **Open > Administration > Open > Administration > Administration Console.**



The left panel of the Administration Console window displays the **System Health** and **Administration Task** sections. The right panel displays details about the item selected in the left panel. The **Nodes** menu item under **System Health** is selected by default.

If your PPM Center instance has a server cluster configuration, the **Node Name** column lists all of the nodes in the cluster. If yours is a single-server configuration, then the single PPM Server name is displayed.

If you configure secure Web logon for PPM Center, you can launch Administration Console on HTTPS. For details, see *Configuring Secure Web Logon (Optional)* on page 110.

Return to *Administration Tools in the Administration Console*.

Using the Administration Console to View PPM Server Status

You can use the Administration Console in PPM Center to quickly assess the status of the nodes configured for your instance.

For information about how to generate the Server Status Report from the Admin Tools window in the PPM Workbench, see *Running Server Reports from the Admin Tools Window* on page 263.

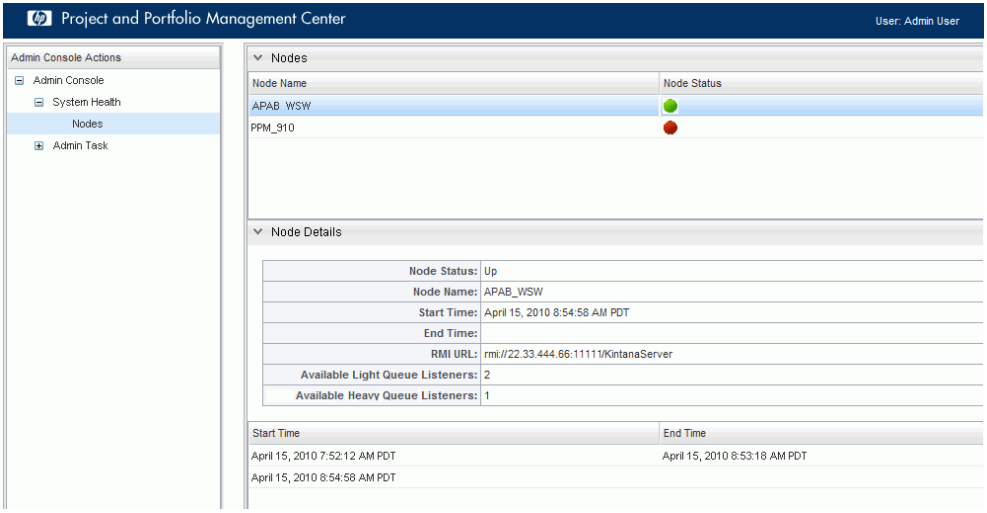
To view the status of nodes in a server cluster from the Administration Console:

1. Open the Administration Console. (See *Opening the Administration Console* on page 242.)

   In the **System Health** section in the left panel, **Nodes** is selected.

   The **Nodes** table in the right panel lists all nodes in the cluster. Icons in the **Node Status** column indicate that a server is up ( ) or down ( ).

2. To view detailed information about a specific node in a cluster, in the **Nodes** table, click the node name.



The following table shows the information that the **Node Details** table displays for the selected node:

Field	Description
Node Status	Node status (either Up or Down).
Node Name	Node name.
Start Time	Day of the week, calendar date, and time the node was last started.
End Time	If the node is down, this field shows when this node was stopped (weekday, calendar date, and time).
RMI URL	URL for Java RMI. Format: `rmi://<IP_address>:<Port>/<Object>`
Available Light Queue Listeners	Number of listeners available on the node to run light background services.
Available Heavy Queue Listeners	Number of listeners available on the node to execute heavy background services.

The **Start Time** and **End Time** columns in the **Node Details** table display the complete history of start and stop times (calendar date and times) for the selected node.

Return to *Administration Tools in the Administration Console*.

Using the Administration Console to View and Modify Parameters

The Administration Console lists two types of server configuration parameters: *static* and *non-static*. After you change the value of a static parameter, you must restart the PPM Server(s) to implement the change. If you change the value of a non-static parameter, there is no need to restart the PPM Server.

The Administration Console displays read-only parameters in gray text. Read-only parameters are either sensitive parameters such as passwords or parameters that cannot be changed without compromising the PPM Center system.

If you start Administration Console using HTTPS, then sensitive parameters that were formerly displayed in grey text become editable and are displayed normally. If, for some reason, you must modify values for other read-only parameters, you must either run the `kconfig.sh` script, or edit the `server.conf` file directly.

The Administration Console displays only the parameters that are defined in knta_server_param_def_nls table. If a parameter is not listed in the Administration Console, then it is probably missing from the knta_server_param_def_nls table.

Although you can modify server configuration parameter values directly in the `server.conf` file or using the configuration tool (`kConfig.sh` script), HP recommends that you modify server configuration parameters using the Administration Console. (See *Modifying Parameters from the Administration Console* on page 247.)

You cannot use the Administration Console to either add parameters to or remove parameters from the `server.conf` file.

For information about how to run the `kConfig.sh` script, see *Standard Configuration* on page 100. For information about how to change parameter values in the `server.conf` file, see Appendix A, *PPM Center Configuration Parameters,* on page 389.

Return to *Administration Tools in the Administration Console*.

Viewing Parameters from the Administration Console

To view parameters from the Administration Console interface, do the following:

1. Open the Administration Console. (See *Opening the Administration Console* on page 242.)

2. In the left panel of the Administration Console window, expand the **Administration Task** section, and then click **Application Configuration.**

   The **Edit PPM Application Parameters** table in the right panel lists all the server configuration parameters, along with their descriptions and current values for the selected scope. (The **Scope** value defaults to **Cluster,** or for a stand-alone instance, the name of the single PPM Server for the instance.)

3. (Cluster configuration only) To specify the scope of the parameters to view, do one of the following:

   - To view the parameters that are common to all of the nodes in the cluster, from the **Scope** list, select **Cluster.**

   - To view parameters for a specific node, from the **Scope** list, select the node name.

   If you select the name of the primary node from the **Scope** list, no parameters are listed in the **Edit PPM Application Parameters** table. Instead, parameters and values for the primary node are listed after you select **Cluster** from the **Scope** list.

Return to *Administration Tools in the Administration Console*.

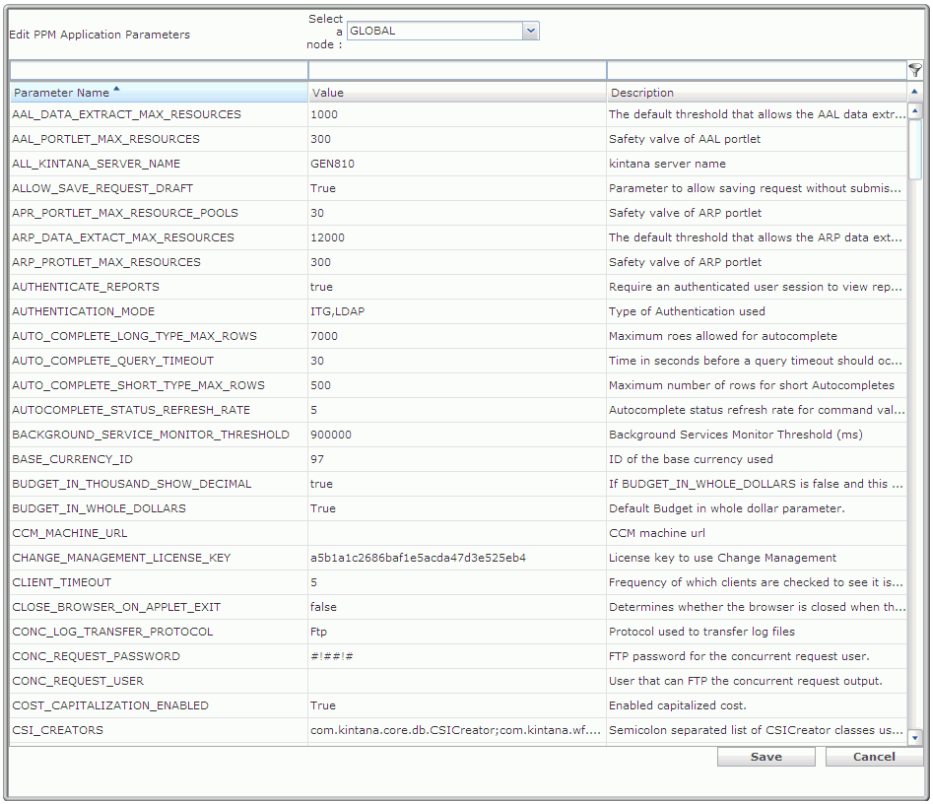
Modifying Parameters from the Administration Console

To modify parameters from the Administration Console interface:

1. Open the Administration Console. (See *Opening the Administration Console* on page 242.)

2. In the left panel of the Administration Console window, expand **Administration Task,** and then click **Application Configuration.**

   If your PPM Center is a stand-alone instance, the **Scope** list displays the name of the single PPM Server. If you have a server cluster configured, then the **Scope** list displays **Cluster,** and the names all nodes in the cluster.

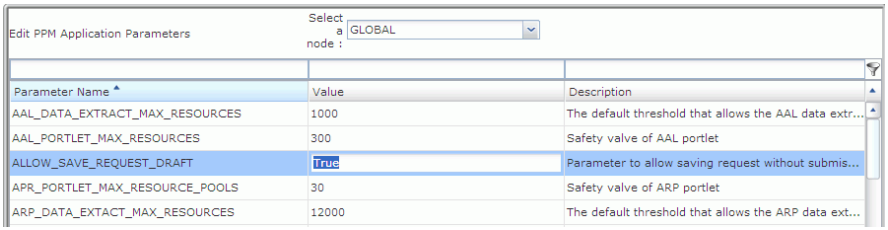


Edit PPM Application Parameters — Select a node : GLOBAL

Parameter Name	Value	Description
AAL_DATA_EXTRACT_MAX_RESOURCES	1000	The default threshold that allows the AAL data extr...
AAL_PORTLET_MAX_RESOURCES	300	Safety valve of AAL portlet
ALL_KINTANA_SERVER_NAME	GEN810	kintana server name
ALLOW_SAVE_REQUEST_DRAFT	True	Parameter to allow saving request without submis...
APR_PORTLET_MAX_RESOURCE_POOLS	30	Safety valve of ARP portlet
ARP_DATA_EXTACT_MAX_RESOURCES	12000	The default threshold that allows the ARP data ext...
ARP_PROTLET_MAX_RESOURCES	300	Safety valve of ARP portlet
AUTHENTICATE_REPORTS	true	Require an authenticated user session to view rep...
AUTHENTICATION_MODE	ITG,LDAP	Type of Authentication used
AUTO_COMPLETE_LONG_TYPE_MAX_ROWS	7000	Maximum roes allowed for autocomplete
AUTO_COMPLETE_QUERY_TIMEOUT	30	Time in seconds before a query timeout should oc...
AUTO_COMPLETE_SHORT_TYPE_MAX_ROWS	500	Maximum number of rows for short Autocompletes
AUTOCOMPLETE_STATUS_REFRESH_RATE	5	Autocomplete status refresh rate for command val...
BACKGROUND_SERVICE_MONITOR_THRESHOLD	900000	Background Services Monitor Threshold (ms)
BASE_CURRENCY_ID	97	ID of the base currency used
BUDGET_IN_THOUSAND_SHOW_DECIMAL	true	If BUDGET_IN_WHOLE_DOLLARS is false and this ...
BUDGET_IN_WHOLE_DOLLARS	True	Default Budget in whole dollar parameter.
CCM_MACHINE_URL		CCM machine url
CHANGE_MANAGEMENT_LICENSE_KEY	a5b1a1c2686baf1e5acda47d3e525eb4	License key to use Change Management
CLIENT_TIMEOUT	5	Frequency of which clients are checked to see it is...
CLOSE_BROWSER_ON_APPLET_EXIT	false	Determines whether the browser is closed when th...
CONC_LOG_TRANSFER_PROTOCOL	Ftp	Protocol used to transfer log files
CONC_REQUEST_PASSWORD	#!##!#	FTP password for the concurrent request user.
CONC_REQUEST_USER		User that can FTP the concurrent request output.
COST_CAPITALIZATION_ENABLED	True	Enabled capitalized cost.
CSI_CREATORS	com.kintana.core.db.CSICreator;com.kintana.wf....	Semicolon separated list of CSICreator classes us...

Save Cancel

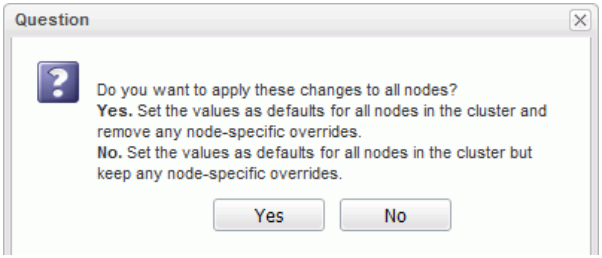
3. Click the **Value** box for the parameter to modify, and then type a new value to replace the existing value.



4. Repeat step 3 for each parameter you want to change, and then click **Save.**

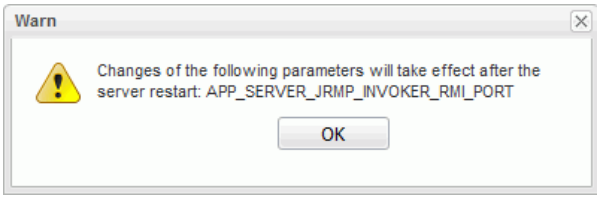
Make sure that you save your changes before you close the Administration Console. Otherwise, any changes you made are lost.

5. (Cluster only) If **Cluster** is the selected scope, and you change the value of a parameter, and then click **Save,** a dialog box opens and gives you the option of applying the changed parameter value across all nodes.



To apply the new value to all nodes in the cluster, click **Yes.** To retain node-specific overrides for the parameter, click **No.**

6. If you change the value of a static parameter, the following warning is displayed to advise you that you must restart the PPM Server to implement the change.



7. To implement your changes, stop the nodes, run `<PPM_Home>/bin/kUpdateHtml.sh`, and then restart the nodes, one at a time.

For information about how to stop and start PPM Servers, see *Starting and Stopping the PPM Server* on page 95. For information about the `kUpdateHtml.sh` script, see *kUpdateHtml.sh* on page 505).

The parameter values that you modify from the Administration Console take effect the next time the parameter values are used.

Return to *Administration Tools in the Administration Console*.

Working with Fiscal Periods from the Administration Console

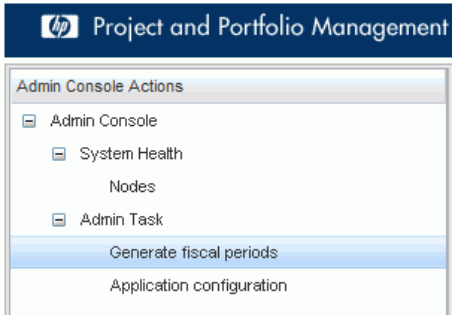
You can use the Administration Console to generate fiscal periods that reflect your organization's fiscal calendar. You can also use Administration Console to generate translations of fiscal period names, shift existing fiscal periods, and import and export fiscal periods.

For information about how to use the `kGenFiscalPeriods.sh` script to generate fiscal periods, see the *Generating Fiscal Periods* guide.

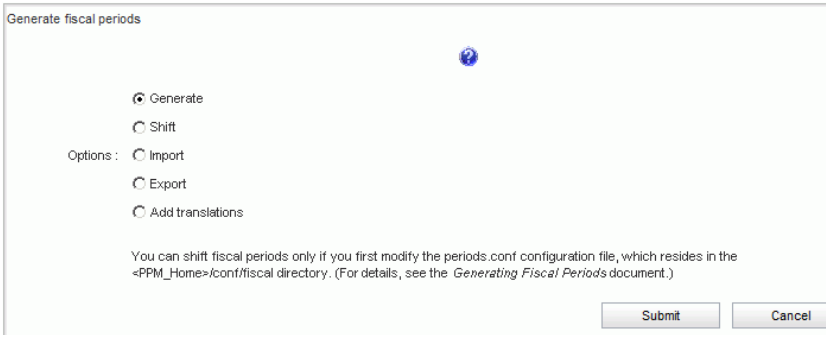
Generating Fiscal Periods from the Administration Console

To generate fiscal periods from the Administration Console:

1. Open the Administration Console. (See *Opening the Administration Console* on page 242.)
2. In the left panel of the Administration Console, expand the **Administration Task** section, and then select **Generate fiscal periods.**



3. In the **Generate fiscal periods** panel on the right, leave the **Generate** option selected.



4. In the **Start Year** and **End year** boxes, type the starting and ending years for the fiscal periods you want to generate.

   The Administration Console generates the fiscal periods, and then lists all existing fiscal periods (for all period types) in the Administration Console.

   If a gap exists between the latest existing fiscal period year and the starting year you specify, the Administration Console generates fiscal periods for all of the intervening years.

5. To persist the generated fiscal periods, click **Commit.**

6. To implement your changes, stop, and then restart, the PPM Servers.

   For information about how to stop and start PPM Servers, see *Starting and Stopping the PPM Server* on page 95.

Return to *Administration Tools in the Administration Console*.

Using the Administration Console to Shift Existing Fiscal Periods

So that all fiscal periods match the fiscal year, you can use the Administration Console to do one or more of the following:

- Change the starting day of the week (`START_DAY_OF_WEEK`) of your organization's fiscal year
- Change the starting month (`START_MONTH` and `START_MONTH_FOR_NEXT_FISCAL_YEAR`) of your organization's fiscal year.

The changes you make apply to existing fiscal periods as well as to fiscal periods to be generated later, so that all fiscal periods match the fiscal year.

Shifting fiscal periods changes period data in the database. HP strongly recommends that you back up the configuration file before you perform this procedure. For information about how shifting fiscal periods affect functionality in PPM Center, see the *Generating Fiscal Periods* guide.

To shift existing fiscal periods from the Administration Console:

1. Open the `periods.conf` file in a text editor. The `periods.conf` file is located in the `<PPM_Home>/conf/fiscal` directory.
2. To change the starting month, change the `START_MONTH` parameter value to the number that represents the month the fiscal year starts. For example, you would use 11 to represent November.

   For detailed information about how to set values for the parameters in the `periods.conf` file, see the *Generating Fiscal Periods* guide.

3. To change the starting month of your organization's fiscal year, set the `IS_START_MONTH_FOR_NEXT_FISCAL_YEAR` parameter value to `true` or `false`, depending on the relationship between fiscal years and calendar years.

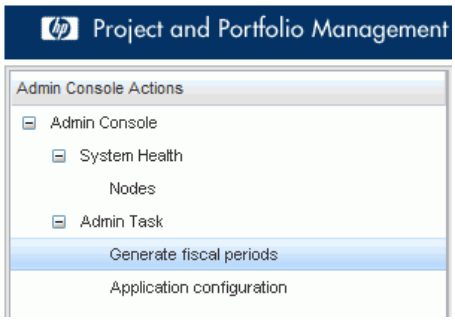
   The default value of `false` indicates that the start month does *not* belong to the next fiscal year.

4. To change the starting day of the week, set the `START_DAY_OF_WEEK` parameter to a number between 1 and 7, with 1 representing Sunday, and 7 representing Saturday.

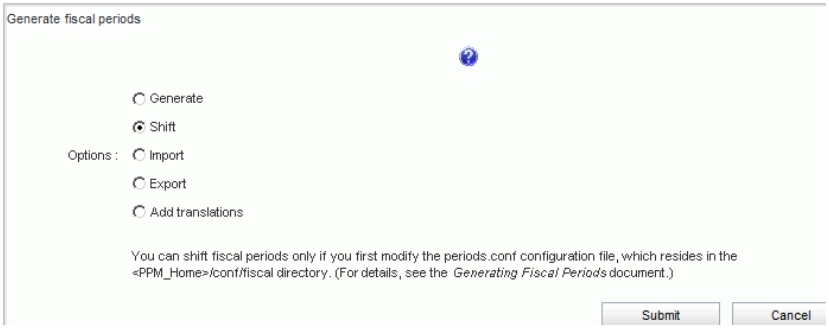
5. Save and close the `periods.conf` configuration file.

6. Open the Administration Console. (See *Opening the Administration Console* on page 242.)

7. In the left panel of the Administration Console, expand the **Administration Task** section, and then select **Generate fiscal periods.**



8. In the right panel, select the **Shift** option.



9. Click **Submit.**

   The Administration Console shifts the existing fiscal periods, and then lists all fiscal periods (for all period types) in the Administration Console.

10. To persist the shifted fiscal periods, click **Commit.**

11. To implement your changes, restart the nodes. For instructions, see *Starting and Stopping the PPM Server* on page 95.

The adjusted fiscal periods apply to the display of financial data in all languages.

Using the Administration Console to Import Fiscal Periods

To import the modified period definitions into PPM Center:

1. Open the Administration Console. (See *Opening the Administration Console* on page 242.)
2. In the left panel, expand the **Administration Task** section, and then select **Generate fiscal periods.**
3. In the right panel, select the **Import** option.

4. Click **Browse,** and then navigate to and select the `<Period_Definitions_Filename>.csv` period definition file.
5. Click **Submit.**

   The imported fiscal periods are listed in the right panel.
6. To persist the imported fiscal periods, click **Commit.**
7. To implement your changes, stop, and then restart, the PPM Servers.

For information about how to stop and start PPM Servers, see *Starting and Stopping the PPM Server* on page 95.

Return to *Administration Tools in the Administration Console*.

Using the Administration Console to Export Fiscal Periods

To export current fiscal period definitions from PPM Center to a specific file:

1. Open the Administration Console. (See *Opening the Administration Console* on page 242.)
2. In the left panel, expand the **Administration Task** section, and then select **Generate fiscal periods.**
3. In the right panel, select the **Export** option.

4. In the **File** box, type the name of the `<Period_Definitions_Filename>.csv` file to which you want to copy fiscal period definitions.

5. Click **Submit.**

   The **Exported Periods** tab lists the fiscal periods to be exported.

Exported Periods

Start Date	End Date	Period Type	Long Name	Short Name
Jan 1, 1998	Dec 31, 1998	YEAR	1998	98
Jan 1, 1998	Jun 30, 1998	HALF_YEAR	H1 1998	H1 98
Jul 1, 1998	Dec 31, 1998	HALF_YEAR	H2 1998	H2 98
Jan 1, 1998	Mar 31, 1998	QUARTER	Q1 1998	Q1 98
Apr 1, 1998	Jun 30, 1998	QUARTER	Q2 1998	Q2 98
Jul 1, 1998	Sep 30, 1998	QUARTER	Q3 1998	Q3 98
Oct 1, 1998	Dec 31, 1998	QUARTER	Q4 1998	Q4 98
Jan 1, 1998	Jan 31, 1998	MONTH	January 1998	Jan 98
Feb 1, 1998	Feb 28, 1998	MONTH	February 1998	Feb 98
Mar 1, 1998	Mar 31, 1998	MONTH	March 1998	Mar 98
Apr 1, 1998	Apr 30, 1998	MONTH	April 1998	Apr 98
May 1, 1998	May 31, 1998	MONTH	May 1998	May 98
Jun 1, 1998	Jun 30, 1998	MONTH	June 1998	Jun 98
Jul 1, 1998	Jul 31, 1998	MONTH	July 1998	Jul 98
Aug 1, 1998	Aug 31, 1998	MONTH	August 1998	Aug 98
Sep 1, 1998	Sep 30, 1998	MONTH	September 1998	Sep 98
Oct 1, 1998	Oct 31, 1998	MONTH	October 1998	Oct 98
Nov 1, 1998	Nov 30, 1998	MONTH	November 1998	Nov 98
Dec 1, 1998	Dec 31, 1998	MONTH	December 1998	Dec 98
Jan 1, 1999	Dec 31, 1999	YEAR	1999	99
Jan 1, 1999	Jun 30, 1999	HALF_YEAR	H1 1999	H1 99
Jul 1, 1999	Dec 31, 1999	HALF_YEAR	H2 1999	H2 99
Jan 1, 1999	Mar 31, 1999	QUARTER	Q1 1999	Q1 99
Apr 1, 1999	Jun 30, 1999	QUARTER	Q2 1999	Q2 99
Jul 1, 1999	Sep 30, 1999	QUARTER	Q3 1999	Q3 99
Oct 1, 1999	Dec 31, 1999	QUARTER	Q4 1999	Q4 99
Jan 1, 1999	Jan 31, 1999	MONTH	January 1999	Jan 99
Feb 1, 1999	Feb 28, 1999	MONTH	February 1999	Feb 99
Mar 1, 1999	Mar 31, 1999	MONTH	March 1999	Mar 99

Export to file Cancel

6. Click **Export to File.**

   The File Download dialog box opens.

7. Click **Save.**

8. Navigate to the PPM Server directory in which you want to store the file, and the save the file.

   On the PPM Server, the default directory for the fiscal periods definition file is `<PPM_Home>/bin/fiscal/output`.

9. Check the directory you specified and verify that the file you exported now resides there.

For information about how to use the `kGenFiscalPeriods.sh` script to export fiscal period definitions, see the *Generating Fiscal Periods* guide.

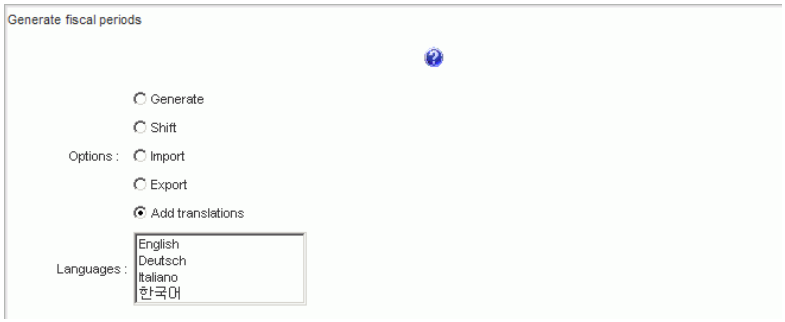
Return to *Administration Tools in the Administration Console*.

Using the Administration Console to Generate Translations for Fiscal Periods

You can use the Administration Console to generate translations for fiscal periods in any of the languages installed on PPM Center. Month names and period formats are as specified in the language configuration files. After you generate the translations, users see the fiscal periods in the session language selected at logon. The translated periods cover the same time span as the periods covered for existing languages.

To generate translations of fiscal periods in any languages installed on PPM Center:

1. Open the Administration Console. (See *Opening the Administration Console* on page 242.)
2. In the left panel, expand the **Administration Task** section, and then select **Generate fiscal periods.**
3. In the right panel, select the **Generate translations** option.



4. In the **Languages** list, select one or more of the languages. (You can use the **`Ctrl`** key or the `Shift` key to select multiple languages.)

5. Click **Submit.**

   In the right panel, a tab that lists the translated periods is displayed for each of the selected languages.

Italian

Start Date	End Date	Period Type	Long Name	Short Name
Jan 1, 1998	Dec 31, 1998	YEAR	1998	98
Jan 1, 1998	Jun 30, 1998	HALF_YEAR	H1 1998	H1 98
Jul 1, 1998	Dec 31, 1998	HALF_YEAR	H2 1998	H2 98
Jan 1, 1998	Mar 31, 1998	QUARTER	Q1 1998	Q1 98
Apr 1, 1998	Jun 30, 1998	QUARTER	Q2 1998	Q2 98
Jul 1, 1998	Sep 30, 1998	QUARTER	Q3 1998	Q3 98
Oct 1, 1998	Dec 31, 1998	QUARTER	Q4 1998	Q4 98
Jan 1, 1998	Jan 31, 1998	MONTH	gennaio 1998	gen 98
Feb 1, 1998	Feb 28, 1998	MONTH	febbraio 1998	feb 98
Mar 1, 1998	Mar 31, 1998	MONTH	marzo 1998	mar 98
Apr 1, 1998	Apr 30, 1998	MONTH	aprile 1998	apr 98
May 1, 1998	May 31, 1998	MONTH	maggio 1998	mag 98
Jun 1, 1998	Jun 30, 1998	MONTH	giugno 1998	giu 98
Jul 1, 1998	Jul 31, 1998	MONTH	luglio 1998	lug 98
Aug 1, 1998	Aug 31, 1998	MONTH	agosto 1998	ago 98

6. To persist the generated translations, click **Commit.**

7. To implement your changes, stop, and then restart, the PPM Servers.

   For information about how to stop and start PPM Servers, see *Starting and Stopping the PPM Server* on page 95.

For information about how to use the `kGenFiscalPeriods.sh` script to create periods for languages installed on a PPM Center instance, see the *Generating Fiscal Periods* guide. For information about using multiple languages on a single instance of PPM Center, see the *Multilingual User Interface Guide.*

Return to *Administration Tools in the Administration Console*.

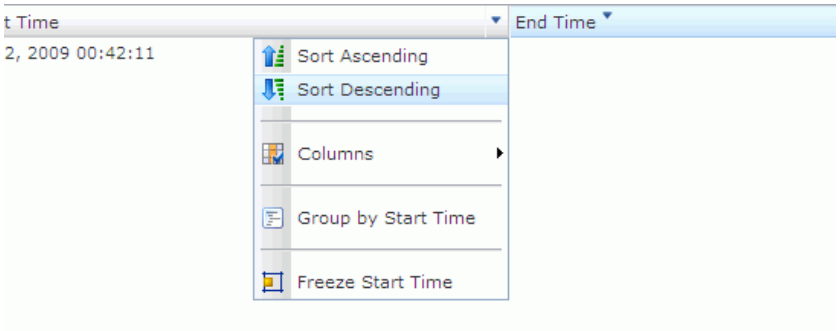
Changing the Display of Data in Administration Console Tables

This section addresses the various ways you can arrange the data displayed in the right panel of the Administration Console.

Changing Sort Order

To change the sort order of the values in a column in the Administration Console:

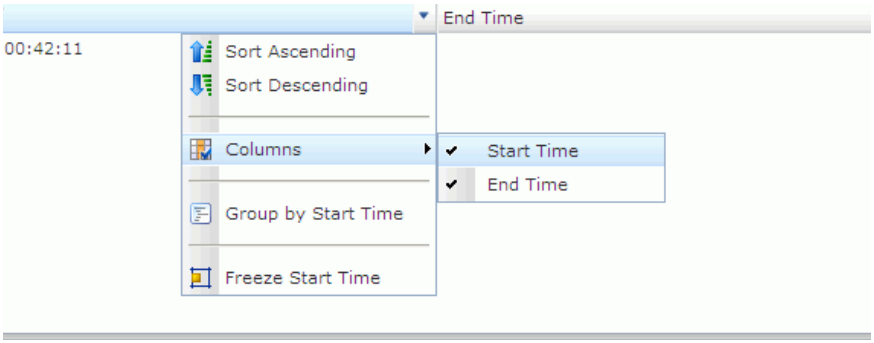
- Right-click the column heading, and then select **Sort Ascending** or **Sort Descending.**



Toggling Column Display

To toggle the display of the columns in the right panel:

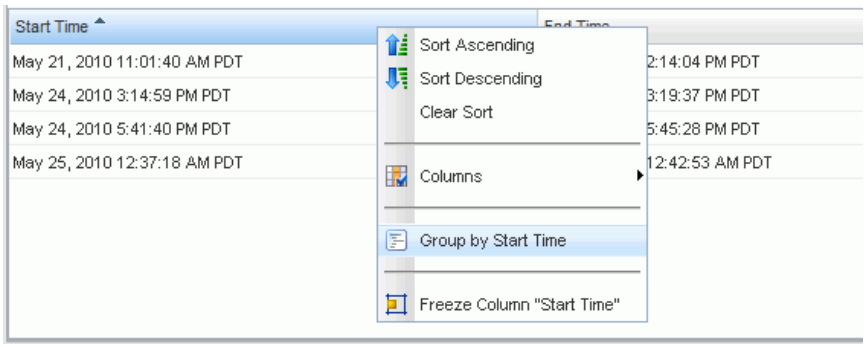
- Right-click any column heading, select **Columns,** and then select (or clear) a column heading in the shortcut menu.



Grouping Displayed Data

To group displayed data based on the dimension in the heading:

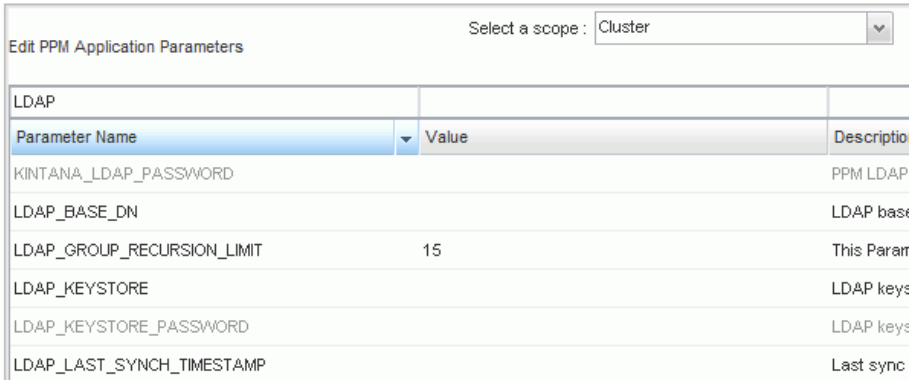
- Right-click the column heading, and then select **Group by *<Heading_Name>*** from the list.



Filtering Displayed Data

To filter displayed data based on a character string in parameter names, assigned values, or in descriptions:

- Place your cursor in the filter field above a column heading, and then type the text for the filter.

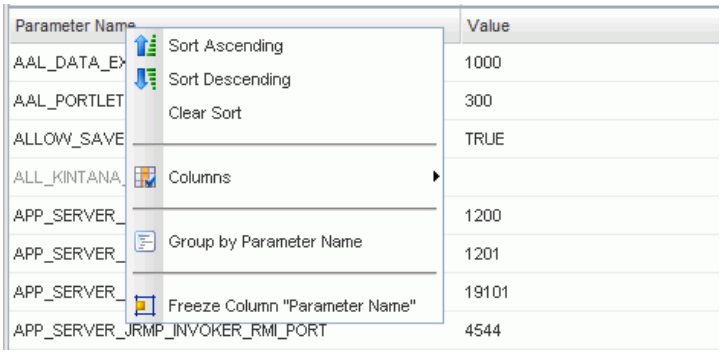


The table displays all data that include the string you specified (specific to the column).

Freezing Column Width

To freeze any column so that its width does not change if you resize the window:

- Right-click the column heading, and then select **Freeze Column "*<Column_Name>*"** from the shortcut menu.



Return to *Administration Tools in the Administration Console*.

Server Tools In the PPM Workbench

The following sections provide information about the administration tools you can access through the PPM Workbench:

- *Access Grants Required to Use Server Tools*
- *Accessing the PPM Workbench Server Tools*
- *Running Server Reports from the Admin Tools Window*
- *Running Server Reports from the Command Line*

Access Grants Required to Use Server Tools

Table 7-2 lists the names and descriptions of the three access grants that give users various levels of access to the Server Tools window.

Table 7-2. Server tools access grants

Access Grant	Permissions
Sys Admin: View Server Tools	Lets the user view the Admin Tools and SQL Runner windows in read-only mode.
Sys Admin: Server Tools: Execute Admin Tools	Lets the user run server reports in the Admin Tools window and view the SQL Runner window in read-only mode.
Sys Admin: Server Tools: Execute SQL Runner	Lets the user run SQL queries in the SQL Runner window and view the Admin Tools window in read-only mode.

For more information about security groups and access grants, see the *Security Model Guide and Reference.*

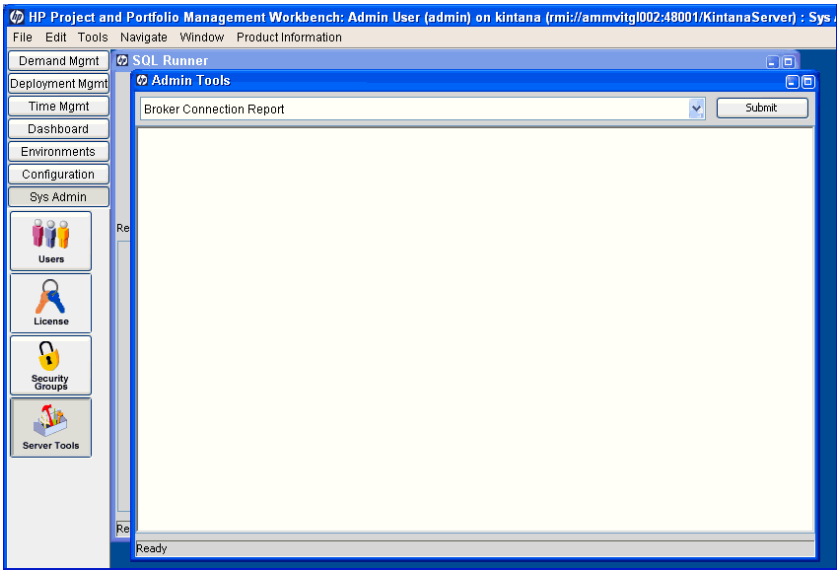
Accessing the PPM Workbench Server Tools

To access the server tools in the PPM Workbench:

1. Log on to PPM Center.
2. On the **Open** menu, click **Administration > Open Workbench.**

   The PPM Workbench opens.
3. On the shortcut bar, click **Sys Admin > Server Tools.**

   The Admin Tools and the SQL Runner windows open.



Return to *Server Tools In the PPM Workbench*.

Running Server Reports from the Admin Tools Window

Use the Admin Tools window to run server reports such as Server Status Report and Cache Manager Statistics. *Table 7-3* lists the server reports that you can generate from the Admin Tools window.

Table 7-3. Server reports (page 1 of 3)

Report Name	Description
Broker Connection	Information about open database pool connections, organized by connection ID.
Broker In Use Sessions	Information about database pool connections in use, organized by user. If the server parameter `DB_SESSION_TRACKING` is set to `true`, this report also shows stack traces of where the connection is allocated.
Broker Performance	Statistics on database connection usage in the connection pool, to help assess system performance. For performance reasons, the PPM Server holds a connection pool to the database and reuses these connections for accessing the database. Prepared statements created within a connection are also held open in a cache. If the PPM Server cannot allocate more connections, threads that need to access the database might need to wait for a connection. This report also shows: • Number of threads waiting for connections • Average duration threads had to wait for connections • Percentage of threads that had to wait for connections • Total number of connection requests, and if JDBC logging is enabled • Statement cache hit rate percentage (over the last 100 statements)
CacheManager Sizes	Displays the number of objects in the cache of each entity, the total cache size (in KB), and the average size of each cached object type.

Table 7-3. Server reports (page 2 of 3)

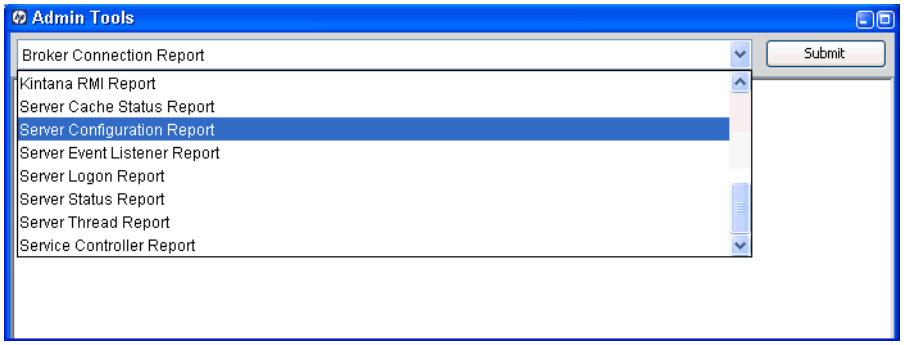
Report Name	Description
CacheManager Statistics	Displays useful statistics on the caching behavior of each cacheable entity in PPM Center, including: • Hits, misses, and hit rate • Number of cache flushes (broken down by the categories “old,” “idle,” “reclaimed,” and “max cache size reached”) • Average load time • Cached object count and maximum idle time
Client Font	All supported fonts for the PPM Center installation.
Client Property	Details about the environment of the client computer currently running the PPM Workbench.
Client Time Zone	All time zones recognized by the client.
Execution Dispatcher Manager	Batch executions in progress.
Execution Dispatcher Pending Batch	Batches pending execution due to the lack of available execution manager threads.
Execution Dispatcher Pending Group	Batches pending group execution (batches that are grouped together) due to the lack of available Execution Manager threads.
Installed Extensions	Displays the names and versions of HP Deployment Management Extensions installed (if any).
JVM Memory	Free and total memory in the PPM Server JVM.
Kintana RMI	All RMI connection threads.

Table 7-3. Server reports (page 3 of 3)

Report Name	Description
Server Cache Status	Shows the following cache information: • Cached entities • Number of units that can be cached • Number of free units • The number of hits and misses, and the miss rate • Number of entities swapped • Amount of memory taken up by the cache **Note:** Although this report displays information that is similar to the that displayed in the CacheManagerStatistics report, the data is for a different set of cached objects.
Server Configuration	All server parameters in effect for each of the active servers. Includes parameters not specifically set in the `server.conf` file.
Server Event Listener	Event messages that the PPM Server can send to the client.
Server Logon	Information about all users logged on to the PPM Server(s) and logon information such as IP address and idle time. This information is used to determine PPM Server load. If server clustering is used, this report provides a picture of load distribution.
Server Status	Status information about PPM Server(s): • Whether the server is available and its start time • Length of time the server has been available • Number of users logged on to the server • Number of users active during the last minute You can also use the Administration Management Console to view the status of PPM Servers. For information about the Administration Management Console, see *Using the Administration Console to View PPM Server Status* on page 243.
Server Thread	Information about running threads within a PPM Server(s). This information is used to determine which services are running. If a server cluster is used, this report also provides information about which server is running these services.

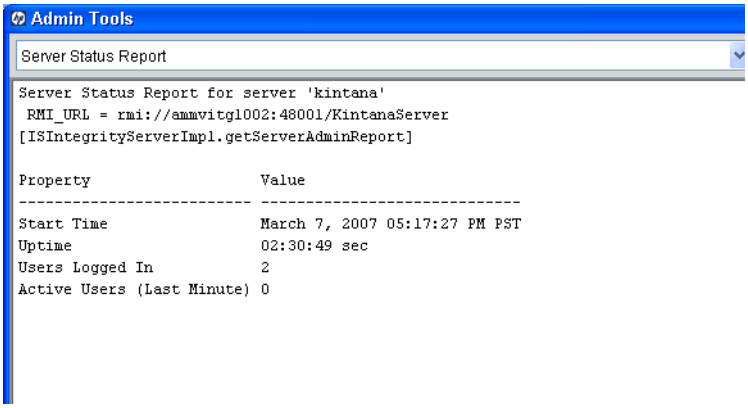
To select and run a server report:

1. In the expanded report list, select a report.



2. Click **Submit.**

   The Admin Tools window displays the output of the selected report.



If you run a report on a PPM Center instance that supports multiple languages, then the resulting report is generated in the language you selected at logon (your *session* language). Because the report is only generated once, the language used to display the contents does not change, and any user who later views the report sees it in its original language. For information about multilingual support in PPM Center, see the *Multilingual User Interface Guide.*

Running Server Reports from the Command Line

You can also run server reports directly from a command line on the PPM Server using the `kRunServerAdminReport.sh` script, which is located in the `<PPM_Home>/bin` directory. For more information about the `kRunServerAdminReport.sh` script, see *kRunServerAdminReport.sh* on page 501.

Return to *Server Tools In the PPM Workbench*.

Running SQL Statements in the SQL Runner Window

You can use the SQL Runner window to run database queries directly against the PPM Center database schema using the PPM Workbench instead of using an external program such as SQL*Plus. One benefit of using SQL Runner is that you can gain access to the database directly, without having to submit the database password. Developers and administrators can also use the SQL Runner window to test custom validations and request rule SQL, among other things.

To run an SQL statement from the SQL Runner window:

1. If the Admin Tools window hides the SQL Runner window, minimize it.
2. In the **SQL Statement** box, type the SQL statement to run.

   Make sure that your SQL statement does not end with a semicolon (;).

3. To run the SQL statement, click **Run SQL.**

   The SQL Runner window displays the list of results in the table below the SQL statement. It also displays timing information such as how long the statement took to run, and how much of that time was spent in the database.

4. To view the results as text, click **Open As Text.**

Table 7-4 lists the controls in the SQL Runner window.

Table 7-4. Controls in the SQL Runner window

Control Name	Control Type	Description
SQL Statement	Text box	Use this box to type an SQL query for running and testing purposes. **Note:** Make sure that you do not include a semicolon (;) at the end of your SQL statement.
Server Roundtrip	Read-only text box	Amount of time (in milliseconds) spent sending the SQL statement out to the network and back. Used to show network latency and performance.
SQL execution	Read-only text box	Amount of time (in milliseconds) the database spent actually executing the SQL statement. Use the displayed information to tune validations or write complex statements to address performance concerns.
ResultSet Extraction	Read-only text box	Amount of time (in milliseconds) that the server spent processing the SQL statement results.
Total time	Read-only text box	Total amount of time (in milliseconds) spent running the SQL statement.
Run SQL	Button	Runs the SQL statement displayed in the **SQL Statement** box.
Clear	Button	Clears the window.
Ping Server	Button	Tests the connection speed between the client and the PPM Server.
Ping DB	Button	Tests the connection speed between the client and the database (through the PPM Server).
Open As Text	Button	Opens results in a text window. You can cut and paste information from this window.

Return to *Server Tools In the PPM Workbench*.

Running an SQL Script with SQL*Plus on a Windows System

If your PPM Center instance is running on a Windows system, and you are using the SQL*Plus utility to run an SQL script, the utility “expects” to get the exact number of parameters defined in the script. Some versions of SQL*Plus ignore null command-line parameters and get hung up waiting for missing parameter values.

Example

In the following line, the second parameter is null. But, because SQL*Plus is a command-line utility, it waits for the user to input the second parameter value.

```
ppm/ppm@ppm10a @somescript.sql "Y" ""
```

To work around this problem, add the following to the `server.conf` file:

```
SQLPLUS_VERSION=<SQL_Plus_Version_Number>
```

A valid version number is `90101`.

Return to *Server Tools In the PPM Workbench*.

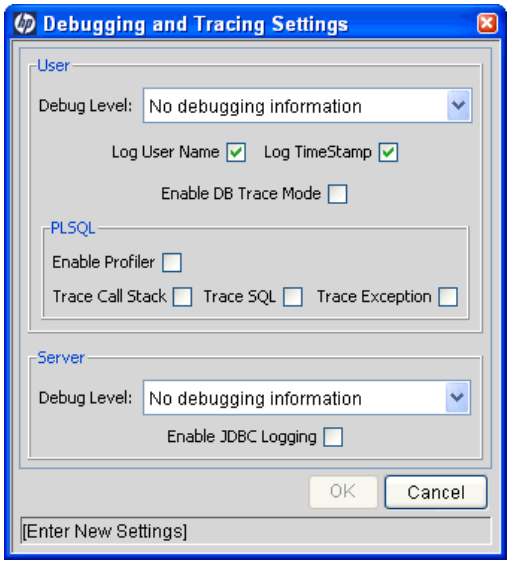
Setting Debugging and Tracing Parameters

You use the Debugging and Tracing Settings dialog box to set debugging and tracing parameters at both the user and server levels.

To open the Debugging and Tracing Settings dialog box:

1. Log on to PPM Center.
2. On the **Open** menu, click **Administration > Open Workbench.**

3. On the **Edit** menu, click **Debug Settings.**



4. To override the default debug level set for your PPM Center sessions, from the **Debug Level** list in the **User** section, select a different value.

The **Debug Level** list values map to `DEFAULT_USER_LOGGING_LEVEL` values in the `server.conf` file as follows:

- **No debugging information** is equivalent to the parameter value `ERROR`. Only errors are logged.
- **Normal debugging information** is equivalent to the parameter value `INFO`. Errors and information that describes the normal tasks that the running server is performing are logged.
- **Maximum debugging information** is equivalent to the parameter value `DEBUG`. This setting provides the most logging information. In addition to the normal debugging information, information is also logged for various server functions.

  This additional debugging information can be useful for troubleshooting any problems you encounter in PPM Center. If a problem arises, you can set the debug level to **Maximum debugging information,** perform the

problematic action again, and then check the server logs for information that can help resolve the issue.

Make sure that you do not to leave the server running in debug mode for too long. A large volume of extra information is written to the logs, taking up disk space much more quickly than during normal operation. The extra logging overhead can affect system performance.

Log User Name Setting

If you want your user name written into the log for each line of debugging text that corresponds to actions you have performed, select this checkbox. This can be helpful if you need to sift through the server logs to find information relevant to your user session. (The **Log User Name** checkbox corresponds to the `ENABLE_SQL_TRACE` configuration parameter.)

Log TimeStamp Setting

If you want a timestamp written into the log for each line of debugging text that corresponds to actions you have performed, select this checkbox. The timestamp can help you locate information in the server log files about events that occurred at a specific time, or to determine how much time elapsed between specific logged statements.

Bear in mind that including the timestamp adds text to each logged statement. This bloats the log file and can make it more difficult to read. (The **Log TimeStamp** checkbox corresponds to the `ENABLE_TIMESTAMP_LOGGING` parameter in the `server.conf` file.)

Enable DB Trace Mode Setting

To enable the SQL trace facility during your PPM Center session, select the **Enable DB Trace Mode** checkbox. This facility ensures that performance statistics for all SQL statements that you run are placed into a trace file. (The **Enable DB Trace Mode** checkbox corresponds to the `ENABLE_SQL_TRACE` server configuration parameter.)

PL/SQL Settings

The **PLSQL** field provides the following Procedural Language/Structured Query Language (PL/SQL) options:

- Select the **Enable Profiler** checkbox to profile the run-time behavior of the PL/SQL code that PPM Center applications use by calling the Oracle-supplied PL/SQL package `DBMS_PROFILER`.

You must set up the PL/SQL package. For an example of how to do this, see *Example of how to set up the Oracle profiler:* on page 272.

The profiling information is logged in a JDBC log file in the PPM Center `log` directory. Enabling the profiler can help you to identify performance bottlenecks.

Because running the `DBMS_PROFILER` package might slow system performance and reduce storage space, HP recommends that you use it only for debugging.

Example of how to set up the Oracle profiler:

```
CONNECT sys/password@service AS SYSDBA
@$ORACLE_HOME/rdbms/admin/profload.sql

CREATE USER profiler IDENTIFIED BY profiler DEFAULT
TABLESPACE users QUOTA UNLIMITED ON users;
GRANT connect TO profiler;

CREATE PUBLIC SYNONYM plsql_profiler_runs FOR
profiler.plsql_profiler_runs;
CREATE PUBLIC SYNONYM plsql_profiler_units FOR
profiler.plsql_profiler_units;
CREATE PUBLIC SYNONYM plsql_profiler_data FOR
profiler.plsql_profiler_data;
CREATE PUBLIC SYNONYM plsql_profiler_runnumber FOR
profiler.plsql_profiler_runnumber;

CONNECT profiler/profiler@service
@$ORACLE_HOME/rdbms/admin/proftab.sql
GRANT SELECT ON plsql_profiler_runnumber TO PUBLIC;
GRANT SELECT, INSERT, UPDATE, DELETE ON plsql_profiler_data
TO PUBLIC;
GRANT SELECT, INSERT, UPDATE, DELETE ON plsql_profiler_units
TO PUBLIC;
GRANT SELECT, INSERT, UPDATE, DELETE ON plsql_profiler_runs
TO PUBLIC;
```

- Select the **Trace Call Stack, Trace SQL,** and **Trace Exception** checkboxes to enable the Oracle `DBMS_TRACE` package functionality that the PL/SQL programs (used by PPM Center applications) use.

  The output of the profiling information is saved to a JDBC log file in the `<PPM_Home>/server/<PPM_Server_Name>/log` directory.

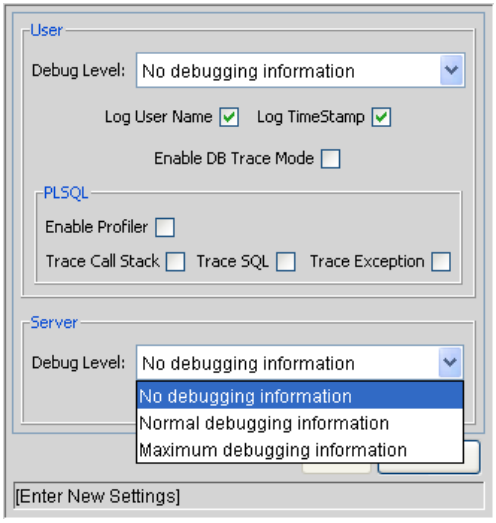
Because running the `DBMS_TRACE` package can have a negative effect on system performance and storage space, use it only for debugging.

Server Settings

To override the default logging level for the entire PPM Server, and not just your user session:

1. Under **Server,** in the **Debug Level** list, select one of the following.

The following settings correspond to the settings for the `DEFAULT_SERVER_LOGGING_LEVEL` server configuration parameter. The value names, however, are different.



- **No debugging information** is equivalent to the `DEFAULT_SERVER_LOGGING_LEVEL` parameter value `ERROR`. Only errors are logged.
- **Normal debugging information** is equivalent to the parameter value `INFO`. Errors and information that describes the normal tasks that the running server is performing are logged.

- **Maximum debugging information** is equivalent to the parameter value `DEBUG`. This setting provides the most logging information. In addition to the normal debugging information, information is also logged for various server functions.

  This additional debugging information can be useful when troubleshooting any problems you encounter in PPM Center. If a problem arises, you can set the debug level to **Maximum debugging information,** perform the problematic action again, and check the server logs for information that can help resolve the issue.

For more information about the `DEFAULT_SERVER_LOGGING_LEVEL` parameter, see Appendix A, *PPM Center Configuration Parameters,* on page 389.

2. To have the PPM Server(s) maintain a Java Database Connectivity (JDBC) log file, select the **Enable JDBC Logging** checkbox.

Return to *Server Tools In the PPM Workbench*.

System Logging in PPM Center

Every error message logged in PPM Center includes a unique identifier that you can use to locate the corresponding error in the log file.

.

Error messages are displayed in users' session languages, but log file content is not. For information about session and system languages, see the *Multilingual User Interface Guide.*

The following six types of exceptions occur in PPM Center:

- User errors.
- Internal errors
- Warnings
- Informational
- Status advisories
- Questions

Only messages for internal errors display the correlation information.

Service and context information are placed in the log messages based on the values of two logging parameters in the `logging.conf` file, which are described in the following sections.

Context Option Logging Parameter

You can use the context option parameter (`com.kintana.core.logging.context.option`) to specify extra information to include in server exception logs. The possible values are listed in the following table.

Value (bitwise combination value in binary	**Additional Information Logged**
0 (default)	None
001	Context information, if provided
010	All of the stack trace for log messages, including those without exception
3	All

If bit 1 is set to 1 (001), then the server logs include any exception context information available. If bit 2 is set to 1 (010), then the server logs include stack trace for log messages, including messages without an exception. The combination of these bits determines the overall setting. If all bits are set (value 3), then all details are logged.

Redirecting Log File Output

If you need to direct log output to a specific log file, and not to server log and console, you can do so using the logging configuration parameters. For example, the activity monitor and Background Services monitor log content to the `thresholdLog.txt` file. The content is not shown on the server log or in the console.

Class Filters Logging Parameter

You can use the class filters parameter (`com.kintana.core.logging.class.filters`) to specify the class names to include in the stack trace (substring of stack trace classname, including packages). To reduce the log file size, PPM Center uses this parameter value to filter out the classes that are of no interest in stack traces.

If you specify multiple classes or packages, use commas to separate them. If the full class name in a stack trace contains one of the specified classes or package names, then that line is preserved. For example, if the value is set to `com.kintana,com.mercury,` then any class names that contain the `com.kintana` or `com.mercury` strings are kept.

The number of traces filtered out is added to server logs after the stack trace. The `com.kintana.core.logging.class.filters` parameter has no default value. If you do not set a value, no classes are filtered out of the stack trace.

For descriptions of the parameters in the `logging.conf` file (located in the `<PPM_Home>/conf` directory), see *Logging Parameters* on page 469.

Log Levels for the install.sh Script

The log for the `Install.sh` script (`ppm_install.log`) uses the default INFO log level.

The possible log levels are as follows:

- **ERROR.** Print only error log (not recommended)
- **INFO.** Print error and information log (default)
- **DEBUG.** Print error, information and debug log (used in debugging)

Maintaining Log Files

The PPM Server generates log files in the file system. Depending on the type of log file, certain maintenance practices should be employed to maintain the file system. The following sections provide maintenance recommendations for each type of log file.

Server Log Files

Server log files are stored in the `<PPM_Home>/server/<PPM_Server_Name>/logs` directory. Server log files are named `serverLog.txt` and `serverLog_timestamp.txt`. The log timestamp setting (see *Log TimeStamp Setting on page 271*) uses the format `YYYYMMDD_HHMMSS` for the date and time the log was rotated.

Active PPM Servers log their output to the `serverLog.txt` file. The `serverLog_timestamp` files are archived versions of the `serverLog.txt` file. The size of these old log files are determined by the `ROTATE_LOG_SIZE` server parameter in the `server.conf` file. This parameter may be set to any value (in kilobytes) to control the rotation. A high value results in fewer but larger log files.

Generally, server log files are required only when contacting HP Software Support to resolve server issues. In most cases, it is safe to delete these log files on a regular basis.

The following parameters determine the data volume to be written to the logs by the server:

- `DEFAULT_SERVER_LOGGING_LEVEL`
- `DEFAULT_USER_DEBUG_LEVEL`
- `RMI_DEBUGGING`

In the `server.conf` file, set these parameters to their default values:

```
com.kintana.core.server.SERVER_DEBUG_LEVEL=none
com.kintana.core.server.DEFAULT_USER_DEBUG_LEVEL=none
com.kintana.core.server.RMI_DEBUGGING=false
com.kintana.core.server.ENABLE_LOGGING=true
```

By setting these parameters to their default settings, only critical error events are written to the server logs. This decreases the number of server logs generated in the file system, thereby improving system performance.

If the server experiences technical difficulties or server logs are required by HP Software Support, increase the debug level.

Unless instructed otherwise by HP Software Support, always set the `RMI_DEBUGGING` parameter to `false`.

To change the `USER_DEBUG_LEVEL` parameter dynamically at runtime, change the `DEFAULT_USER_DEBUG_LEVEL` parameter in the **Edit > Debug Settings** screen group in the PPM Workbench interface. You can also retrieve current server settings by accessing the Server Tools window and running the Server Configuration report.

Unless instructed by HP Software Support, do not run a production server with the debug levels set to `Maximum`. This can generate very large log files in the file system that could degrade system performance.

Enabling HTTP Logging

Do not enable HTTP logging if you use an external Web server.

To enable HTTP logging:

1. Stop the PPM Server.
2. Set the `ENABLE_WEB_ACCESS_LOGGING` server configuration parameter to `true`.
3. Run the `kUpdateHtml.sh` script.
4. Start the server.

The internal Web log is saved in NCSA Common format.

```
host rfc931 username date:time request statuscode bytes
referrer user_agent cookie
```

Example

```
127.0.0.1 - - [11/Jan/2008:1908:16 +0000] "GET/ppm/web/knta/
global/images/date_time.gif HTTP/1.1"200 155 "http://
localhost:8080/ppm/web/knta/crt/RequestCreateList.jsp"
"Mozilla/4.0 (compatible; MSIE 6.0; Windows; .NET CLR 1.0.3705;
.NET CLR 1.1.4322)" JSESSIONID=5pk1oof3fd65q
```

Report Log Files

Report execution log files are stored in the `<PPM_Home>/logs/reports` directory. Report execution log files are named `rep_log_ID.html`. The report log ID setting corresponds to the report submission ID.

Use report execution log files to determine why a report executions failed or took too much time to complete.

These log files are not purged automatically. Generally, report log files are required only to debug timely report requests. In most cases, it is safe to delete these log files on a regular basis.

Execution Log Files

During normal package and request processing, execution log files are generated:

- For workflow steps running as `EXECUTE_OBJECT_COMMANDS` or `EXECUTE_REQUEST_COMMANDS`
- When resolving a validation defined using command execution logic

Execution log files from these executions are stored in the following directories:

- `<PPM_Home>/logs/PKG_Package_ID`
- `<PPM_Home>/logs/REQ_Request_ID`
- `<PPM_Home>/logs/VAL_Validation_ID`

If disk space becomes limited over time, you might need to purge or archive these log files. If the log files are deleted, the detailed execution logs are no longer available for a package or request.

Execution Debug Log Files

If the `USER_DEBUG_LEVEL` or `SERVER_DEBUG_LEVEL` parameter is set to `HIGH`, additional execution debugging data is written to the execution debug log file. This file is named `exe_debug_log.txt` and is located in the `<PPM_Home>/logs/` directory.

If the server is running with full debugging enabled, this file grows over time. Generally, execution debug log files are required only by HP Software Support to debug the execution engine. In most cases, it is safe to delete these log files on a regular basis.

Temporary Log Files

Various other files generated in the `<PPM_Home>/logs/temp` directory are stored for temporary purposes. Unless requested otherwise by HP Software Support, you can delete these log files on a regular basis.

Periodically Stopping and Restarting the Server

The PPM Server generally requires very little maintenance. To help ensure your system operates smoothly, HP recommends that you stop and restart the PPM Server(s) once a month.

If your PPM Center instance includes multiple nodes in a cluster configuration, you must start these nodes one at a time. Make sure that you wait until each node is fully started before you start the next node.

For information about starting and stopping the PPM Server, see *Starting and Stopping the PPM Server* on page 95.

Maintaining the Database

Many IT departments have a policy of periodically changing the passwords of their database schemas. This section covers common topics related to maintaining the Oracle database that is part of PPM Center.

Changing PPM Center Data

Updating PPM Center master data directly in the database can cause various errors to occur. HP highly recommends that you not make changes directly to the PPM Center database, and instead use the PPM Center user interface to make changes.

If you absolutely must update the database directly, it is important that you understand the underlying data model design before you update the tables and views associated with the Multilingual User Interface (MLU). Before you update MLU views or tables, make sure that your Oracle NLS_LANG parameter is set to the same language as your PPM Center instance. As always, make it a point to check the data in the views and tables before you commit changes to the database. (For information about the PPM Center data model, see the *Data Model Guide*.)

Changing the Database Schema Passwords

If you must change the PPM Center database schema passwords, be sure to change them both in the database and in the `server.conf` file. Before you change all the database schema passwords, consider the following:

- Check your environment definitions to determine whether any contain a password that is to be changed. You can use the tool `<PPM_Home>/bin/kEnvUpdatePassword.sh` to automatically change all occurrences of a specific password for a particular host and user name.

  This functionality is also available from the **Environments** section of the PPM Workbench. (Open an environment on the Environment page, and then, on the **Environment** menu, click **Update Password.**)

  - Check both server and client passwords, as well as database passwords.
  - Check passwords associated with application codes.

- Although it is not a recommended practice, you can hard-code passwords into commands in workflow steps, requests, and object types.
- There is no need to change commands that use tokens for passwords (that is, `SOURCE_ENV.DB_PASSWORD`), as long as the password is changed in the respective environment definitions.

To change the PPM Center database schema passwords:

1. Make sure that all users are logged off the system.
2. Stop the PPM Server. (For information about how to stop PPM Servers, see *Starting and Stopping the PPM Server* on page 95.)
3. Change passwords, as necessary in the database.
4. To change the passwords in the `server.conf` file, run the `kConfig.sh` script to set the `DB_PASSWORD`, `CONC_REQUEST_PASSWORD`, and `RML_PASSWORD` server parameters.

   When changing the passwords, do not edit the `server.conf` file directly. To encrypt password values correctly, use the `kConfig.sh` script.

5. Restart the PPM Server.

Maintaining Temporary Tables

The PPM Server uses several tables for temporary storage during processing (for example, during package migration) for:

- Logon attempts
- Debug messages
- Commands and parameters

PPM Server uses a set of services to monitor and clean up these temporary tables. Make sure the cleanup parameters (described in *Cleanup Parameters* on page 304 and in Appendix A, *PPM Center Configuration Parameters,* on page 389) are set so that the temporary tables do not use too much database space.

KNTA_LOGON_ATTEMPTS Table

The `KNTA_LOGON_ATTEMPTS` table contains information about attempts to log on to the PPM Server during the previous 14 days. This information includes:

- `USER_ID` of users who attempted to log on
- Status (success or failure) of each logon attempt
- Messages generated during the logon attempt

The `KNTA_LOGON_ATTEMPTS` table is only for auditing purposes. The PPM Server does not require the data to function.

If logon attempts succeed, the records for those most of those attempts are purged. However, the last successful logon based on a combination of `USER_ID` and IP address is retained.

If a logon attempt fails, the corresponding record remains in the table for future reference. You must delete the failed logon attempt records manually. the record of the last successful logon attempt also remains in the `KNTA_LOGON_ATTEMPTS` table.

The data is automatically purged after the time interval specified by the `DAYS_TO_KEEP_LOGON_ATTEMPT_ROWS` server parameter setting.

KNTA_DEBUG_MESSAGES Table

The `KNTA_DEBUG_MESSAGES` table contains any debugging text that HP PL/SQL database packages generate. After you analyze this data, you can safely purge it. The PPM Server purges this data automatically at the frequency determined by the `HOURS_TO_KEEP_MESSAGE_ROWS` server configuration parameter setting.

Backing Up PPM Center Instances

Backing up a PPM Center instance involves backing up both the file system and the database schema. HP stores all PPM Center configuration and transaction data in its associated database schema.

Because this information is so important, HP also recommends that you back up the database schema daily. You can use the Oracle export command to perform the backup, or use the hot backup procedure, which does not require that you shut down the PPM Server. For information about how to export a database schema, see your Oracle database documentation.

HP recommends that you back up the `<PPM_Home>/logs` directory daily. This directory contains transactional history files for each migrated package or request.

Before you make critical changes to PPM Center, perform a full backup of the database schema and complete `<PPM_Home>` directory.

It is not necessary to back up registry settings.

Checking PPM Center License Status

You can use the license reader tool to access information related to licenses on your organization's PPM Center instance, without having to restart the PPM Server. This simple utility reads the encrypted license file and provides the following licensing information:

- Which PPM Center products are licensed for use on your instance
- IP address of the licenses machine
- Expiration dates for licenses
- Number of licenses available for different PPM Center modules.

➤ You cannot use this script to modify the license information, only to read it.

To use the license reader, run the following command:

```
kLicenseReader.sh [-filename <License_File_Name>]
[-filepath <License_File_Path>] [-help]
```

If you do not specify the file name, the license reader uses the default file name `license.conf`. If you do not specify the file path, the license reader uses the default file path `<PPM_Home>/conf`.

8 Improving System Performance

This chapter provides information about how to identify and correct performance problems on your PPM Center system, as well as what you can do to improve system performance.

For more information on improving performance, see the *PPM Center Performance Best Practices Guide.*

Identifying Performance Problems

This chapter provides information about how to isolate performance problems, collect statistics about the database schema, and troubleshoot performance problems. For detailed information on how to tune your PPM Center instance to maximize performance, see the *Performance Best Practices Guide.*

Isolating Performance Problems

The section titled *Configuring or Reconfiguring the Database* on page 119 and Appendix A, *PPM Center Configuration Parameters,* on page 389 contain information on the initial settings that HP recommends for the Oracle database and PPM Server. If performance slows after these settings are in place, use the methods outlined in the flowcharts shown in *Figure 8-1* on page 288, *Figure 8-2* on page 289, and *Figure 8-3* on page 290 to isolate performance problems and determine how to fix them.

Figure 8-1. Identifying and addressing system performance problems

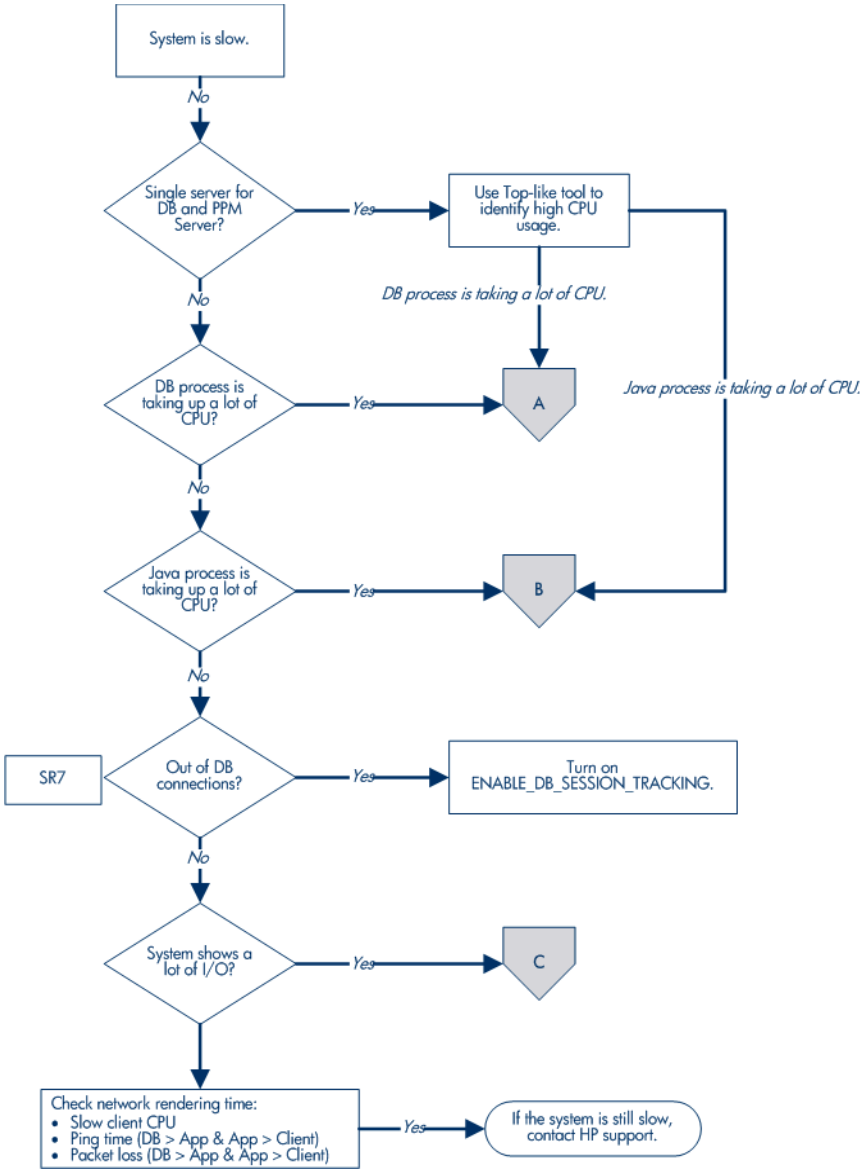


Figure 8-2. Identifying and addressing database performance problems (A)

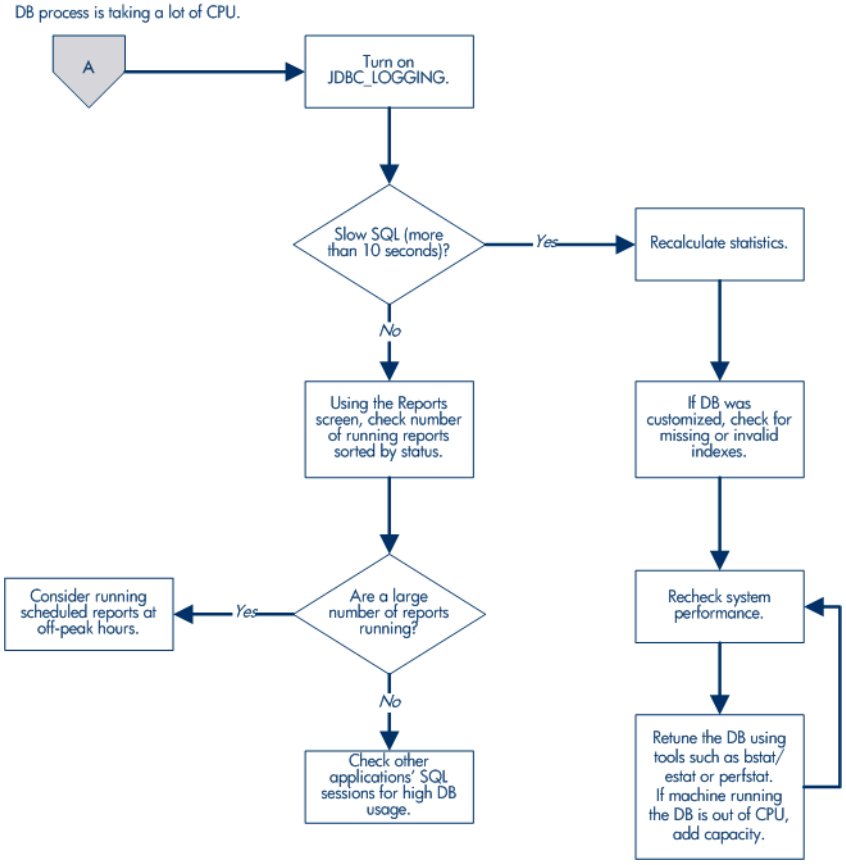


Figure 8-3. Identifying and addressing Java process performance problems (B)

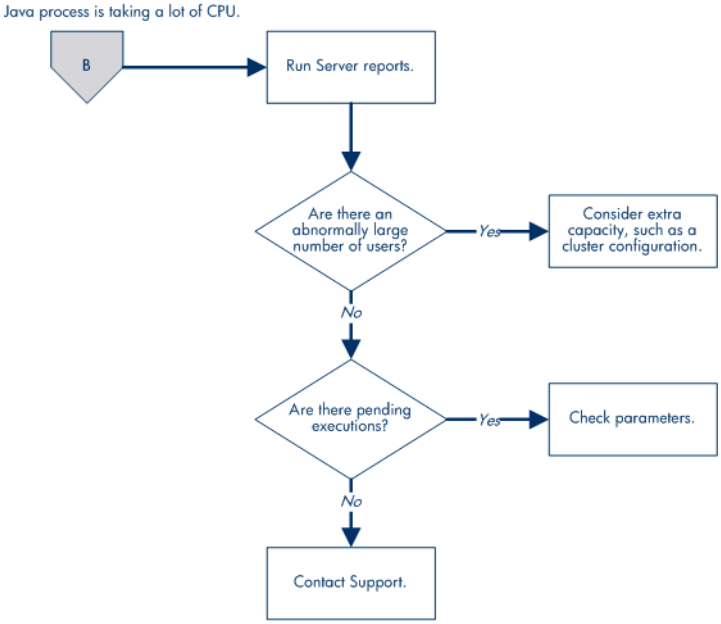
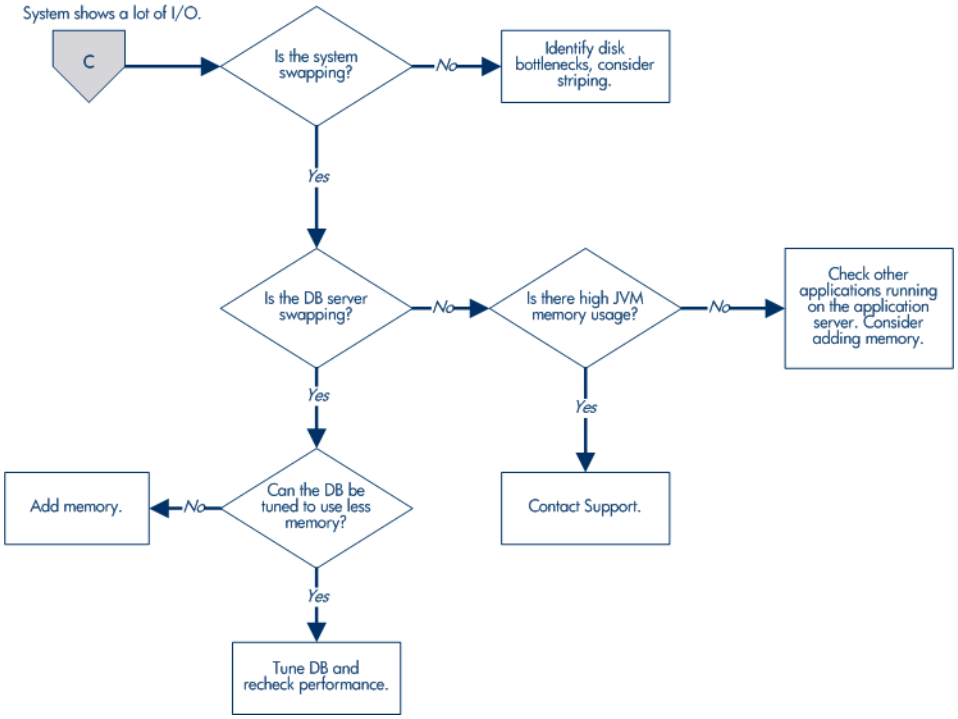


Figure 8-4. Identifying and addressing I/O performance problems (C)



Collecting Database Schema Statistics

This section provides information about how to collect statistics on the Oracle database schema.

Scheduling Oracle Statistics Calculation Service to Collect Statistics

Collect database schema statistics if you:

- Apply field-level security to a request type with existing requests
- Apply dynamic security to a workflow with existing instances
- Add field group(s) for Distributed Management Objects (DMO) or PMO
- Use Microsoft Project to import large projects or many projects

You can set Oracle Statistics Calculation service in PPM Center to automatically collect the database schema data periodically. For information on how to schedule this service, see *Enabling and Scheduling PPM Center Services* on page 239.

Using the dbms_stats Package to Collect Additional Statistics

If the statistics that the PPM Center service collects are insufficient, you can use the `dbms_stats` package to gather additional statistics. Oracle provides this package as part of the database.

To gather additional statistics, run the following script.

```
begin
dbms_stats.gather_schema_stats (ownname => <PPM_User>,
cascade => TRUE,
method_opt => 'FOR ALL COLUMNS SIZE SKEWONLY'
);
end;
/
```

You typically run the `dbms_stats` package as the SYSTEM user. To run it as a PPM Center user, grant the privilege to run the package by running the following SQL statement as the SYSTEM user from an SQL*Plus session.

```
grant execute on dbms_stats to <PPM_User>;
```

The first time you run the `dbms_stats` package, use `method_opt => 'FOR ALL COLUMNS SIZE SKEWONLY'`. After the system has been up and running for a while, use `method_opt => 'FOR ALL COLUMNS SIZE AUTO'`.

Sampling a Percentage of Data

With large databases, analysis can take up to three hours to run to completion. For large PPM Center installations, you can sample a percentage of data in each object instead of data from the entire PPM Center database schema.

Sampling a percentage of data may not be effective for small data sets. However, after the data set has grown, this method is almost as effective as calculating statistics for the entire database schema.

To calculate statistics on a percentage of the data, run the following script.

```
begin
dbms_stats.gather_schema_stats (ownname => <PPM_User>,
cascade => TRUE,
method_opt => 'FOR ALL COLUMNS SIZE SKEWONLY',
estimate_percent => <Percentage_To_Sample>
);
end;
/
```

Troubleshooting Performance Problems

This section provides information about common performance problems and how to correct them. If you are not using the default or recommended settings, reset your parameters to those values before you try other solutions.

Consider upgrading to the latest PPM Center service pack. HP has a regular and well-established service pack release cycle. Much of the development effort that goes into these service packs is focused on resolving known performance issues. Review the *Release Notes* for the latest service pack to determine whether it addresses the performance problem you are experiencing.

Scheduled Reports Do Not Run on Schedule

Problem

Although the PPM Server has capacity available, the next scheduled tasks do not start.

Possible source

All listeners on the light-weight service queue are busy running other services.

Solution

Do one of the following:

- Add another node to the PPM Server cluster with services enabled.
- Increase the value for the `LIGHT_QUEUE_MAX_CONCURRENT_CONSUMERS` parameter in the `server.conf` file for one of the nodes in the cluster.

Packages Do Not Execute

Problem

Packages do not execute.

Possible source

There are not enough execution managers available to service the packages that the system processed.

Solution

Increase the `MAX_EXECUTION_MANAGERS` server configuration parameter value. For information about this parameter, see Appendix A, *PPM Center Configuration Parameters*, on page 389.

Nightly Reports on Sunday Do Not Finish On Time, System Slows on Monday

Problem

By default, database server statistics are collected at 1:00 a.m. on Sundays. For large installations, collection take so long that it is not completed on time and system performance is slower on Monday.

Solution

Reschedule the statistics collection to a time that works better for your organization. Determine the most active system time by running the Server Logon report, which checks the number of active users. For details on how to run the report, see *Running Server Reports from the Admin Tools Window* on page 263 and *Running Server Reports from the Command Line* on page 267.

Consider using the estimate method instead of the compute method to gather statistics.

Monitor CPU use. If the system slows because of high peak load, you might require more hardware or faster hardware.

For more information about gathering statistics, see *Collecting Database Schema Statistics* on page 292.

Improving System Performance

This section provides information about how you can improve system performance. For additional information about improving performance on your PPM Center instance, see the *PPM Center Performance Best Practices Guide*.

Minimizing the Performance Impact of Running Background Services

As a system administrator, you can schedule, monitor, and distribute background services across all nodes in a PPM Server cluster. You can proactively monitor application performance to identify threats and eliminate them before system-wide performance problems or instability occur.

The following subset of PPM Center background services can be run in parallel to optimize performance and minimize bottlenecks:

- Request Status Export Service
- FX Rate Update Service
- Evaluate TM Approvers Service
- Task Actual Rollup Service.
- Financial Metrics Update Service
- Staffing Profile Period Sum Update Service
- Project Planned Value Update Service
- Resource Pool Rollup Service

These services can be run on the same or on separate server cluster nodes on two different entities, such as projects, work plans, time sheets, and so on.

For information about how you can monitor background service activity, see *Background Services Monitor* on page 318.

Recommendations for Running Background Services

Keep in mind that the flexible service framework that you get with a clustered server environment comes with some risk. Use the following guidelines in enabling and scheduling the background services on your instance:

- **Run the system with background services in an isolated environment.** For medium to large PPM Center deployments, HP recommends that you dedicate one PPM Server on a single JVM to processing PPM Center background services. This minimizes the impact that running services has on users, and enables you to better monitor background service performance. For optimal performance, set the `SERVICES_ENABLED` server configuration parameter to false to turn off services on nodes devoted to user traffic.

  By default, nodes that run background services in a server cluster environment have one heavy-service consumer and two light-service consumers. This configuration reduces both memory consumption and CPU usage.

- **Schedule PPM Center services to run when they are least likely to affect system performance.** Try to schedule background services to run during periods of low activity, such as weekends and non-working hours.

  For Information about PPM Center background services and instructions on how to schedule them, see *PPM Center Background Services* on page 232.

- **Assess and adjust the frequency with which background services are run.** Some of the services may run more often than necessary, while other services may need to be run more frequently. For PPM Servers devoted to services, schedule services to run only as often as necessary.

- **Disable unnecessary background services.** Your instance may be running more background services than you need.

  For Information about PPM Center background services and instructions on how to enable or disable them, see *PPM Center Background Services* on page 232.

- **Start by running services on a single node, and then add services nodes as required.** Although you can easily run services across multiple nodes, HP recommends that you start by running services on a single node. Later, when the workload calls for it, and after you determine that services are running correctly, you can add PPM Servers as dedicated services nodes (JVMs) that have the same configuration as the initial services node (one heavy service and two light services). It is always better to run with fewer services nodes and retain most processing capacity for end-user activity.
- **Test your solution.** As always, test your solution to determine what is optimal for your environment.

Tuning Java Virtual Machine (JVM) Performance

Because the PPM Server uses JSP, a Java compiler must be available in the environment path where the server is started.

Running in Interpreted Mode

To improve performance, the Java virtual machine (JVM) uses a just-in-time (JIT) compiler. For debugging purposes, you can disable the JIT compiler and run the JVM in interpreted mode. Exceptions that you encounter while running in interpreted mode contain line numbers that are helpful in debugging.

To run the JVM in interpreted mode, set a variable in the server environment, as follows (use the Bourne or K shell):

```
JAVA_COMPILER=None
export JAVA_COMPILER
```

To avoid performance degradation, do not run the JVM in interpreted mode for extended periods in a production environment.

Debugging

The PPM Server startup script (`kStart.sh`) contains two JVM parameters that you can use for debugging. The `kStart.sh` JVM debugging parameters are `-ms1280m` and `-mx1280m`. These specify that the JVM starts up with a heap size of 1280 MB (1.2 GB), and is limited to a maximum heap size of 1280 MB.

These settings are usually sufficient. For sites with heavy usage, however, consider overriding the default maximum heap size using the `SERVER_MAX_HEAP_SIZE` parameter in the `server.conf` file. The amount of memory required depends on factors such as cache sizes and number of Oracle connections.

After you first start the PPM Server following an installation or upgrade, the server occupies approximately 750 MB in memory. As you use the product, the cache fills up and the JSPs are loaded into memory. Over time, the system gradually uses more memory. This is normal, and memory usage levels out over time. In most cases, memory usage can increase to a maximum of 1 GB.

Setting Heap Size

If your Java program requires a large amount of memory, you may find that, at some point, the virtual machine starts to throw OutOfMemoryError instances as it attempts to instantiate objects. This can result from your program using more memory than is available. In this case, you can use command-line options to increase the heap size allocated by the Java Virtual Machine (JVM). If not specified, the heap size defaults to 1 MB, and can increase to as much as 16 MB if your program requires more memory. (To set the initial amount of memory allocated for your program, use the -Xms option.)

Tuning Server Cluster Performance

High transaction volumes and a large number of concurrent users on a PPM Server can degrade server response time. If the PPM Server is running on a multiprocessor system, spare CPU may be available, but JVM limitations can prevent the system from using the spare CPU.

In this case, consider using a PPM Server cluster. In this system configuration, multiple PPM Servers point to the same database instance and can be started

on one or more systems. In addition to added capacity, running on multiple systems increases availability.

To use your multiple-CPU system effectively, this may be necessary on a two-CPU system, and it is required on systems with more than two CPUs.

For information about how to set up a server cluster, see *Configuring a Server Cluster* on page 172.

Improving Input/Output Throughput

The distribution of input and output across multiple disks is an important factor in database performance. If consistently high input/output (I/O) occurs on one or more disks housing the database, service time on that disk degrades. To address this problem, replan the database layout to improve application performance.

You can split the PPM Center database into the following segments:

- PPM Center tables
- PPM Center indexes
- Redo logs
- Rollback tablespaces
- Temporary tablespaces
- System tablespace
- Tablespace for management and related utilities

HP recommends that PPM Center database instances with moderate transaction volume (instances with more than 5,000 requests per month) have at least four discrete disks, divided as shown in *Table 8-1*.

Table 8-1. Database disk recommendations

Disk	Recommendations for Data Placement
1	PPM Center tables
2	PPM Center indexes
3	Redo logs
4	• Rollback tablespaces • Temporary tablespaces • System tablespace • Tablespace for management and related utilities

For PPM Center database instances that have higher transaction volumes (more than 10,000 requests per month), HP recommends you do the following:

- Place each piece of the database on its own separate disk.
- Stripe the data and index tablespaces across multiple disks to provide adequate disk throughput.

For PPM Center database instances with an extremely high transaction volume (over 25,000 requests per month), move specific tables and indexes to separate tablespaces on separate disks. This provides better control and further increases available I/O throughput.

Improving Advanced Searches

PPM Center users can search for requests based on custom fields defined in request types, request header types, and user data. Users can perform advanced searches to locate requests based on information that is defined as critical to business processes.

As the number of requests logged increases, users performing advanced searches can experience slower performance. To improve performance during advanced searches, use the following guidelines:

- Specify additional request header fields in the advanced searches. Header fields are automatically indexed by PPM Center, and therefore yield faster returns.
- Add indexes to a limited number of detail fields, preferably fields that are commonly used in advanced searches. Take care not to add too many indexes, since this can affect the performance of inserts and updates to the database.
- Set the `DEFAULT_REQUEST_SEARCH_ORDER_BY_ID` server configuration parameter value to `true` to remove the sort order column on a request search. Record sorting slows performance.

- Change the value set for the `REQUEST_SEARCH_RESULTS_MAX_ROWS` server configuration parameter to restrict the maximum number of records retrieved.
- For portlet search queries, lower the value set for the `PORTLET_MAX_ROWS_RETURNED` server configuration parameter. For most portlets, 20 to 50 records is adequate. The default is 200.

Adjusting Server Configuration Parameters

This section provides information about PPM Server parameters related to system performance and usage considerations for these parameters.

Server configuration parameter fall into the following categories:

- Cleanup parameters
- Debug parameters
- Timeout parameters
- Scheduler/services/thread parameters
- Database connection parameters
- Cache parameters

Most of the parameters are defined in the `server.conf` file. For a list of PPM Server parameters, see Appendix A, *PPM Center Configuration Parameters*, on page 389. The following sections provide descriptions of the parameters in each system performance parameter category.

Cleanup Parameters

The following parameters, which are defined in the `server.conf` file, determine when the PPM Server invokes services to clean up database tables:

- `DAYS_TO_KEEP_APPLET_KEYS` determines how many days to keep applet keys in the KNTA_APPLET_KEYS table.
- `DAYS_TO_KEEP_COMMAND_ROWS` determines how many days to keep records in the prepared commands tables.
- `DAYS_TO_KEEP_INTERFACE_ROWS` determines how many days to keep records of all interfaces.
- `DAYS_TO_KEEP_LOGON_ATTEMPT_ROWS` determines how many days to keep records of all logon attempts.
- `HOURS_TO_KEEP_DEBUG_MESSAGE_ROWS` determines how long (in hours) to keep rows in the `KNTA_DEBUG_MESSAGES` table.

For descriptions of and valid settings for these parameters, see Appendix A, *PPM Center Configuration Parameters,* on page 389

Debug Parameters

Debug parameters control the debug and log output from the PPM Server. Debug parameters are either high-level or low-level.

High-Level Debug Parameters

You can change high-level debug parameters without causing system downtime on the PPM Server. Users who have the required privileges can configure these parameters by selecting **Edit > Debug Settings** from the PPM Workbench.

The high-level debug parameters are:

- `DEFAULT_USER_DEBUG_LEVEL` (defined in the `logging.conf` file) control the debugging level.
- `ENABLE_JDBC_LOGGING` (defined in the `server.conf` file) determines whether the server maintains a JDBC log file. If it is enabled, JDBC

logging records SQL runs against the database, the amount of time required to run the SQL, and the amount of time required to retrieve the results.

- `ENABLE_SQL_TRACE` (defined in the `server.conf` file) determines whether performance statistics for all SQL statements run are placed into a trace file.
- `SERVER_DEBUG_LEVEL` (defined in the `logging.conf` file) controls the verbosity of logs generated by independent server processes such as EmailNotificationAgent.

For more information about the high-level debug parameters, see Appendix A, *PPM Center Configuration Parameters,* on page 389 and *Logging Parameters* on page 469.

Low-Level Debug Parameters

Enable the low-level debug parameters only if you require debugging information for a specific area. Enabling these parameters can degrade system performance because they consume additional CPU and generate large log files.

HP strongly recommends that you consult HP Software Support before you enable low-level debug parameters.

The low-level debug parameters, which are all defined in the `logging.conf` file are:

- `ENABLE_DB_SESSION_TRACKING`
- `ENABLE_LOGGING`
- `ENABLE_TIMESTAMP_LOGGING`
- `EXECUTION_DEBUGGING`
- `JDBC_DEBUGGING`
- `WEB_SESSION_TRACKING`

For more information about low-level debug parameters, see *Logging Parameters* on page 469.

Timeout Parameters

Timeout parameters determine how long the PPM Server waits before it times out. You can set timeout values for logon sessions, command runs, and workflows.

The timeout parameters, which are all defined in the `server.conf` file, are:

- `CLIENT_TIMEOUT` determines the interval (in minutes) at which PPM Workbench sessions send a message to inform the PPM Server that the client is active.
- `DB_LOGIN_TIMEOUT` determines the duration (in seconds) for the PPM Server to keep trying to log on to the database before reporting that the database is unavailable.
- `DEFAULT_COMMAND_TIMEOUT` determines the duration (in seconds) for the PPM Server to keep trying to run commands before timing out.
- `PORTLET_EXEC_TIMEOUT` determines the duration (in seconds) after which portlets time out.
- `SEARCH_TIMEOUT` determines the duration (in seconds) after which searches time out.

Scheduler/Services/Thread Parameters

Scheduler/services/thread parameters, which are all defined in the `server.conf` file, control scheduling, services, and thread-related server activities.

The scheduler/services/thread parameters are:

- `AUTOCOMPLETE_STATUS_REFRESH_RATE` determines the frequency (in seconds) with which the command status is refreshed to provide a list of values in an auto-complete field.
- `EXCEPTION_ENGINE_WAKE_UP_CHECK_FREQUENCY` determines the interval (in seconds) that elapses before a task is verified for exceptions

- `MAX_EXECUTION_MANAGERS` determines the number of command executions that can run simultaneously. Organizations processing a high volume of packages may require a larger number of execution managers.
- `MAX_RELEASE_EXECUTION_MANAGERS` determines the number of command executions that can run in a release distribution simultaneously. Organizations that process a high package volume may require more release execution managers.
- `REPORTING_STATUS_REFRESH_RATE` determines the frequency (in seconds) with which the report status is refreshed and displayed to the user.
- `THREAD_POOL_MAX_THREADS` determines the maximum number of packages to run simultaneously within a release distribution. If a large number of packages in a distribution are processing, increase this value to improve performance.
- `THREAD_POOL_MIN_THREADS` determines the minimum number of packages to be run simultaneously within a release distribution.
- `WF_SCHEDULED_TASK_INTERVAL` establishes the frequency (in seconds) with which the PPM Server checks for pending scheduled tasks, and starts the tasks if worker threads are available.
- `WF_SCHEDULED_TASK_PRIORITY` determines the priority of scheduled tasks. Because scheduled tasks run in the background, it may be useful to run these tasks at a lower priority than the threads servicing user-oriented interactive tasks.

Database Connection Parameters

Database connection parameters relate to the management of the database connection pool that the PPM Server maintains. After the PPM Server starts, one database connection is established. Increased usage spawns additional database connections.

The database connection parameters, which are all defined in the `server.conf` file, are as follows:

- `DB_LOGIN_TIMEOUT` determines the amount of time that the PPM Server is to continue to try to log on to the database (acquire the JDBC connections that make up the connection pool) before reporting that the database is unavailable.
- `MAX_DB_CONNECTION_IDLE_TIME` determines the amount of time (in minutes) that an unused database connection is held open before it is closed and removed from the pool.
- `MAX_DB_CONNECTION_LIFE_TIME` determines the duration (in minutes) that a database session is held open before it is closed and removed from the pool. Some Oracle cleanup operations that should be run periodically occur only at the end of database sessions. Do not keep database sessions open for the life of the PPM Server.
- `MAX_DB_CONNECTIONS` determines the maximum size of the database connection pool that the PPM Server creates.
- `MAX_STATEMENT_CACHE_SIZE` determines the maximum number of prepared statements cached per database connection.

Logging Parameters

The logging parameters are in the `logging.conf` file. For information on the logging parameters that affect system performance, see *System Logging in PPM Center* on page 275. For descriptions of all logging parameters, see *Logging Parameters* on page 469.

Cleanup Services

Cleanup services determine which services the PPM Server invokes to clean up database tables. You enable (or disable) and schedule cleanup services from the Schedule Services page in the PPM Center standard interface. For instructions on how to enable and schedule services, see *Enabling and Scheduling PPM Center Services* on page 239.

Monitoring Activity in PPM Center

This section contains information about configurable monitors that capture information on user interface activity, portlets, and background services in PPM Center. These monitors capture UI activities (mainly URL requests), background service runs, and portlet activity on the PPM Server.

Action Monitor

The PPM Center action monitor tracks activity in the standard interface. To control the monitor, you use the `ENABLE_UI_MONITOR` and `UI_MONITOR_THRESHOLD` server configuration parameters. The `ENABLE_UI_MONITOR` parameter turns the monitor on (the default), and the `UI_MONITOR_THRESHOLD` parameter controls the threshold value of the action monitor, which is set in milliseconds.

Changing the Action Monitor Parameters Using the JMX Console

You can change the values of the `ENABLE_UI_MONITOR` and `UI_MONITOR_THRESHOLD` parameters using the `kConfig.sh` script or, you can change them through the JMX console without having to stop, and then start the PPM Server. Keep in mind that if you change these parameter settings through the JMX console, your changes do not persist. The next time you start the PPM Server, the parameter settings revert to the values specified for them in the `server.conf` file.

To enable or disable the UI monitor through the JMX console:

1. Go to the JMX MBean agent view in the JMX console.
2. In the **ppm.monitor** section, access the ActionMonitorFilter MBean.
3. Use the `EnableMonitor` parameter to enable or disable the UI monitor.

To change the `UI_MONITOR_THRESHOLD` value:

1. Go to the JMX MBean view in the JMX console.
2. In the **ppm.monitor** section, access the ActionMonitorTrigger MBean.

3. Change the value of the `Threshold` parameter.

> You can also use the `PersistState` parameter in the JMX console to enable or disable persisting the captured information to the log file.

Viewing the Action Monitor Information in Real Time

To see the information captured by the action monitor in real time, go to the JMX console and access the ActionMonitor MBean in the **ppm.monitor** section. The ten UI activities (URL requests) that most affect performance are listed, as well as their average, minimum, and maximum execution times.

Action Monitor Information Log

If the latency for serving the URL request exceeds the threshold value, the captured information is saved to the `thresholdLog.txt` file, which resides in the `<PPM_Home>/server/<PPM_Server_Name>/logs` directory.

The information is formatted as follows:

```
["UI",<User_Name>,"<Requested_URL>","<Execution_Time>(ms)"]
```

The information resembles the following:

```
ActionMonitorLogger:2009/01/11-23:26:-6.179 PST:
"UI",admin,"http://37.30.24.33:8080/itg/web/knta/global/
AutoCompPopup.jsp","78(ms)" ;
```

> For information on how to create a report on action monitor activity, see *Server Performance Reports* on page 314.

Portlet Monitor

The PPM Center portlet monitor tracks the load time of portlets. You can use the information it generates to determine the impact of portlet activity on system performance. To control the portlet monitor, you use the `ENABLE_PORTLET_MONITOR` and `PORTLET_MONITOR_THRESHOLD` server configuration parameters. The `ENABLE_PORTLET_MONITOR` parameter turns the monitor on (the default), and the `PORTLET_MONITOR_THRESHOLD` parameter controls the monitor threshold value, which is set in milliseconds.

Changing the Portlet Monitor Parameters Using the JMX Console

You can change the values of the `ENABLE_PORTLET_MONITOR` and `PORTLET_MONITOR_THRESHOLD` parameters using the `kConfig.sh` script or, you can change them through the JMX console without having to stop and then the PPM Server. Keep in mind that if you change these parameters through the JMX console, your changes do not persist. The next time you start the PPM Server, the settings for these parameters revert to the values specified for them in the `server.conf` file.

To enable or disable the portlet monitor through the JMX console:

1. Go to the JMX MBean agent view in the JMX console.
2. In the **ppm.monitor** section, access the Portlet Monitor Filter MBean.
3. Use the `EnableMonitor` parameter to enable or disable the portlet monitor.

To change the `PORTLET_MONITOR_THRESHOLD` value:

1. Go to the JMX MBean view in the JMX console.
2. In the **ppm.monitor** section, access the PortletMonitorTrigger MBean.
3. Change the value of the `Threshold` parameter.

You can also use the `PersistState` parameter in the JMX console to enable or disable persisting the captured information to the log file.

Viewing the Portlet Monitor Information in Real Time

To see the information captured by the portlet monitor in real time, go to the JMX console and access the PortletMonitor MBean in the **ppm.monitor** section. The ten portlets that most affect performance are listed along with their average, minimum, and maximum load times.

For information about how to improve portlet performance, see the *PPM Center Performance Best Practices Guide.*

Portlet Monitor Information Log

If the amount of time required to load the portlet exceeds the threshold value, the captured information is saved to the `thresholdLog.txt` file, which resides in the `<PPM_Home>/server/<PPM_Server_Name>/logs` directory.

For information on how to create a report on portlet monitor activity, see *Server Performance Reports*.

Server Performance Reports

You can create server performance reports that are based on action monitoring and portlet monitoring results. To do this, you first create a report type, and then create reports of that type from the PPM Center standard interface. The following sections provide instructions for performing these tasks.

Creating a Portlet Performance Report Type

To create a portlet performance report type:

1. Log on to PPM Center.
2. From the menu bar, select **Open > Administration > Open Workbench.**

   The PPM Workbench opens.
3. From the shortcut bar, select **Configuration > Report Types.**

   The Report Type Workbench opens.
4. Click **List.**
5. On the **Results** tab, scroll down to and select **(REFERENCE)Portlet Performance Report.**
6. Click **Copy.**
7. In the Copy Report Type window, do the following:
   a. In the **Report Type Name** field, type a name such as “Server Performance Report.”

      The report type you create here includes both the action performance report subtype and the portlet performance report subtype.

   b. Click the **Reference Code** field.
   c. A reference code is automatically created for the report type based on the name you typed in. You can leave this default value, or type a different reference code.

d. If you are working in a multilingual UI (MLU), select the **Copy existing translations** checkbox.

e. Click **Copy.**

8. In the dialog box that opens, click **Yes,** to open the new report type for editing.

9. Select the new report name, and then click **Open.**

   The Report Type window opens.

10. In the **Description** field, type a new report type description.

11. For **Enabled,** select **Yes.**

    The **Results** tab in the Report Type Workbench lists the new report.

Creating Server Performance Reports

To create a portlet performance report or action performance report:

1. After you have created the report type from the PPM Workbench, return to the PPM Center standard interface.

2. From the menu bar, select **Create > Report.**

   The Submit New Report page opens.

3. From the **Report Category** list, select **Administrative.**

4. From the list of Administrative reports, select the report type you created from the PPM Workbench (see *Creating a Portlet Performance Report Type* on page 314).

   The Submit Report: <*Report_Type_Name*> window opens.

5. Provide information in the fields listed in the following table.

Field (* Required)	Description
From	Use the Date Time Chooser to specify the start date and time for data to include in the report.
To	Use the Date Time Chooser to specify the end date and time for data to include in the report.
*Report Name	From this list, select one of the following: • **Portlet Performance Report** • **Page Performance Report**
Report Type	From this list, select the level of detail to include in the report. The options are: • **Summary.** Presents aggregated results on average execution time, and the maximum, minimum, and total time taken to invoke the page or portlet (portlets). • **Detail.** Presents information on the poorest performing page or portlet based on execution time. • **Summary and Detail.** Presents information on the poorest performing page or portlet, aggregated results on average execution time, and the maximum, minimum, and total time taken to invoke the page or portlets.
Top N	Use this field to specify how many of the portlet or page invocations with the worst response times to include in the report. The default is 10.
Portlet Name (Enabled for portlet performance reports only)	If you are creating a portlet performance report, use this multiselect to limit the report to one specific portlet. If you do not specify a portlet, the report includes information on all portlets in PPM Center. If you are creating a page performance report, this field in disabled.

Field (* Required)	Description
Report Period	From this list, select the value that indicates the frequency with which to run the report. The choices are **Daily** (default), **Weekly, Hourly,** and **All.**
Run Report Immediately	Select this option to run the report now (the default).
Run Report On	Select this option to run the report on a specific calendar date. If you select this option, then you must use the multiselect to specify the date and time to run the report the first time.
Repeat Every	If you select the **Run Report On** option, you can then have the report run at regular intervals by selecting this checkbox. Use the adjacent number field and list to specify the report run intervals.
Until	Use the multiselect to specify a date and time at which to stop running the report at the set interval.
Send email to	To send an notification email after the report is completed, select this checkbox. To have the notice sent to a user other than you, use the now-enabled multiselect to select the user.
Add a Notification	Click this button to open the Edit Advanced Notifications window and configure a custom notification.

6. Click **Submit.**

Background Services Monitor

The PPM Center Background Service monitor is controlled using the `ENABLE_BACKGROUND_SERVICE_MONITOR` and `BACKGROUND_SERVICE_MONITOR_THRESHOLD` server configuration parameters. The `ENABLE_BACKGROUND_SERVICE_MONITOR` parameter turns the monitor on (the default). The `BACKGROUND_SERVICE_MONITOR_THRESHOLD` parameter controls the threshold value (in milliseconds) for the monitor. If the runtime of a background service exceeds the threshold value, this is recorded in the `thresholdLog.txt` file. The following is an example of the `thresholdLog.txt` file contents:

For information on all of the background services in PPM Center, see *PPM Center Background Services* on page 232.

Changing the Background Services Parameters Using the JMX Console

You can change the values of the `ENABLE_BACKGROUND_SERVICE_MONITOR` and `BACKGROUND_SERVICE_MONITOR_THRESHOLD` parameters using the `kConfig.sh` script, or you can change them through the JMX console without having to stop and start the PPM Server. However, keep in mind that if you change them through the JMX console, your changes do not persist. The next time you start the PPM Server, the values for these parameters revert to the values specified for them in the `server.conf` file.

To enable or disable the Background Services Monitor through the JMX console:

1. Go to the JMX MBean agent view in the JMX console.
2. In the **ppm.monitor** section, access the BackgroundServiceMonitorAspect MBean.
3. Use the `EnableMonitor` parameter to enable or disable the UI monitor.

To change the UI_MONITOR_THRESHOLD value:

1. Go to the JMX MBean view in the JMX console.
2. In the **ppm.monitor** section, access the Action Monitor Trigger MBean.
3. Change the value of the `Threshold` parameter.

You can also use the `PersistState` parameter in the JMX console to enable or disable persisting the captured information to the log file.

Viewing the Background Services Monitor Information in Real Time

Administrators can view the Background Services Monitors in real time through the JMX console, and use the information to isolate performance issues in the field. To see the information captured by the Background Services Monitor in real time, go to JMX console and access the BackgroundServiceMonitor MBean in the **ppm.monitor** section. Here, all of the background services that have been executed on the system are listed. Information about the background service runs that most affect performance are listed, as are the minimum, maximum, and average execution times for these services.

Background Services Monitor Information Log

If background services activity exceeds the configured threshold value, the captured information is saved to the `thresholdLog.txt` file, which resides in the `<PPM_Home>/server/<PPM_Server_Name>/logs` directory.

The logged information has the following format:

```
Format is ["BackgroundService",<Background_Service_
Name>,"Execution Time - <Execution_Time>(ms)","Execution End
Time - <Time_Service_Run_Finished>","Entity Type - <Entity_
Type>","Entity Id - <Entity_ID>"]
```

The information resembles the following:

```
BackgroundServiceMonitorLogger:2009/01/11-23:26:07.992 PST:
"BackgroundService",Workflow Timeout Reaper,"Execution Time-
31(ms)","Execution End Time - Sun Jan 11 23:26:07 PST
2009","Entity Type - 0","Entity Id - 0" ;
```

Viewing the Services Audit Results Page

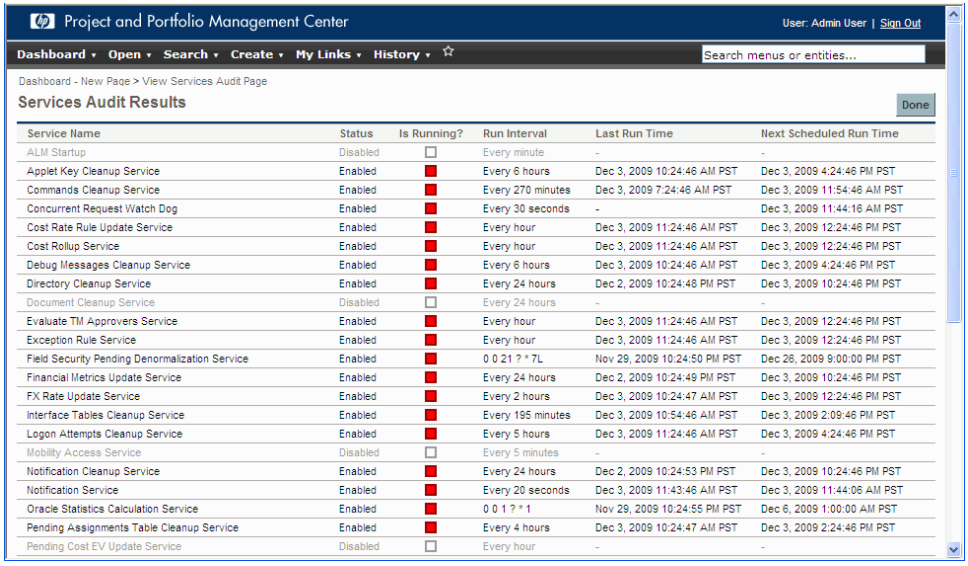
The Services Audit Results page in the standard interface enables you to quickly see the following:

- Status (enabled or disabled) of all background services
- Whether a service is currently running
- Run interval set for each service
- Time and date each service last ran
- Time and date services are to run next.

The Services Audit Results page is read-only. You cannot disable or enable, or reschedule a service from this page. For instruction on how to enable and schedule services, see *Enabling and Scheduling PPM Center Services* on page 239.

To view the Services Audit Results page:

1. Log on to PPM Center.
2. On the **Open** menu, click **Administration > View Services Audit Page.**



Service Name	Status	Is Running?	Run Interval	Last Run Time	Next Scheduled Run Time
ALM Startup	Disabled		Every minute	-	-
Applet Key Cleanup Service	Enabled		Every 6 hours	Dec 3, 2009 10:24:46 AM PST	Dec 3, 2009 4:24:46 PM PST
Commands Cleanup Service	Enabled		Every 270 minutes	Dec 3, 2009 7:24:46 AM PST	Dec 3, 2009 11:54:46 AM PST
Concurrent Request Watch Dog	Enabled		Every 30 seconds	-	Dec 3, 2009 11:44:16 AM PST
Cost Rate Rule Update Service	Enabled		Every hour	Dec 3, 2009 11:24:46 AM PST	Dec 3, 2009 12:24:46 PM PST
Cost Rollup Service	Enabled		Every hour	Dec 3, 2009 11:24:46 AM PST	Dec 3, 2009 12:24:46 PM PST
Debug Messages Cleanup Service	Enabled		Every 6 hours	Dec 3, 2009 10:24:46 AM PST	Dec 3, 2009 4:24:46 PM PST
Directory Cleanup Service	Enabled		Every 24 hours	Dec 2, 2009 10:24:48 PM PST	Dec 3, 2009 10:24:46 PM PST
Document Cleanup Service	Disabled		Every 24 hours	-	-
Evaluate TM Approvers Service	Enabled		Every hour	Dec 3, 2009 11:24:46 AM PST	Dec 3, 2009 12:24:46 PM PST
Exception Rule Service	Enabled		Every hour	Dec 3, 2009 11:24:46 AM PST	Dec 3, 2009 12:24:46 PM PST
Field Security Pending Denormalization Service	Enabled		0 0 21 ? * 7L	Nov 29, 2009 10:24:50 PM PST	Dec 26, 2009 9:00:00 PM PST
Financial Metrics Update Service	Enabled		Every 24 hours	Dec 2, 2009 10:24:49 PM PST	Dec 3, 2009 10:24:46 PM PST
FX Rate Update Service	Enabled		Every 2 hours	Dec 3, 2009 10:24:47 AM PST	Dec 3, 2009 12:24:46 PM PST
Interface Tables Cleanup Service	Enabled		Every 195 minutes	Dec 3, 2009 10:54:46 AM PST	Dec 3, 2009 2:09:46 PM PST
Logon Attempts Cleanup Service	Enabled		Every 5 hours	Dec 3, 2009 11:24:46 AM PST	Dec 3, 2009 4:24:46 PM PST
Mobility Access Service	Disabled		Every 5 minutes	-	-
Notification Cleanup Service	Enabled		Every 24 hours	Dec 2, 2009 10:24:53 PM PST	Dec 3, 2009 10:24:46 PM PST
Notification Service	Enabled		Every 20 seconds	Dec 3, 2009 11:43:46 AM PST	Dec 3, 2009 11:44:06 AM PST
Oracle Statistics Calculation Service	Enabled		0 0 1 ? * 1	Nov 29, 2009 10:24:55 PM PST	Dec 6, 2009 1:00:00 AM PST
Pending Assignments Table Cleanup Service	Enabled		Every 4 hours	Dec 3, 2009 10:24:47 AM PST	Dec 3, 2009 2:24:46 PM PST
Pending Cost EV Update Service	Disabled		Every hour	-	-

Using the Watchdog Tool

Watchdog is a stand-alone tool that issues a command to generate a thread dump whenever memory exceeds the configured threshold after a full garbage collection (GC). This tool requires that the Java garbage collection log be turned on at startup.

Watchdog monitors the memory space through the GC log that the PPM Server generates. If the memory used after garbage collection is greater than a set threshold value, the Watchdog issues a command to generate a thread dump, and the thread dump is captured in the server log. You can configure the Watchdog tool to send out email notifications about this event.

The Watchdog tool does not affect the PPM Center functionality. It is platform-dependent because it uses different mechanisms to generate thread dumps on Windows than on other, UNIX-like platforms.

Watchdog is not currently supported on AIX systems.

The memory used after a full GC is compared with the threshold. The Watchdog tool is interested in the following record in the GC log:

- With the JVM `-server` option:

```
7.138: [Full GC [PSYoungGen: 3016K->0K(229376K)] [PSOldGen:
0K->2956K(524288K)] 3016K->2956K(753664K) [PSPermGen:
9983K->9983K(20480K)], 0.1605436 secs]
```

- Without the `-server` option:

```
147.032: [Full GC 147.032: [Tenured:
30756K->34733K(227584K), 0.2966210 secs]
50507K->34733K(253184K), [Perm : 33487K->33487K(131072K)],
0.2967583 secs]
```

In the second example (without the `-server` option), the Watchdog reads the record and parses out the memory used before GC as 50507K, and memory used after GC as 34733K. The Watchdog then compares the memory used after GC, 34733K in this case, with the set threshold. If the threshold is set to 30, then the record triggers a thread dump. If the threshold is set to 35, it does not.

When the memory first exceeds threshold, PPM Center is considered to be entering a critical condition. A thread dump is triggered and a notification is sent.

After the next full GC, if the memory still exceeds the threshold (PPM Center remains in critical condition). No dump is generated as long as the memory is still higher after entering critical condition.

When the memory used falls below the threshold in subsequent GCs, PPM Center is considered to be exiting a critical condition. In this case, no thread dump is generated. You can configure the Watchdog tool to send out email notifications about this event.

If, after exiting a critical state, the memory used again exceeds the set threshold, a new critical condition starts. A thread dump is triggered and a notification is sent (if set up) every time PPM Center enters the critical condition.

Generating the GC Log

The Watchdog utility requires that a Java GC log file be present.

To enable verbose GC logging when starting PPM Center:

1. Navigate to the *<PPM_Home>*/bin directory and open the `kStart.sh` file in a text editor.
2. Locate the following `SYSTEM_PROPS` lines in the script:

   ```
   SYSTEM_PROPS="$SYSTEM_PROPS -Djava.io.tmpdir=$KNTA_HOME/
   server/$SERVER_NAME/tmp"
   SYSTEM_PROPS="$SYSTEM_PROPS -
   Djava.security.auth.login.config=$KNTA_HOME/server/$SERVER_
   NAME/deploy/admin-jmx.war/conf/auth.conf"
   ```

3. Add the following line to generate the GC metric output file:

   ```
   SYSTEM_PROPS="$SYSTEM_PROPS -Xloggc:gclog_"$SERVER_NAME"_
   `date +%Y%m%d_%H%M`.gc -XX:+PrintGCTimeStamps -
   XX:+PrintGCDetails"
   ```

4. Stop, and then start the PPM Server in debug mode by running:

   ```
   sh ./kStart.sh -debug
   ```

The *<PPM_Home>* directory now contains a file with garbage collection metrics, and the *<PPM_Home>*`/bin` directory contains the PPM Server log.

On Windows systems, you may need to start the PPM Server using `kStart.sh`. If you start the PPM Server in service mode, the Watchdog utility may not work.

Running Watchdog

To use the Watchdog tool:

1. Make sure that the *<PPM_Home>* directory contains the GC log file.
2. Navigate to *<PPM_Home>*`/utilities/watchdog/conf` directory, and open the `watchdog.properties` file.
3. Enter the values for the parameters listed in the following table.

Name	Description	Required	Default
gclog_filename	Name of the GC log file. The file name is based on the name provided during the `kStart.sh` run after the `-Xloggc:gclog` flag.	Yes	N/A
memory_threshold	Memory threshold set in MB	Yes	300
enable_email_notification	Enable email notification	No	true
enable_thread_dump	Enable thread dump	No	true
smtp_host	Host name of the SMTP server	Yes, if email notification is enabled	N/A
sender_addr	Sender email address	Yes, if email notification is enabled	N/A

Name	Description	Required	Default
recipients_addr	Specifies email recipients addresses. Use commas to separate multiple addresses.	Yes, if email notification is enabled	N/A
node_name	PPM Server name		Kintana
debug	Enables debugging		false
use_jmx	Specifies the use of JMX to retrieve the thread dump. To use this option, start the PPM Server with the following system properties: • `-Dcom.sun.management.jmxremote.port=5001` • `-Dcom.sun.management.jmxremote.ssl=false` • `-Dcom.sun.management.jmxremote.authenticate=false`	No	false
jmx_output_filename	If you do not want the JMX thread dump saved in the server log, use this parameter to specify a different file name (full directory path).	No	
jmx_url	Specifies the URL used to access the PPM Server through JMX. Use the following format: `/jndi/rmi://localhost:5001/jmxrmi`	No	
mem_used_after_gc_position_index	Indicates the number to select as the memory used after a full garbage collection in a Full GC record.	No	9
monitored_gc_record_indicator	Indicates the text string to use to identify the full GC record in the GC log.	No	Full GC

4. Find out the Java process ID of the PPM Server you want to monitor, and then run the following:

```
sh ./kWatchdog.sh pid
```

Example `watchdog.properties` file:

```
gclog_filename=c:/ppm/gclog.log
memory_threshold=600
enable_email_notification=true
enable_thread_dump=true

smtp_host=smtp-nowest.bcp.com
sender_addr=jim.wang@bcp.com
recipients_addr=jim.wang@bcp.com

node_name=Kintana
debug=false

use_jmx=false
jmx_output_filename=c:/ppm/jmx.log
jmx_url=/jndi/rmi://localhost:5001/jmxrmi
```

9 Migrating Entities

About Entity Migration

Entity migrators are HP Deployment Management object types. Each migrator is designed to migrate a specific kind of PPM Center entity and all of its dependent objects from one PPM Center instance to another.

You can use HP Deployment Management to manage configuration changes to PPM Center. HP Deployment Management comes with an out-of-the-box set of object types, or *entity migrators*, that you can use to move PPM Center configuration entities (workflows, request types, and so on) between PPM Center instances. If you maintain scratch instances for developing and testing PPM Center configurations before you deploy them into your production instance, you must use these entity migrators, and develop a workflow that drives configuration changes through your source configuration management deployment process.

Migrating configurations using entity migrators and workflows lets you automate and standardize a change-control process for your PPM Center implementation. You can build a workflow for every migrator object type, or create a single generic workflow for all migrator object types.

➤ You can only migrate entities between PPM Center instances of the same version.

You can migrate the following PPM Center entities:

- Special commands
- Object types
- Portlet definitions
- PPM Dashboard modules
- PPM Dashboard data sources
- Project types
- Work plan templates
- Report types
- Request header types
- Request types
- User data contexts
- Validations
- Workflows

Migration Order

If you plan to migrate request type, workflow, project type, and work plan template configurations that are related to each other, you must perform the migration in the following order:

1. Request type
2. Workflow
3. Request type again (if circular references exist between request type and workflow)
4. Work plan template
5. Project type

Overview of Entity Migration

Consider a scenario in which you want to migrate configuration entities between your “QA” and “Production” instances of PPM Center. You can automate and track the migration using either the source instance (QA) or the destination instance (Production). In the example that follows, you are using the destination instance to control the migration.

You migrate PPM Center entities in the same way that you perform any other deployment management process. To prepare for the entity migration you do the following:

- Set up the environment definitions for your “QA” and “Production” instances.
- Configure a workflow that directs the migration process (necessary approvals, and an automated execution step that specifies your “QA” and “Production” environments as source and destination, respectively).

After you perform these tasks, you can use HP Deployment Management packages to specify the entities to migrate. Create a package, specify your migration workflow, and add package lines using the entity migratory object types for each PPM Center configuration entity that you want to migrate.

When the automated migration execution workflow step is run, the following events occur (remember that, in this example, you are running the migration in the destination, or Production, environment):

1. The Production server connects to the QA server using Telnet or SSH, and then submits a request for the specified configuration data.
2. The QA server extracts the requested configuration data from its database and generates an XML representation of the data.
3. The QA server writes the extracted XML data into a set of temporary XML files, and packages that set of files together in a Zip file.

4. The Production server copies the Zip file that contains the bundled XML data from QA to Production.

   If you want to perform version control on changes to PPM Center configuration entities as they are migrated, you can version the compressed file that is extracted from the source instance.

   HP recommends that you not extract this file manually, except for debugging purposes.

5. The Production server unpacks the migrated compressed file into temporary storage, and reads the associated XML files.

6. The Production server imports the configuration data to its database, and then generates an execution log.

Example Migration: Extracting a Request Type

The following example shows a procedure that you can use to migrate a request type from a QA instance of PPM Center to a production instance.

To create, submit, and process migrations, you must have the required licenses and access grants. For more information, see the *Security Model Guide and Reference*.

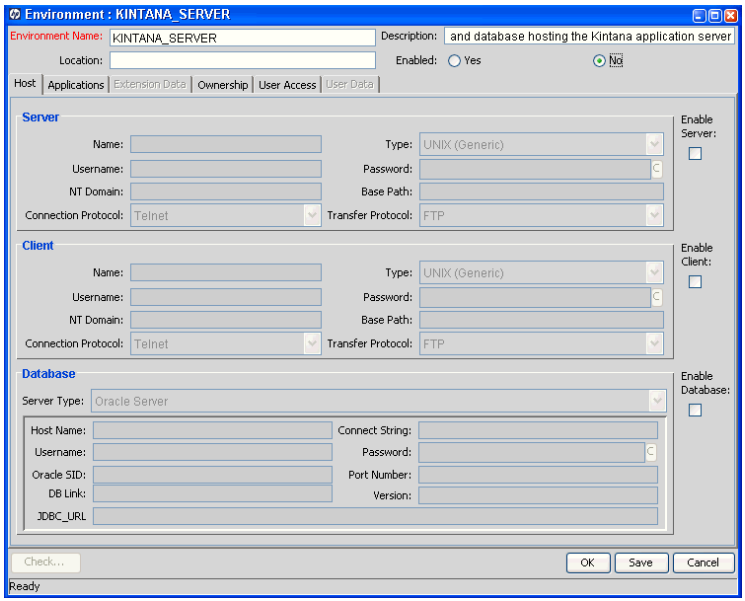
Before you perform the following steps, make sure that you have a valid user account in both the source and destination instances, and that these accounts have the same user name. When the migrator extracts an entity from the source instance, and then imports it into the destination instance, it provides your security information.

To migrate a request type:

1. If the environment definition for the PPM Server is not configured, then you must first create the `KINTANA_SERVER` environment, as follows.

   Because you control this migration from the production instance, the environment you define represents the destination for entity migrations.

   a. In the Environment Workbench, open the `KINTANA_SERVER` environment.



   b. To the right of the **Server** section, select the **Enable Server** checkbox.

   c. Provide the server information.

   Because this environment definition represents the PPM Server that you are using to run the migration, there is no need to specify connection information for it. The migrator performs the required actions locally, without opening a separate Telnet or SSH session.

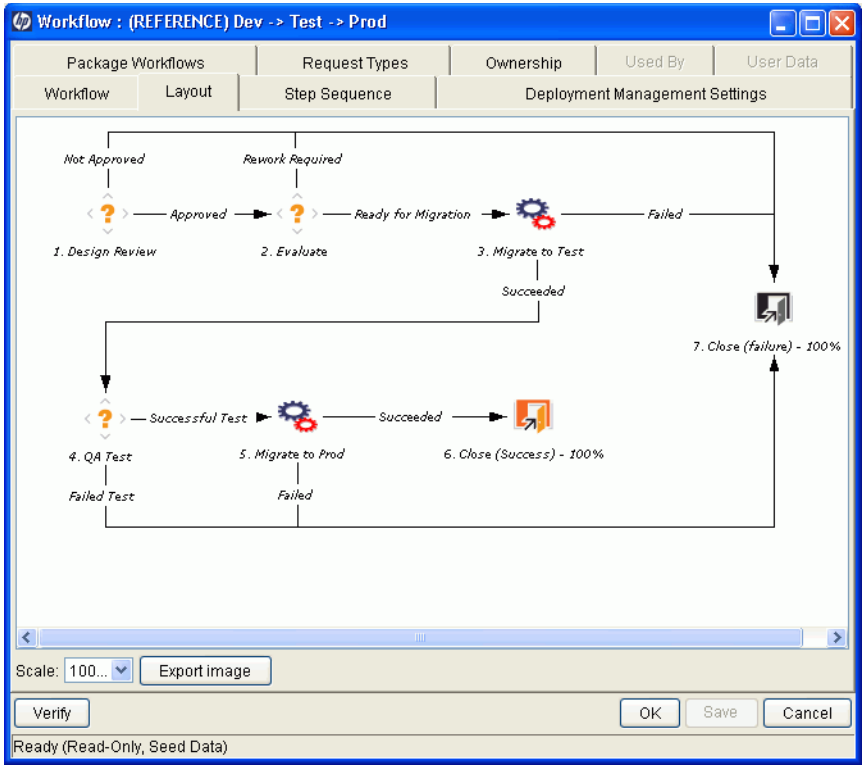
   d. Define and enable the source environment.

   You must specify connection information for the source environment, including the user name and password, base path, and connection and transfer protocols.

2. Create a deployment management workflow.

   For information about how to create a workflow, see the *HP Deployment Management Configuration Guide.*

   Specify the QA environment as the source, and the production environment (`KINTANA_SERVER`) as the destination of the execution step.



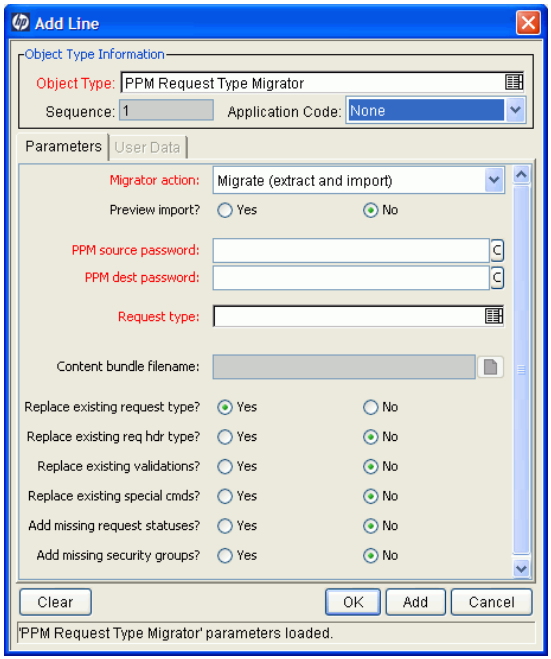
3. Create a package.

   For information about packages and how to create a package, see the *HP Deployment Management User's Guide.*

4. In the Package: <*Package_Name*> window, in the **Workflow** field, specify the workflow you created.

5. Click **New Line.**

   The Add Line dialog box opens.



6. In the **Object Type** field, type **`PPM Request Type Migrator.`**

7. On the **Parameters** tab, provide the following information

Field Name	Description
Migrator action	To control how extensive a migration to perform, in this list, select **Extract only.**
PPM source password	In this field, type the password for your PPM Center account on the source instance.
PPM dest password	In this field, type the password for your PPM Center account on the destination instance.
Request type	In this field, type the name of the request type that you want to migrate.

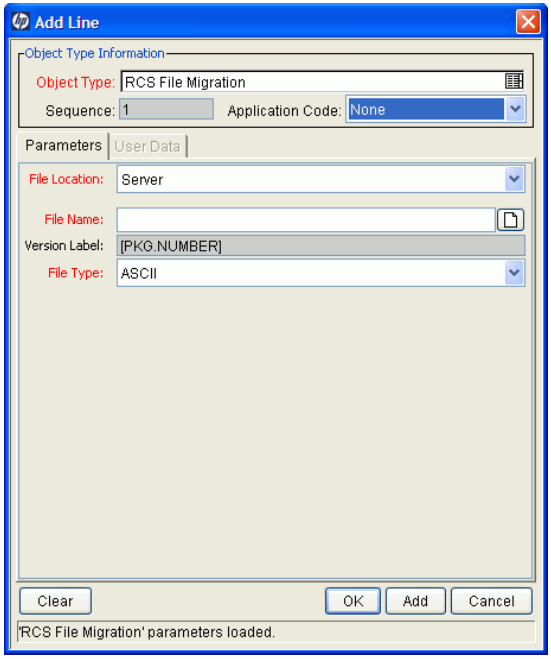
For information on **Migrator action** list dependencies, see *Migrator Action List*.

8. Submit the workflow.
9. Process the workflow.
10. Check the execution log to verify that the migration completed successfully.

Defining Entity Migrators

Each object type for the PPM Center entity migrators has a set of parameters similar to those described in this section (and as shown in the previous example). The RCS File Migrator shown in *Figure 9-1* is an example.

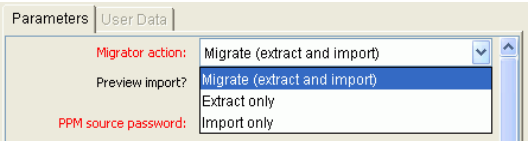
Figure 9-1. Add Line dialog box for the RCS File Migrator



Migrator Action List

To control how extensive a migration to perform, use the **Migrator action** list on the **Parameters** tab of the Add Line dialog box. *Figure 9-2* shows the **Migrator action** list.

Figure 9-2. Migrator action list



In the **Migrator action** list, you can select one of the following actions:

- **Migrate (extract and import)**
- **Extract only**
- **Import only**

Table 9-1 lists the controls in the Add Line dialog box that are affected by the migrator action you select, and provides information about how each control is affected.

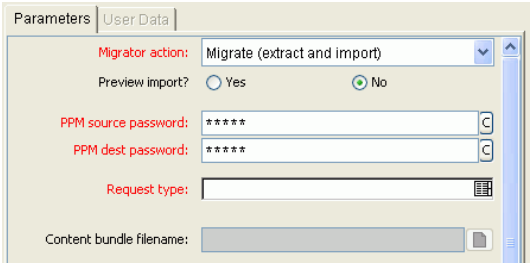
Table 9-1. Migrator action list dependencies

Control and Control Set Names	Extract and Import	Extract Only	Import Only
Preview Import	Enabled	Disabled	Enabled
Target entity field	Required	Required	Disabled
Content bundle fields	Disabled	Enabled	Required
Import behavior fields	Enabled	Disabled	Enabled
Source password	Required	Required	Disabled
Destination password	Required	Disabled	Required

Basic Parameters

Whether the basic parameters are required or simply available depends on the migrator action you select. In *Figure 9-3*, the parameters are the entity name (in this case, the request type), content bundle directory, and content bundle filename.

Figure 9-3. Basic parameters



Content Bundle Controls

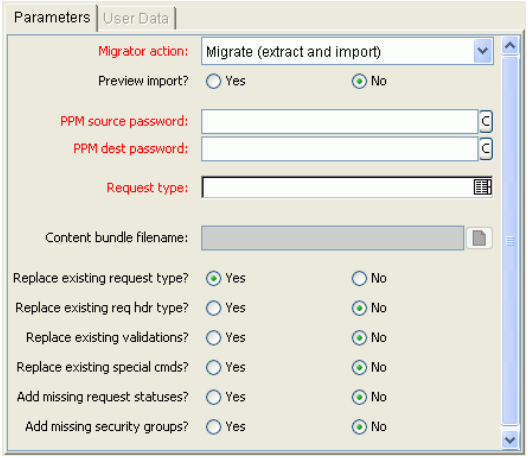
The behavior of controls related to the content bundle depends on the migrator action you select, as follows:

- If you select **Migrate (extract and import),** the migrator maintains its own internal scheme for naming and locating the temporary bundled XML data. This content bundle is extracted from the source, migrated to the destination, imported, and then cleaned up, all as part of the same execution step. The user cannot edit the content bundle information.
- If you select **Extract only,** you can specify the content bundle location and filename, or accept the default values. This lets you specify a location and naming convention that is easier to remember so that you can locate the extracted content bundle and use it as necessary (for example, check it into your version control system). By default, the migrator creates the bundle in the file system of the source PPM Server under the `<PPM_Home>/transfers` directory. The filename is based on the type of entity migrated, its package number, and its package line number.
- If you select **Import only,** you must specify the name and location of an existing content bundle file to import. You can select the file by browsing the file system of the destination PPM Server.

Import Flags

Use the import flags listed in the lower portion of the **Parameters** tab (shown in *Figure 9-4*) to control migrator behavior.

Figure 9-4. Import flags



The available import flags vary with object type.

Preview Import Option

If you set **Preview Import?** to **Yes,** the migrator does not actually import the migrated entity into the destination instance, but instead, simulates the migration and generates an execution log.

Import Behavior Controls

The following settings modify the specific import behavior for the entity to migrate.

- **Replace existing request type?** If the entity to migrate already exists in the target PPM Center instance, you can decide whether or not to replace it. The default selection is **Yes.**

  If the entity does not exist in the destination instance, it is created.

- **Replace existing req hdr type?** If the request type to be migrated references a request header type that already exists in the target PPM Center instance, you can decide whether or not to replace it. The default value is **No.**

- **Replace existing validations?** If the target entity references validations that already exist in the target PPM Center instance, you can decide whether or not to overwrite them. The default value is **No.**

  Regardless of the value, any validations that are missing from the destination instance are automatically created.

- **Replace existing special cmds?** If the validation to be migrated references PPM Center special commands (including parent and child special commands) that exist in the target PPM Center instance, you can decide whether or not to replace them. The default value is **No.**

- **Add missing request statuses?** If the request type to be migrated references request statuses that do not exist in the target PPM Center instance, you can decide whether or not to create them. The default value is **No.**

- **Add missing security groups?** If the entity to be migrated references security groups that are not included in the target instance, you can add those security groups. The default value is **No.**

  Only the list of associated access grants, but not associated users, is transferred.

Password Fields

If the **Migrator action** list displays **Migrate (extract and import),** then the **PPM source password** and **PPM dest password** fields (*Figure 9-5*) are enabled.

Figure 9-5. Password fields

Source Password Field

When the migrator contacts the source server, it uses the credentials of the current PPM Center user to authorize the entity extraction. This user must be part of a security group that contains the access grant “System Admin: Migrate HP PPM Objects.” Confirm the user password for the source server in the **PPM source password** field.

Destination Password Field

When the migrator contacts the destination server, it uses the credentials of the current PPM Center user to authorize the entity import. This user must be part of a security group that has the “Sys Admin: Migrate HP PPM Objects” access grant. Confirm the user password for the destination server in the **PPM dest password** field.

Internationalization List

Typically, in an environment in which you are managing configuration across multiple PPM Servers, all of the PPM Center databases involved have the same localization settings. However, if you must migrate configuration entities between PPM Center databases that have different localization settings, you can change the localization-checking behavior of the migrator by changing the value of the **Internationalization** list.

By default, the **Internationalization** list is invisible to users on migrator object types. But the control is enabled and set to **Same language and character set.** To change this setting:

1. Log on to PPM Center.
2. From the menu bar, select **Administration > Open Workbench.**

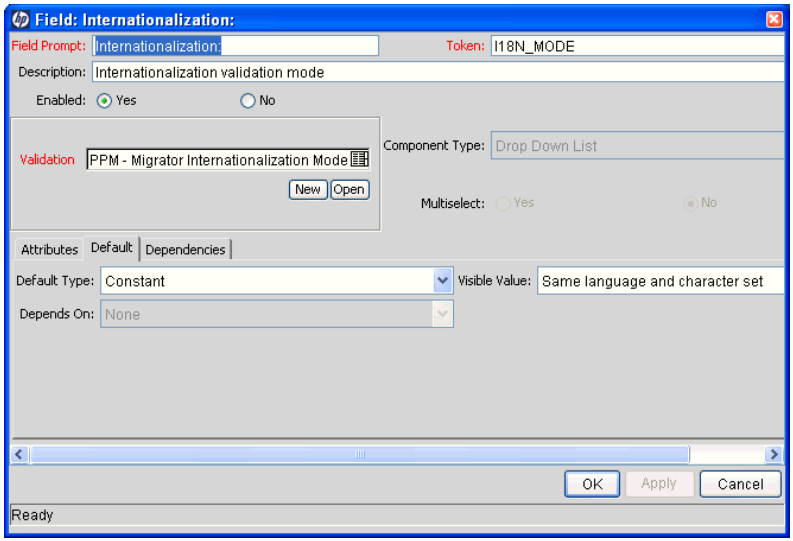
   The PPM Workbench opens.
3. From the shortcut bar, select **Deployment Mgmt > Object Types.**

   The Object Type Workbench window opens.
4. Click **List.**
5. In the **Object Name** column on the **Results** tab, double-click **PPM Request Type Migrator.**

   The Object Type: PPM Request Type Migrator window opens.
6. In the **Prompt** column on the **Fields** tab, double-click **Internationalization.**

   The Field: Internationalization window opens.

7. Click the **Default** tab.



8. From the **Visible Value** list, select one of the following:

- **Same language and character set.** This is the default option for migrating entities between PPM Center instances running under the same language and character set configuration. It is the most conservative option; any difference in locale, language, or character set between the source and destination servers is flagged as an error and the migration fails.

- **Different language or character set.** This option lets you override character set or language incompatibilities within the same localization. Use this option if you know that the language or character set settings are different across the source and destination servers, but you want to run the migration anyway and you do not anticipate the differences to cause problems with the entity data you want to migrate. For example, if the destination character set is a superset of the source character set, then you know that data extracted from the source is valid in the destination.

- **Different localization.** This option lets you migrate content between instances belonging to different localizations (for example, English to German, or German to English). This is the least restrictive option for migrating configuration data across PPM Servers that have different

locale settings. Selecting this value could potentially result in invalid data (unsupported characters, and so on) in the destination instance. Be sure to examine (and possibly update) the migrated entity data to make sure that it is valid in the destination.

Environment Considerations

When migrating entities, HP Deployment Management logs on to remote machines in the same way another user would (that is, using FTP, SCP, SSH, or Telnet). HP Deployment Management can log on to a remote server using any existing operating system user name and password.

HP recommends that you generate a new user (for example, PPM Center) on every machine to which HP Deployment Management has access. A user you create for this purpose must have full access to the `<PPM_Home>` directory on the PPM Server, and read and write permissions on other required directories.

Setting Stream Encoding for an Environment

In a Deployment Management scenario, the stream encoding specifies which character encoding scheme PPM Center's command execution engine is to use to send and receive commands to a remote computer (via telnet/SSH or FTP/SCP). This setting is important if your PPM Center instance supports multiple languages, especially in supporting remote executions in IT environments where non-English operating systems are more common.

When configuring an Environment in PPM, the stream encoding for the client (token: CLIENT_STREAM_ENCODING) specifies which encoding the client machine uses and therefore, the encoding that PPM uses in communicating with the client machine.

The stream encoding for the PPM Server (token: SERVER_STREAM_ENCODING) specifies the encoding the server machine uses and, therefore, the encoding that PPM Center uses to communicate with that server machine.

This is important if PPM Center is installed on a machine whose default encoding is set to, say, UTF-8, but must communicate with remote computers

that have, for example, Shift-JIS (a Japanese encoding) or CP-1251 (a Latin encoding) as the default encoding. Having this information known and configured in advance helps PPM Center send messages and commands that those machines can correctly interpret and to decode messages that the remote machines return.

Environment Connection Protocol

The environment definition must include information about the communication protocol (for example, Telnet) to be used to connect to the server or client. For information about connection protocols that PPM Center supports, see the *System Requirements and Compatibility Matrix* and the *HP Deployment Management Configuration Guide*.

Environment Transfer Protocol

The environment definition must include information about the transfer protocol to be used to transfer files to or from machines specified in the environment definition. Choose the transfer protocol that best suits your business and technology needs. Consider factors related to security and performance when selecting the transfer protocol. Work with the application administrator to determine which connection protocols are supported for the machines housing the deployment environments.

For information about transfer protocols, see the *HP Deployment Management Configuration Guide.*

Setting the SERVER_ENV_NAME Parameter

The PPM Center migrators depend on the `SERVER_ENV_NAME` server configuration parameter. This parameter specifies the name of an environment definition in the PPM Center system that describes the host server running that PPM Center instance.

When you installed PPM Center, the installer automatically defined the `KINTANA_SERVER` environment. This name is set as the default value of the `SERVER_ENV_NAME` server configuration parameter. PPM Center often refers to this parameter to find the environment definition that contains information about the computer(s) that hosts the PPM Server and database. For this reason, it is important that you keep this server configuration parameter synchronized with the name of the corresponding environment definition, as follows:

```
SERVER_ENV_NAME=KINTANA_SERVER
```

Security Considerations

This section provides information about security considerations related to ownership and entity restrictions.

Migration and Ownership

Different groups of PPM Center users have ownership and control over different PPM Center entities. These groups are called ownership groups. Unless a global permission has been designated to all users for an entity, members of ownership groups are the only users who have the right to edit, delete, or copy that entity. The ownership groups must also have the proper access grant for the entity in order to complete those tasks.

Application administrators can assign multiple ownership groups to entities. The ownership groups have sole control over the entity, providing greater security. Ownership groups are defined in the Security Groups window. Security groups become ownership groups when used in the ownership configuration.

Ownership applies to PPM Center entities during migrations in the following ways:

- If no ownership security is configured for the entity, any user who has permission to perform migrations can migrate it.
- If entity ownership is configured and the user migrating is not in the ownership group, the migration fails.
- If entity ownership is configured and the user migrating is in the ownership group, the migration succeeds.
- If entity ownership is configured and the user migrating is not in the ownership group but has the Ownership Override access grant, the migration succeeds.

➤ These conditions apply to entity import, but not to entity export.

Migrations and Entity Restrictions

A report type might refer to security groups through entity restrictions. The Report Type migrator transfers references to security groups, but does not create any new security groups in the destination instance of PPM Center. If the referenced security group does not exist in the destination instance, the reference is discarded in transit. A message to that effect is displayed in the migration execution log.

If the source instance contains security groups that do not exist in the destination instance during migration, the entity restrictions for the migrated report type might be inaccurate. Therefore, after migration, manually verify report types that contain entity restrictions in the destination instance.

Entity Migrators

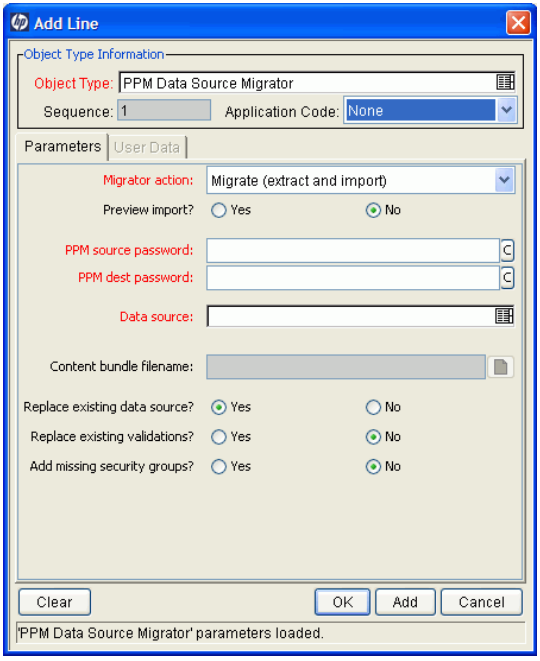
This section provides descriptions of PPM Center entity migrators.

Data Source Migrator

You can use the Data Source Migrator to move a data source that you created in the Data Source Workbench between the PPM Center instances. (Data sources provide data displayed in PPM Dashboard portlets.)

Figure 9-6 on page 345 shows the parameters for the Data Source migrator as they are displayed during package line creation.

Figure 9-6. Data Source Migrator

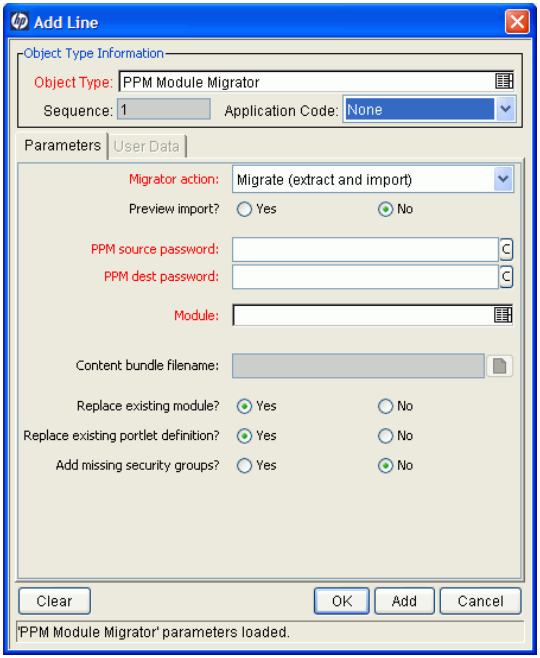


For information about the fields in this migrator, see *Defining Entity Migrators* on page 334. For information about how to create a portlet data source, see the *Creating Portlets and Modules* guide.

Module Migrator

In the PPM Center standard interface, a module is the set of pages that an administrator sets up for users to view and navigate in the PPM Dashboard. You can use the Module Migrator to move PPM Center modules from one PPM Center environment to another.

Figure 9-7. Module Migrator



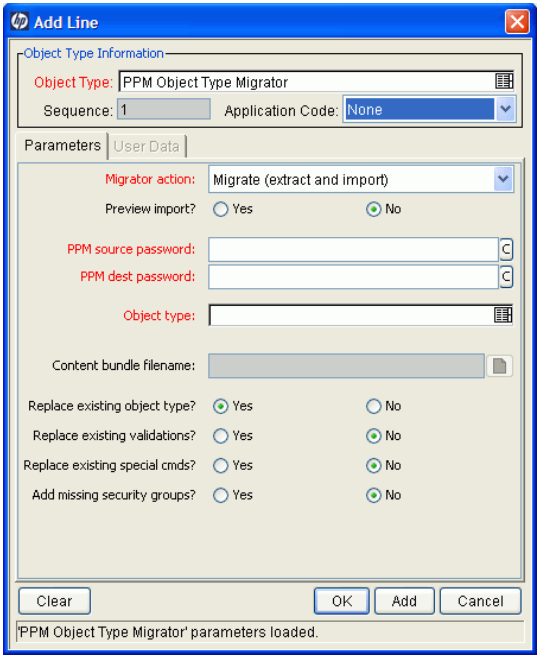
For information about the fields in this migrator, see *Defining Entity Migrators* on page 334. For information about how to create modules, see the *Creating Portlets and Modules* guide.

Object Type Migrator

The Object Type Migrator (*Figure 9-8* on page 347) contains the additional option **Replace existing special cmds?** If the validation to be migrated references PPM Center special commands (including parent and child special commands) that exist in the target PPM Center instance, you can decide whether or not to replace them. The default value is **No.**

Regardless of the migrator settings, special commands missing from the destination instance are created automatically.

Figure 9-8. Object Type Migrator



For information about most of the controls in this migrator window, see *Defining Entity Migrators* on page 334.

Configuration Considerations

The PPM Object Type Migrator also transfers the following information:

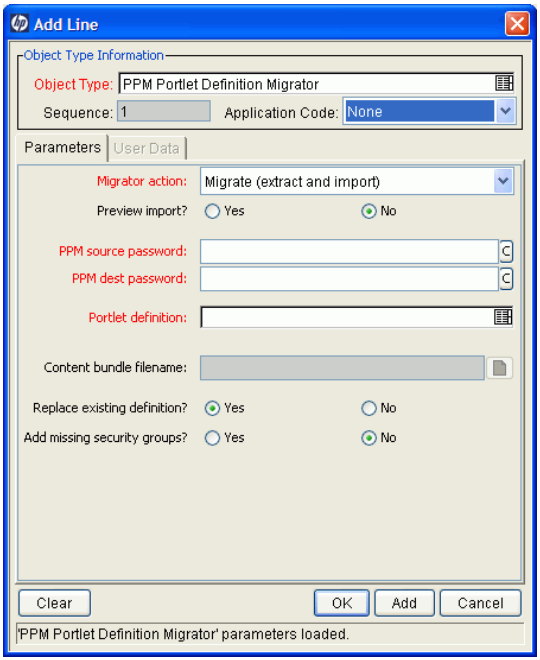
- Special commands referenced by command steps
- Validations referenced by fields
- Environments referenced by validations
- Special commands referenced by validations
- Special commands referenced by other special commands
- Ownership group information for the entity

The migrator transfers references to environments from validations, but does not create any new environments. If the referenced environment does not exist in the destination instance, the migration fails. If this happens, create the missing environment manually in the destination instance.

Portlet Definition Migrator

The Portlet Definition Migrator (*Figure 9-9*) contains all standard entity migrator object type fields. If you migrate a portlet definition to replace an existing enabled portlet definition the destination instance of PPM Center, the migrated changes are applied to all users who have added the same portlet to their PPM Dashboard pages.

Figure 9-9. Portlet Definition Migrator

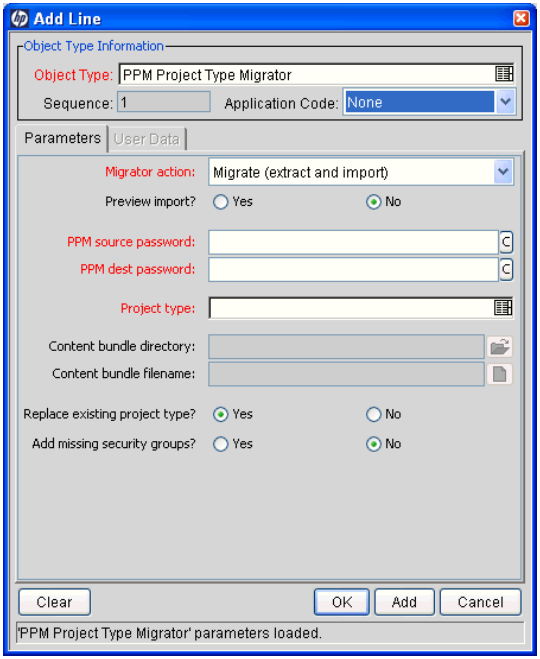


For information about the fields in this migrator, see *Defining Entity Migrators* on page 334.

Project Type Migrator

You can define project types in a development or testing instance of PPM Center, and then use the Project Type Migrator (*Figure 9-10*) to migrate them to production after testing.

Figure 9-10. Project Type Migrator



The Project Type Migrator migrates the following:

- Header information such as name and enabled flag
- All policies (including all attributes)
- References to request types for project, issue, and so on

If the migrator cannot locate these objects in the destination instance, then it drops the references and writes a warning message into the migrator log file. The migrator report contains information about the resolution (or loss) of each entity association.

Project types are connected to work plan templates, resource pools, project requests, and issue requests. None of these entities are migrated with project

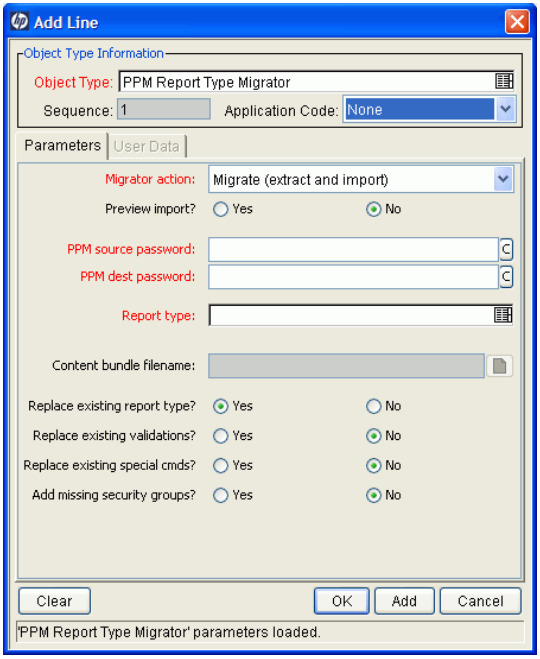
types. However, if these entities exist in the destination instance, the connection to them is maintained (the migrators identify entities by name). Because project types are useless without an associated project request, you must either migrate the associated request type first, so that the link to the project type is resolved when you migrate the project type is migrated, or edit the project type after you migrate it.

The Project Type Migrator does not transport secondary objects as dependencies.

Report Type Migrator

The Report Type Migrator (*Figure 9-11*) contains the additional option **Replace Existing special cmds?** If the validation to be migrated references PPM Center special commands (including parent and child special commands) that already exist in the target PPM Center instance, you can choose to replace them (or not). (The default value is **No.**) Regardless of their values, PPM Center automatically re-creates special commands that are missing from the destination instance.

Figure 9-11. Report Type Migrator



For information about most of the fields in this migrator, see *Defining Entity Migrators* on page 334.

Configuration Considerations

The Report Type Migrator also transfers the following information:

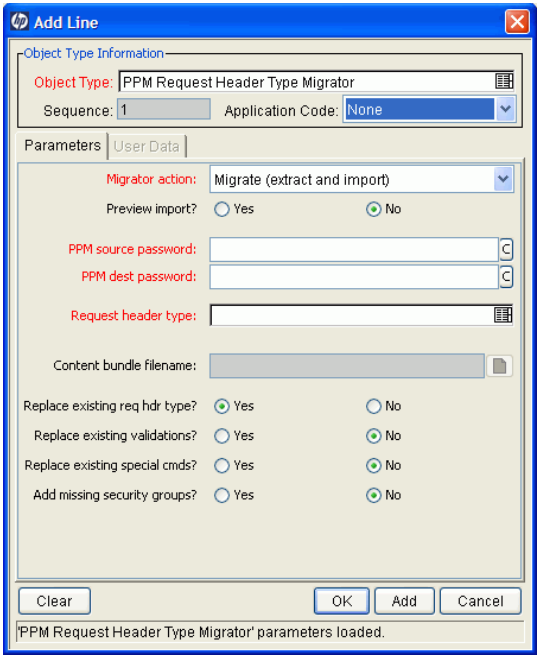
- Special commands referenced by command steps
- Validations referenced by fields
- Environments referenced by validations
- Special commands referenced by validations
- Special commands referenced by other special commands
- Ownership group information for the report type

The Report Type Migrator transfers references to environments from validations, but does not create an environment. If the referenced environment does not exist in the destination instance, the migration fails. If this occurs, you must create the missing environment manually in the destination instance.

Request Header Type Migrator

The Request Header Type Migrator (*Figure 9-12*) contains the additional option **Replace Existing special cmds?** If the validation to be migrated references PPM Center special commands that already exist in the target PPM Center instance, you can decide whether or not to replace them. This includes both parent and children special commands. (The default value is **No.**) Regardless of their values, PPM Center automatically re-creates special commands that are missing from the destination instance.

Figure 9-12. Request Header Type Migrator



For information about most of the fields in this migrator, see *Defining Entity Migrators* on page 334.

Configuration Considerations

The Request Header Type Migrator also transfers the following information:

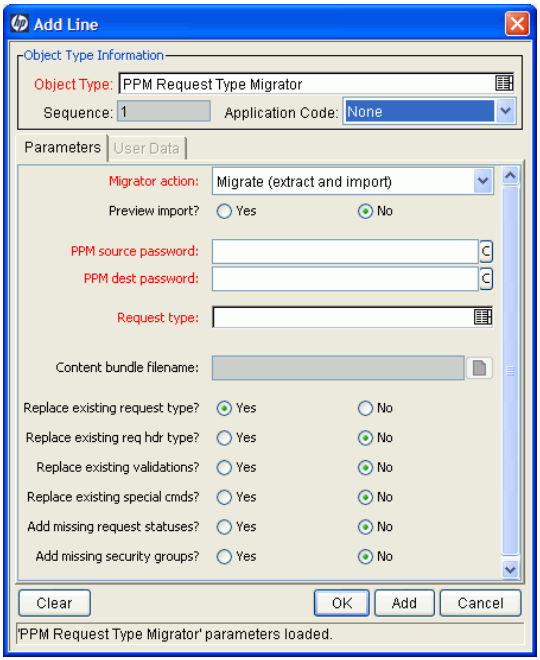
- Validations referenced by fields
- Environments referenced by validations
- Special commands referenced by validations
- Special commands referenced by other special commands
- Ownership group information for the request header type

The Request Header Type Migrator transfers references to environments from validations, but does not create an environment. If the referenced environment does not exist in the destination instance, the migration fails. In this case, you must create the missing environment manually in the destination instance.

Request Type Migrator

The Request Type Migrator (*Figure 9-13*) has additional import behavior options from which to choose.

Figure 9-13. Request Type Migrator



The additional import behavior options are as follows:

- **Replace existing req hdr type?** If the request type to be migrated references a request header type that already exists in the target PPM Center instance, you can decide whether or not to replace it. The default value is **No.**

- **Replace Existing special cmds?** If the validation to be migrated references PPM Center special commands that already exist in the target PPM Center instance, you can decide whether or not to replace them. This includes both parent and children special commands. The default value is **No.**

  Regardless of their values, PPM Center automatically re-creates special commands that are missing from the destination instance.

- **Add missing request statuses?** If the request type to be migrated references request statuses that do not exist in the target PPM Center instance, you can decide whether or not to create them. The default value is **No.**

  In the execution log, a message is displayed for each referenced request status that is not created.

If this option is set to **No,** and one of the missing request statuses is the initial status of the request type, the migration fails. In this case, you must create the request status for the initial status manually.

Configuration Considerations

The Request Type Migrator also transfers the following information:

- Request header types referenced by the request type
- Special commands referenced by command steps
- Validations referenced by fields of the request type or request header type
- Environments referenced by validations
- Special commands referenced by validations
- Special commands referenced by other special commands already referenced elsewhere
- Request statuses referenced by the request type
- Security groups referenced by the request type (on the **Access** tab)
- Workflows referenced by the request type
- Notifications referenced by the request type
- Ownership group information for the request type

The Request Type Migrator transfers references to environments from validations, but does not create an environment. If the referenced environment does not exist in the destination instance, the migration fails. In this case, you must create the missing environment manually in the destination instance.

Simple default rules, defined in the request type **Rules** tab, might reference users, workflows, or other objects. The Request Type Migrator transfers these

references, but does not create a missing user or workflow. If the referenced user or workflow does not exist in the destination instance, the reference is discarded in transit, and a message to that effect appears in the migration's execution log. You must manually reconfirm advanced default rules after migration.

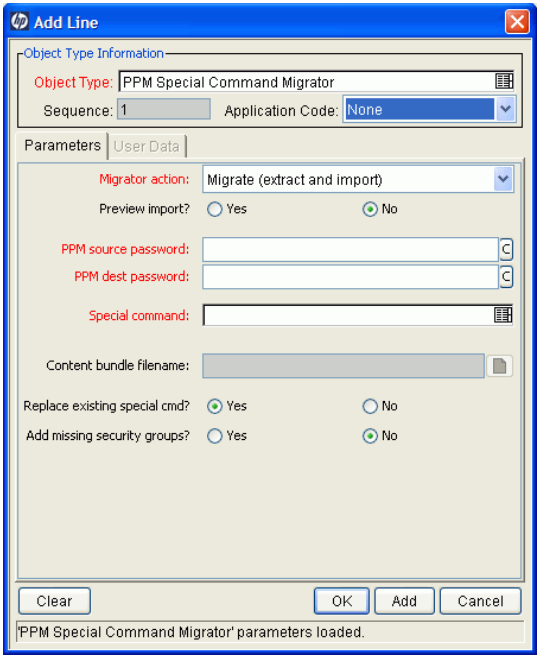
Circular references between request types and workflows could make it necessary to migrate either a request type or workflow twice:

- A new request type referring to a new workflow is migrated. Because the new workflow does not exist in the destination instance, not all references to that workflow are included in the new instance destination.
- The new workflow is migrated.
- The new request type is migrated again. This time, since the workflow it refers to exists, the references are included in the destination instance.

Special Command Migrator

If you migrate a workflow step, request type, or object type that contains special commands, the special commands are not migrated along with the entities. You must use the Special Command Migrator (*Figure 9-14*) to move the special commands between instances of PPM Center separately.

Figure 9-14. Special Command Migrator

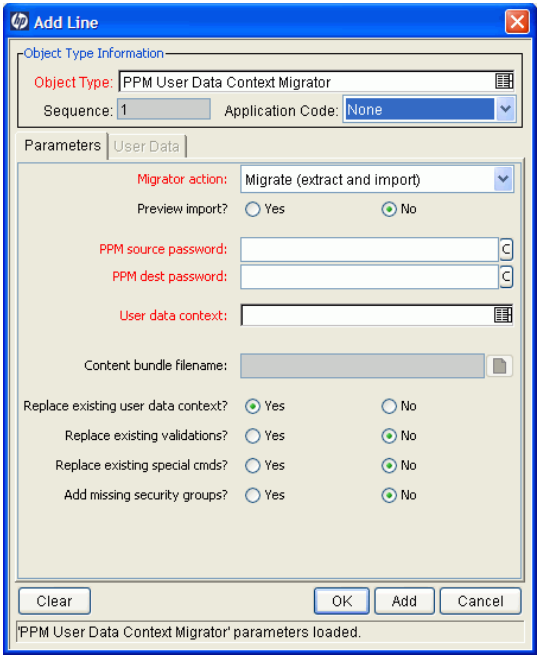


For information about the fields in this migrator, see *Defining Entity Migrators* on page 334.

User Data Context Migrator

The User Data Context Migrator (*Figure 9-15*) contains the additional option **Replace Existing special cmds?** If the validation to be migrated references PPM Center special commands that already exist in the target PPM Center instance, you can decide whether or not to replace them. This includes both parent and child special commands. (The default value is **No.**) Regardless of their values, PPM Center automatically re-creates special commands that are missing from the destination instance.

Figure 9-15. User Data Context Migrator

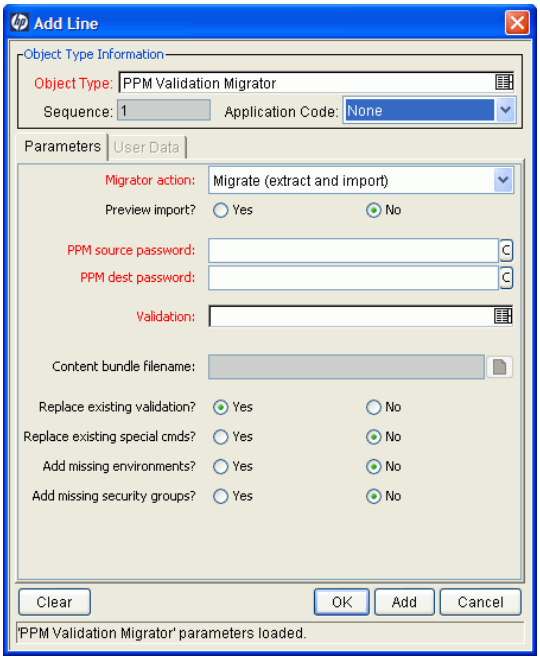


For information about most of the fields in the User Data Context Migrator, see *Defining Entity Migrators* on page 334.

Validation Migrator

The Validation Migrator is shown in *Figure 9-16*.

Figure 9-16. Validation Migrator



This migrator contains the following two additional import behavior options:

- **Replace existing special cmds?** If the validation to be migrated references PPM Center special commands that already exist in the target PPM Center instance, you can decide whether or not to replace them. This includes both special commands directly referenced by the validation, and also special commands referenced by these special commands. (The default value is **No.**) Regardless of their values, PPM Center automatically re-creates special commands that are missing from the destination instance.

- **Add missing environments?** If the validation to be migrated references environments or environment groups that do not exist in the target PPM Center instance, you can decide whether or not to create them (assuming that the option has been marked **Yes**). However, only the environment header information and user data are transferred. Application codes and

extension-specific environment tabs are not transferred. The default value is **No.**

Similarly, environment group application code information is not transferred. If an environment group already exists in the destination instance, it is not updated with environments that were added in the source instance. After migration is complete, if the migrator has created any environments, confirm and complete environment data manually.

For information about the controls in this migrator, see *Defining Entity Migrators* on page 334.

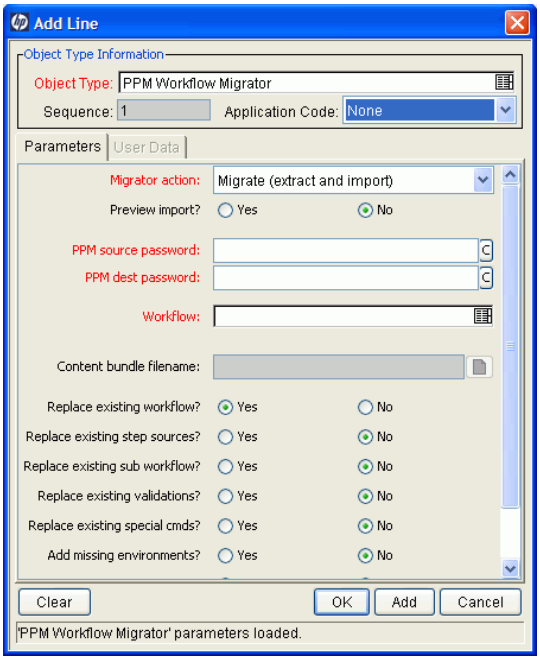
Configuration Considerations

Validation values can also carry context-sensitive user data. When migrating validation values that have such fields, you should manually set up the user data configuration in the destination instance before migration begins.

Workflow Migrator

The Workflow Migrator is shown in *Figure 9-17*.

Figure 9-17. Workflow Migrator



This migrator provides the following additional import behavior options:

- **Replace existing special cmds?** If the workflow to be migrated references PPM Center special commands that already exist in the target PPM Center instance, you can replace them. This includes special commands that the workflow references directly, as well as special commands that these special commands reference. Special commands in validations that the workflow references are also migrated.

  The default value is **No.** Regardless of the value, any special commands missing from the destination instance are created automatically.

- **Replace existing step sources?** If the workflow to be migrated references workflow decision and execution step sources that exist in the target PPM Center instance, you can choose to replace them or leave them in place. However, if workflows in the destination instance are using the existing

step sources, you cannot change certain options (such as **Workflow Scope, Validation,** and **Decision Type**), even if you set **Replace Existing Step Sources?** to **Yes.**

- **Replace existing sub workflow?** To overwrite an existing subworkflow in the target environment when subworkflows are migrated (with or without the main workflow), set this option to **Yes.**

- **Add missing environments?** If the workflow to be migrated references environments or environment groups that do not exist in the target PPM Center instance, you can create the environments or environment groups. However, only the environment header information and user data are transferred. Application codes and extension-specific **Environment** tabs are not transferred. The default value is **No.**

  Similarly, environment group application code information is not transferred. If an environment group exists in the destination instance, it is not updated with environments added to the source instance. If the migrator has created environments, then after migration, make sure that you confirm and complete the environment data manually.

- **Add missing request statuses?** If the workflow to be migrated references request status values that do not exist in the target PPM Center instance, you can create the status values. The default value is **No.**

For information about controls in this migrator, see *Defining Entity Migrators* on page 334.

Configuration Considerations

The Workflow Migrator also transfers the following information:

- Subworkflows that the workflow steps reference
- Special commands that the command steps reference
- Workflow step sources that the workflow steps reference
- Validations that the parameters or workflow step sources reference
- Environments and environment groups that the workflow steps reference

- Environments that the environment groups referenced by workflow steps reference
- Environments that validations reference
- Special commands that validations reference
- Special commands that the workflow step sources reference
- Special commands referenced by other special commands referenced elsewhere
- Security groups that the workflow steps reference
- Request statuses that the workflow steps reference
- Notifications that the workflow steps reference
- Notification intervals that notifications reference
- Security groups that notifications reference
- Ownership group information for the workflow and workflow steps

If a notification in a workflow uses a notification interval that does not exist in the destination instance, the migrator creates this notification interval. The workflow migrator does not replace existing notification intervals in the destination instance.

The Workflow Migrator transfers entity restriction references to object types, but does not create an object type. If the referenced object type does not exist in the destination instance, the migrator discards the reference and records the event in its execution log.

The Workflow Migrator transfers references to request types, but does not create request types. If the referenced request type does not exist in the destination instance, the migrator discards the reference and records the event in its execution log.

If there are circular references between workflows and request types, you may have to migrate either a workflow or request type twice:

- A new request type referring to a new workflow is migrated. Because the new workflow does not exist in the destination instance, all references to that workflow are dropped in transit.
- The new workflow is migrated.
- The new request type is migrated again. This time, because the referenced workflow exists, the references are preserved.

Replacing an Existing Workflow

There are some restrictions on using the Workflow Migrator to make changes to a process that is already in use (by requests or package lines). These restrictions help to ensure that migration does not damage these existing requests or package lines.

Specifically, workflow migration cannot succeed unless the migrator logic finds a workflow step that corresponds to each step in the existing workflow. The following conditions are used to match workflow steps between instances:

- The step source (the particular decision, execution, or condition) of a workflow step is used to match workflow steps. If the step source is not identical, then two workflow steps do not match.
- If both the incoming and existing workflows assign a unique name to each workflow step, these workflow step names are used in combination with the step source to assess the match.
- If a workflow step name is repeated within either workflow, the step sequence is used instead, in combination with the step source, to assess the match.

The Workflow Migrator cannot handle a single change in which both the names of existing workflow steps and the step sequence of existing workflow steps have changed.

To change both the names and step sequences of a workflow:

- Change step names, but do not change any step sequences. Migrate the changed workflow.
- Change step sequences, but do not change any step names. Migrate the changed workflow a second time.

Because of this matching restriction, each open request is on the same process step following the migration as it was before the migration. The migration might have changed the name of this step, but it has not transitioned request workflows.

Keep in mind that the migrator does not prevent the removal of outgoing transitions from workflow steps. Therefore, avoid “stranding” open requests at a workflow step that will be deprecated. When deprecating a process step, remove incoming transitions, but leave at least one outgoing transition from the step. This lets open requests move forward. The execution log for the migration contains a table that lists old and new workflow steps.

HP recommends that you use the **Preview import** mode first when you replace an existing workflow, and inspect this table of matched workflow steps before you run the workflow migration in non-preview mode.

Deprecating a Workflow

When the changes to a workflow are extensive, you can deprecate the existing workflow and bring the changes into the production instance as a new workflow. One advantage of implementing the changes as a new workflow is simplicity, since the new workflow is not required to contain all of the steps of the old workflow for backward compatibility.

To bring a new workflow into a production instance:

1. Rename the existing workflow and disable it in production.

   Disabling the workflow removes it from lists of workflow options when new requests are created. Requests that are in process continue to follow the old workflow until they close, unless each is manually shifted to the new process and transitioned to an appropriate point in the process. Existing defaulting rules and other configurations also continue to refer to the old workflow, regardless of the name change.

2. Migrate the new version of the workflow into the production instance, under the original name.

   Because the production instance no longer contains a workflow by this name, the migrator treats it as a new workflow.

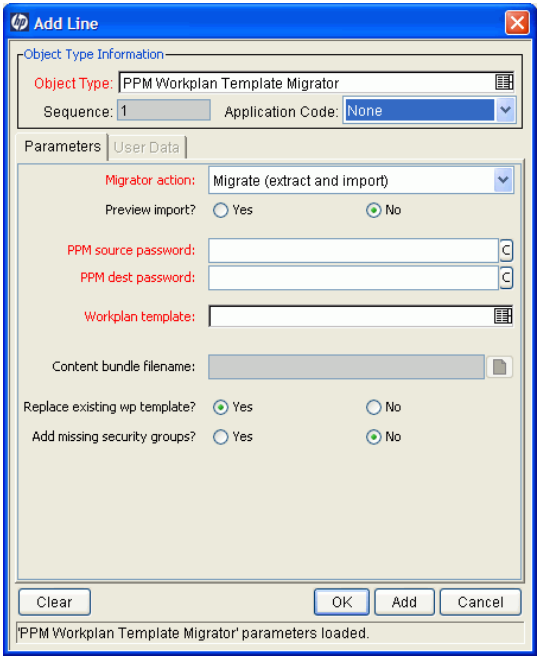
3. After the migration, you can update defaulting rules in request types to reference this new workflow.

   You can do this manually, or by migrating in versions of the request types that refer to the new workflow by its original name.

Work Plan Template Migrator

You can define work plan templates in a development or testing instance of HP Project Management, and then use the Work plan Template Migrator (*Figure 9-18*) to migrate them to production after testing is completed.

Figure 9-18. Work Plan Template Migrator



The Work Plan Template Migrator migrates the following:

- Header information such as work plan template name and list of owners (users)
- Work plan (hierarchy of tasks and task information)
- References to assigned resource groups or users (by reference only—security groups are not treated as dependent objects)

The Work Plan Template Migrator does not transport secondary objects (for example, validations) as dependencies.

10 Migrating Instances

Overview of Instance Migration

Each PPM Center instance consists of a file system and an Oracle database, which can exist on Windows or UNIX machines. You can migrate PPM Center using one of the following methods:

- Copy an entire PPM Center instance (server file system and database schemas) and move it to another location. If you are moving the copied instance to a different machine, you must have a new license key for it.
- Migrate the PPM Server to a different machine, but maintain the existing database schemas. Migrating the server requires a new license key.
- Migrate the database schemas, but maintain the existing PPM Server. Migrating only the database schema does not require a new license key.

Enterprise environments typically have multiple PPM Center instances (for example, development, test, and production). The following sections address the simplest multiple-instance configuration, which consists of a development instance (DEV) and a production instance (PROD). Each is set up on a different machine. You can extend the migration steps to support all of the instances used at your site.

Copying an Instance to Create a New Instance

To create additional PPM Center instances from an existing production (PROD) instance, clone the PROD instance.

To move from a single active instance to multiple instances:

1. Copy the PROD instance to DEV.

   This includes the file system, database, and license information.

2. Configure any changes to HP products in the DEV instance.

   This includes creating or modifying entities such as workflows, object types, request types, validations, security groups, and environments.

3. From the PROD instance, configure a package workflow to import the configuration data from the DEV instance.

4. Migrate data from the DEV instance into the PROD instance.

Running the Installation Script Twice to Create Two Instances

You can set up multiple instances as you first install and set up PPM Center. Configure one instance as the DEV instance, and the other as the PROD instance. This saves you from having to copy data from one instance into another later.

Migrating Document Management (Optional)

If your source machine has document management installed and integrated with PPM Center, see the *Document Management Guide and Reference* for information about how to migrate document management.

Preparing to Migrate

Before you can begin to migrate an entire instance to a different machine, you must obtain a new license key and stop the PPM Server, as described in the following sections.

Obtaining a New License Key

PPM Center is licensed based on the computer that hosts the PPM Server. If you plan to migrate the PPM Server to a different machine, you must obtain a new license key for the target machine. If you plan to migrate only the database schema, you do not need a new license key.

To obtain a new license key:

1. Gather the following information:
   - PPM Center version number
   - Machine IP address
   - Operating system (Windows or UNIX)
   - Server purpose (development, test, or production)
2. Go to the HP Software Support Web site (hp.com/go/hpsoftwaresupport).
3. In the right panel of the HP Customer Support page, click **Submit a License Key Request.**

   The License Request home page opens.
4. In the list of products, to the right of **PPM,** click **Get License.**
5. Provide the required information, and then click **Submit.**

Stopping the PPM Server

To make sure that you do not lose transactions, reports, or logs, stop the PPM Server before you migrate any part of a PPM Center instance. For information about how to stop the server, see *Starting and Stopping the PPM Server* on page 95.

Migrating the PPM Server

Before you migrate the PPM Server, make sure that the target machine meets the requirements described in the document *System Requirements and Compatibility Matrix*.

Migrating to a Windows Machine

To migrate the PPM Server to a Windows machine:

1. Obtain a new license key for the target server, as described in *Obtaining a New License Key* on page 373.
2. Stop the PPM Server.

   For information on how to stop the server, see *Starting and Stopping the PPM Server* on page 95.
3. Migrate the PPM Center file system:
   a. Make a compressed file of the entire `<PPM_Home>` directory.
   b. Copy the compressed file to the target machine, and then extract the file contents.
4. Migrate the PPM Center database schema.

   For information about how to migrate the database schema, see *Migrating the Database Schemas* on page 381.

5. Reconfigure the PPM Server in the target location, as follows:

   a. Run the `kConfig.sh` script, which is located in the `<PPM_Home>/bin` directory.

      The `kConfig.sh` script starts the server configuration utility, which then displays the values for each server parameter from the previous server configuration.

   b. Browse through all server configuration parameters, and make the following updates:

      - Update all parameters that refer to the DNS name or IP address of the old server to instead refer to the DNS name or IP address of the new server.

        - `BASE_URL` specifies the Web location (top directory name) of the PPM Server.

        - `RMI_URL` specifies the port on which the PPM Server listens to initiate RMI client/server communication. (This must be a unique port, distinct from the Web server, SQL*Net, and the HTTP or HTTPS ports.)

      - Update all parameters that reference a specific directory on the old server to instead reference the corresponding directory on the new server. These parameters include:

        - `ORACLE_HOME` specifies the home directory for the Oracle client tools on the PPM Server machine.

        - `BASE_PATH` specifies the full path to the directory where the PPM Server is installed.

        - `ATTACHMENT_DIRNAME` specifies the absolute pathname of the directory where attached documents are to be stored. This directory must give read/write access to Web browsers and, if the system includes an external Web server, exist outside the directory tree.

- `SERVER_TYPE_CODE` specifies the operating system on which the PPM Server is installed. Because you are placing the server on a computer running Windows, make sure you update the value to `Windows`.
- `SERVER_NAME` specifies the name of the PPM Server instance. If multiple PPM Servers are running on the same machine, this name must be unique for each server. If the server is running Windows, this name must match the name of the Windows service name.

c. To implement your changes, run the `kUpdateHtml.sh` script from the `<PPM_Home>/bin` directory.

6. Install Oracle client on the PPM Server.
7. Set the `ORACLE_HOME` environment variable to the directory path where the Oracle client software is installed.
8. Set the `JAVA_HOME` environment variable.
9. Set the `PATH` to include `JAVA_HOME\bin` and `ORACLE_HOME\bin` and make sure that the directory paths contain no spaces.
10. Make sure that the `CLASSPATH` environment variable is set and that the directory path contains no spaces.
11. To create a Windows service to use to start the new PPM Center instance, do the following:

    a. Open a command prompt, and the change to the `<PPM_Home>\bin` directory.

    b. Run `ksvc.exe`, as follows:

    ```
    ksvc install <PPM_Server_Name> -kh <PPM_Home>
    -jh <JAVA_HOME>
    ```

    The value of `<PPM_Server_Name>` is the same as the value set for the `KINTANA_SERVER_NAME` parameter in the `server.conf` file.

To create a Windows service for the nodes in a cluster, run `ksvc.exe` for each node in the cluster.

Examples:

```
ksvc install <Node1_Name> -kh <PPM_Home> -jh <JAVA_HOME>
ksvc install <Node2_Name> -kh <PPM_Home> -jh <JAVA_HOME>
ksvc install <Node3_Name> -kh <PPM_Home> -jh <JAVA_HOME>
```

12. Start the new nodes, one node at a time.

For information about how to start the server, see *Starting and Stopping the PPM Server* on page 95.

Migrating to a UNIX Machine

To migrate the PPM Server to a UNIX machine:

1. Obtain a new license key, as described in *Obtaining a New License Key* on page 373.

2. Stop the PPM Server.

   For information about how to stop the PPM Server, see *Starting and Stopping the PPM Server* on page 95.

3. Migrate the PPM Center file system as follows:

   a. On the PPM Server host machine, navigate to the parent of the `<PPM_Home>` directory.

   b. Using an archiving utility (such as Tar or Zip), create an archive file of the entire `<PPM_Home>` directory.

      Example:

      If the `<PPM_Home>` directory is named "`PPM`," run the command:

      ```
      $ tar cf ppm910.tar PPM
      ```

   c. Using FTP in binary mode, copy the archive file to the target machine. Put the archive file in the parent of the new `<PPM_Home>` directory.

   d. To extract the archive file, run the command:

      ```
      $ tar xf ppm910.tar
      ```

      This creates the new PPM Server directory structure. A directory named PPM is created automatically.

4. Migrate the PPM Center database schema.

   For information about how to migrate the database schema, see *Migrating the Database Schemas* on page 381.

5. Reconfigure the PPM Server in the target location:

   a. Run the `kConfig.sh` script, which is located in the `<PPM_Home>/bin` directory.

      The `kConfig.sh` script starts the server configuration utility, which then displays the values for each server parameter from the previous server configuration.

   b. Browse through all server configuration parameters, and make the following updates:

      - Update all parameters that refer to the DNS name or IP address of the old server to instead refer to the DNS name or IP address of the new server.

        - `BASE_URL` specifies the Web location (top directory name) of the PPM Server.

        - `RMI_URL` specifies the port on which the PPM Server listens to initiate RMI client/server communication. (This must be a unique port, distinct from the Web server, SQL*Net, and the HTTP or HTTPS ports.)

      - Update all parameters that reference a specific directory on the old server to instead reference the corresponding directory on the new server. These parameters include:

        - `ORACLE_HOME` specifies the home directory for the Oracle client tools on the PPM Server machine.

        - `BASE_PATH` specifies the full path to the directory where the PPM Server is installed.

        - `ATTACHMENT_DIRNAME` specifies the absolute pathname of the directory where attached documents are to be stored. This directory must give read/write access to Web browsers and, if the system includes an external Web server, exist outside the directory tree.

- `SERVER_TYPE_CODE` specifies the operating system on which the PPM Server is installed. Because you are placing the server on a computer running UNIX, make sure you update the value to `UNIX`.
- `SERVER_NAME` specifies the name of the PPM Server instance. If multiple PPM Servers are running on the same machine, this name must be unique for each server.

c. To implement your changes, run the `kUpdateHtml.sh` script from the `<PPM_Home>/bin` directory.

6. Place the new `license.conf` file into `<PPM_Home>/conf`.
7. Start the new PPM Server.

For information about how to start the server, see *Starting and Stopping the PPM Server* on page 95.

Migrating the Database Schemas

This section provides the procedures used to migrate the PPM Center database schemas from one database to another.

Export and Import Tools

Using incompatible versions of export and import tools causes errors in instance migration. Make sure that the export and import tools you use are either the same version, or the export tool version is earlier than the import tool version.

If You Use the Extension for Oracle E-Business Suite

If you have HP Deployment Management Extension for Oracle E-Business Suite, you must consider the location of your Primary Object Migrator Host when migrating the PPM Center database schema, because HP Object Migrator might reside in the same database, or even the same schema, as PPM Center.

Migrating the schema does not require migrating the HP Object Migrator instance because the integration method in PPM Center can be refreshed to use the existing HP Object Migrator installation. If Object Migrator shares a database with PPM Center, and you intend to migrate it as well as PPM Center, the destination database must support Object Migrator. (For more information, see the *HP Object Migrator Guide*.)

Unless PPM Center and HP Object Migrator share the same schema, the migration of Object Migrator is completely separate from the migration of PPM Center, and should be completed before you migrate the PPM Center database. Contact HP Software Support Web site (hp.com/go/hpsoftwaresupport) for instructions on how to perform this migration.

If PPM Center and HP Object Migrator share the same schema and you want to migrate both, you must coordinate the migration activities. Contact HP Software Support Web site (hp.com/go/hpsoftwaresupport) for instructions.

Regardless of the configuration, refresh the integration definition after you migrate the PPM Center schemas.

To migrate the database schemas:

1. Stop the PPM Server.

   For information about how to stop the PPM Server, see *Starting and Stopping the PPM Server* on page 95.

2. Export the PPM Center database schema to a file by running the `exp` command as shown in the following example.

   ```
   $ORACLE_HOME/bin/exp USERID=system/<Password>@<DB>
   FILE=<Export_Filename> OWNER=<PPM_Username> LOG=c:/export_
   knta_910.log
   ```

   where

<Password>	represents the password for the system user on the Oracle database
<DB>	represents the database connect string
<Export_Filename>	represents the name of the file that is to contain the export. The filename must have the `dmp` extension (for example, `kntaExport.dmp`).
<PPM_Username>	represents the name of the PPM Center database schema to export.

3. Export the RML schema.

4. Create the new PPM Center database schema:

   a. Run the `CreateKintanaUser.sql` script (located in the *<PPM_Home>*`/install_910/ppm910/system` directory) from SQL*Plus as the SYSTEM user.

      Example:

      ```
      SQL> @CreateKintanaUser.sql PPM_User PPM_Password Data_
      Tablespace Index_Tablespace TEMP_Tablespace Clob_
      Tablespace
      ```

b. Run the `GrantSysPrivs.sql` script (located in the `ppm910/sys` directory) from SQL*PLUS as the SYS DBA user.

5. Create the new PPM Center RML database schema.

   To create a new, empty RML database schema in the target database, run the `CreateRMLUser.sql` script (located in the `ppm910/sys` directory) from SQL*PLUS as the SYSTEM user.

   Example

```
SQL> @CreateRMLUser.sql Rml_User Rml_Password Rml_data_
tablespace Rml_temp_tablespace
```

6. To import data from the export file that you created earlier into the new empty PPM Center database schema, run the `imp` command, as shown in the following example.

```
$ ORACLE_HOME/bin/imp USERID=system/<Password>@<DB>
FILE=<Export_Filename> IGNORE=Y TOUSER=<New_PPM_Username>
FROMUSER=<PPM_Username> LOG=c:/import_knta_910.log
```

   where

<Password>	represents the password for the system user on the Oracle database
<DB>	represents the database connect string
<Export_Filename>	represents the name of the file that is to contain the export. The filename must have the `dmp` extension (for example, `kntaExport.dmp`).
<New_PPM_Username>	represents the name of the new PPM Center database schema
<PPM_Username>	represents the name of the PPM Center database schema previously exported.

7. Import the RML export file.

8. Create the RML-related packages in the RML schema:

   a. `cd <PPM_Home>/install_910/rml`

   b. `sqlplus <RML_Username>/<RML_Password>@<SID>@rml_driver.sql`

9. Grant privileges to the PPM Center RML database schema:

   You can find the following scripts in the `<PPM_Home>/install_910/rml directory.`

   - To set up the permissions between the two.

     ```
     sqlplus <PPM_Username>/<PPM_Password>@SID
     @RMLSetupInPPMSchema.sql <RML_Username>
     ```

   - To create synonyms to PPM Center objects in the RML schema.

     ```
     sqlplus <RML_Username>/<RML_Password>@SID
     @RMLSetupInRMLSchema.sql <PPM_Username>
     ```

10. Configure the database schema to ensure appropriate access to rebuild optimizer statistics.

    If PPM Center and HP Object Migrator share the same database schema, the PPM Center database schema is referred to as the PPM Center account, and the HP Object Migrator schema is referred to as the HP Object Migrator account.

    To provide the necessary grants and permissions to the PPM Center user, run the `GrantSysPrivs.sql` script as SYS DBA.

    ```
    SQL> @GrantSysPrivs.sql <PPM_Username>
    ```

11. If the Extension for Oracle E-Business Suite is in use and HP Object Migrator resides in the same schema as PPM Center, complete the HP Object Migrator migration.

    For assistance, contact HP Software Support Web site (hp.com/go/hpsoftwaresupport).

12. If you are using the Extension for Oracle E-Business Suite, refresh the Primary Object Migrator Host definition.

To validate any invalid PPM Center database objects generated during link regeneration, run the `RecompileInvalid.sql` script, which is located in the `<PPM_Home>/install_910` directory. Run this script from SQL*Plus connected as the new PPM Center database schema account.

13. Reconfigure the PPM Server to connect to the new database schema:

    a. Start the configuration utility by running the `kConfig.sh` script, which is located in the `<PPM_Home>/bin` directory.

    b. Update the server configuration parameters, which are described in Appendix A, *PPM Center Configuration Parameters*, on page 389.

    If you edit the `server.conf` files manually, be sure to run the `kUpdateHTML.sh` script after you complete the edit.

14. Start the PPM Server (see *Starting and Stopping the PPM Server* on page 95).

Troubleshooting Instance Migrations

This section describes common problems that you might encounter as you migrate PPM Center instances.

PPM Server Does Not Start

If you cannot start the PPM Server, check the `serverLog.txt` file (located in the `<PPM_Home>/server/<PPM_Server_Name>/logs` directory) for error messages. If the `serverLog.txt` file contains no error messages, increase the server debug level to determine whether any additional helpful information is written to the log.

To increase the server debug level:

1. Open the `logging.conf` file (located in the `<PPM_Home>/conf` directory) in a text editor.
2. Set the value of the `SERVER_DEBUG_LEVEL` parameter to `HIGH`, and then save and close the `logging.conf` file.
3. Run the `kUpdateHtml.sh` script.
4. Rerun the `kStart.sh` script, and then recheck the `serverLog.txt` file to determine whether it contains any additional information.
5. Open the `logging.conf` file.
6. Restore the default value of the `SERVER_DEBUG_LEVEL` parameter.

   Restoring the default value ensures that the file system does fill up with unnecessary information recorded in the `serverLog.txt` file(s).

7. Run the `kUpdateHtml.sh` script.

Server Starts, but You Cannot Access Applications

If the Web browser accessing the PPM Center URL generates a “Not Found” or an “Access Denied” error, check the `server.conf` file and the external Web server (if one exists) to ensure that the PPM Server installation directory is specified correctly.

If the PPM Server has recently been upgraded and the URL has changed, make sure that any saved links to the previous PPM Center URL (for example, existing requests) are updated to point to the new URL.

A PPM Center Configuration Parameters

Overview of Configuration Parameters

This appendix lists and describes the PPM Center configuration parameters, which are located in three files in the `<PPM_Home>` directory:

- `server.conf`
- `logging.conf`
- `LdapAttribute.conf`

For more information about the PPM Server directory structure and contents, see Appendix B, *Server Directory Structure and Server Tools*, on page 481.

Determining the Correct Parameter Settings

For most PPM Center installations, the default parameter values are optimal. Considerations detailed in the parameter descriptions can help you determine under what circumstances you might want to change the parameter settings.

Required Parameters

In the tables in this appendix, a single asterisk in the **Parameter** column indicates that the parameter is required to set up a PPM Server. Two asterisks in this column indicates that the parameter is required based on the condition of another parameter. For example, the `KINTANA_LDAP_ID` parameter is required only if the `AUTHENTICATION_MODE` parameter is set to `LDAP`.

In a server cluster configuration, required parameters must be set for the primary server. Secondary servers inherit the parameter values from the primary server. To override the inherited value, set the parameter to the value you want in the appropriate secondary server section of the `server.conf` file. For more information about setting up PPM Servers in a server cluster configuration, see *Configuring a Server Cluster* on page 172.

For information about how to specify your own parameters, see *Defining Custom and Special Parameters* on page 102.

Directory Path Names

Use forward slashes (/) to separate directory paths that you specify in the `server.conf` file, regardless of the operating system used. PPM Center automatically uses the appropriate path separators to communicate with Microsoft Windows. HP recommends that you not use backslashes (\) to separate directory paths in the `server.conf` file.

Server Configuration Parameters

The server configuration parameter information on a PPM Center instance comes from the following three different sources:

- KNTA_SERVER_PARAM_DEF_NLS table
- `server.conf` file
- KNTA_APPSERVER_PROPERTIES table

The KNTA_SERVER_PARAM_DEF_NLS table (definitions table) contains all of the server configuration parameters and their default values. The `server.conf` file contains a subset of the server configuration parameters in the KNTA_SERVER_PARAM_DEF_NLS table. If you specify the value for a parameter directly in the `server.conf` file, either manually or from the Administration Console, then that value supersedes the default value for the parameter in the KNTA_SERVER_PARAM_DEF_NLS table.

The KNTA_APPSERVER_PROPERTIES table contains the server configuration parameters and values that the PPM Server ultimately uses. Parameter values in the `server.conf` file are compared with those in the KNTA_SERVER_PARAM_DEF_NLS table. If a non-default value is specified for a parameter in the `server.conf` file, then the parameter is assigned that value in the KNTA_APPSERVER_PROPERTIES table. If a parameter exists only in the KNTA_SERVER_PARAM_DEF_NLS table, then the parameter is assigned the default value in the KNTA_APPSERVER_PROPERTIES table.

Using the Server Configuration Utility to Modify Server Configuration Parameters

The `server.conf` file contains the values of all of the server parameters applied during the last server configuration utility (`kConfig.sh` script) run.

HP recommends that you *not* modify the `server.conf` file directly. Instead, modify parameter values from the Administration Console interface, or use the server configuration utility (`kConfig.sh`), both of which provide a graphical interface that you can use to change the server configuration parameter values.

For information about how use Administration Console to modify parameter values, see *Modifying Parameters from the Administration Console* on page 247.

To edit the `server.conf` file using the server configuration utility:

1. Stop the PPM Server.
2. Run the `kConfig.sh` script.

   After you finish specifying configuration parameter values, the `kConfig.sh` script automatically runs the `kUpdateHtml.sh` script to regenerate the `server.conf` file and apply your changes. For information about the `kUpdateHtml.sh` script, see *kUpdateHtml.sh* on page 505.

   If you make a change to the `server.conf` file that affects more than one node in a cluster, you must:

   - Stop all the nodes in the cluster.
   - Run the `kUpdateHtml.sh` script on each machine.
   - Start all the nodes in the cluster, one at a time.

3. Restart the PPM Server.

To view a list of the server configuration parameter values on an active PPM Server, run the Server Configuration report. (See *Running Server Reports from the Admin Tools Window* on page 263 and *Running Server Reports from the Command Line* on page 267.)

Table A-1 provides descriptions of the configuration parameters in the `server.conf` file. The parameter names listed in the table are shortened versions of the actual names, all of which start with the string `com.kintana.core.server`. For example, the full name of the `CLIENT_TIMEOUT` parameter is `com.kintana.core.server.CLIENT_TIMEOUT`.

Table A-1. Server configuration parameters (page 1 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
AAL_DATA_EXTRACT_MAX_RESOURCES	Determines the maximum number of resources that can be extracted from the database and returned in the Analyze Assignment Load portlet in HP Resource Management. This acts as a safety valve to prevent PPM Center from hanging if a user defines portlet criteria that would return a very large volume of data.	Default: `1000` Valid values: Positive integer
AAL_PORTLET_MAX_RESOURCES	Maximum number of resources pools to be represented in the Analyze Assignment Load portlet in HP Resource Management. If the resource count exceeds the set value, the PPM Server stops calculating and displays no result in the portlet. **Note:** Setting a very high value for this parameter could affect system performance.	Default: `300` Valid values: Any positive integer
ALLOW_SAVE_REQUEST_DRAFT	If set to `true`, `enables` user to save requests without automatically submitting them in the standard interface.	Default: `false` Valid values: `true`, `false`
**APP_SERVER_HAJNDI_BINDING_PORT	Port to which the HA-JNDI server binds as it waits for JNP clients. Define this port if the PPM Server is configured as a node in a server cluster.	Default: `1100` Valid values: Port number unique per node in the cluster.

Table A-1. Server configuration parameters (page 2 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
**APP_SERVER_ HAJNDI_RMI_PORT	JBoss HA-JNDI RMI port. Set a value for this parameter if the PPM Server is part of a server cluster.	Default: `1101` Valid values: Port number must be unique for each node in the cluster.
**APP_SERVER_ JMX_RMI_PORT	JBoss JMX RMI port. Set a value for this parameter if the PPM Server is part of a server cluster.	Default: `19001` Valid values: Port number must be unique for each node in the cluster.
**APP_SERVER_ JRMP_INVOKER_ RMI_PORT	JBoss JRMP Invoker RMI port, which must be defined if the PPM Server is part of a server cluster.	Default: `4444` Valid values: Port value that is unique for the instance, or for the node in the server cluster.
**APP_SERVER_ MULTICAST_PORT	Port used by JBoss's HAPartition service to coordinate cluster nodes. All nodes in the cluster must use the same value. Specify a value that is different than the value set for *MULTICAST_PORT* or the value for the cache multicast port in cache.conf (which is hardcoded to 46545).	Default: `9101` Valid values: Port value that is unique for the node in the server cluster and that is different than the value set for the *MULTICAST_ PORT*.

Table A-1. Server configuration parameters (page 3 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
**APP_SERVER_ NAMING_SERVICE_ BINDING_PORT	JBoss Naming Service binding port. You must set a value for this parameter if the PPM Server is part of a server cluster.	Default: 1199 Valid values: Port value that is unique for the instance, or for the node in the server cluster.
**APP_SERVER_ NAMING_SERVICE_ RMI_PORT	JBoss Naming Service RMI port. You must set a value for this parameter if the PPM Server is part of a server cluster.	Default: 1198 Valid values: Port value that is unique for the instance, or for the node in the server cluster.
**APP_SERVER_ POOLED_INVOKER_ BINDING_PORT	JBoss Pooled Invoker Binding port. You must set a value for this parameter if the PPM Server is part of a server cluster.	Default: 4445 Valid values: Port value that is unique for the node in the server cluster.
**APP_SERVER_ POOLEDHA_ BINDING_PORT	JBoss PooledHA Binding port. You must set a value for this parameter if the PPM Server is part of a server cluster.	Default: 4446 Valid values: Port value that is unique for the node in the server cluster.
**APP_SERVER_ UIL2_BINDING_ PORT	JBoss UIL2 Binding port. You must set a value for this parameter if the PPM Server is part of a server cluster.	Default: 8093 Valid values: If the PPM Server is part of a cluster, specify a value that is unique for the node.

Table A-1. Server configuration parameters (page 4 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
**APP_SERVER_ WEBSERVICE_ PORT	JBoss WebService port. You must set a value for this parameter if the PPM Server is part of a server cluster.	Default: `8083` Valid values: Port value that is unique for the node in a server cluster.
ARP_MAX_ RESOURCE_POOLS	Add this parameter to the `server.conf` file to specify the maximum number of resource pools that can be represented in the Analyze Resource Pool portlet in HP Resource Management. If the resource pool count exceeds the set value, the PPM Server stops calculating and no result is displayed in the portlet. **Note:** Setting a very high value for this parameter could affect system performance.	Default: `30` Valid values: Positive integer
ARP_MAX_ RESOURCES	Add this parameter to the `server.conf` file to specify the maximum number of resources that can be represented in the Analyze Resource Pool portlet in HP Resource Management. If the resource count exceeds the set value, the PPM Server stops calculating and no result is displayed in the portlet. **Note:** Setting a very high value for this parameter could affect system performance	Default: `300` Valid values: Positive integer

Table A-1. Server configuration parameters (page 5 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
ARP_PORTLET_MAX_RESOURCE_POOLS	Maximum number of resources to be represented in the Analyze Resource Pool portlet in HP Resource Management. If the resource pool count exceeds the set value, the PPM Server stops calculating and displays no result in the portlet. **Note:** Setting a very high value for this parameter could degrade system performance.	Default: `30` Valid values: Any positive integer
*ATTACHMENT_DIRNAME	Absolute pathname of the directory where attached documents are to be stored. This directory must: • Give read/write access to Web browsers • Be outside the directory tree if the system includes an external Web server In a server cluster, all servers must be able to access and share the specified directory.	Example `C:/ppm/eon/attachments`
AUTHENTICATE_REPORTS	If set to `true`, access to all reports requires user authentication. (A user must provide a PPM Center user login ID).	Default: `true` Valid values: `true, false`
*AUTHENTICATION_MODE	Method(s) used to authenticate users. To specify multiple modes, use a comma-delimited list of valid values.	Default: `ITG` Valid values: `ITG, LDAP, NTLM, SITEMINDER`

Table A-1. Server configuration parameters (page 6 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
AUTO_COMPLETE_LONG_TYPE_CULLTEXT_REQUIRED	Determines whether the user must enter a filter in the auto-complete dialog box in order to retrieve the initial results for validations of type long. If set to `true`, and no user filter is specified, the validation returns an empty result set. The user must then either click **Find** or select the **Show All** link.	Default: `false` Valid values: `true, false`
AUTO_COMPLETE_LONG_TYPE_MAX_ROWS	Maximum number of rows in long auto-complete lists.	Default: `5000`
AUTO_COMPLETE_QUERY_TIMEOUT	Sets query timeouts on auto-complete lists to prevent excessive database CPU use.	Default: `30` (seconds)
AUTO_COMPLETE_SHORT_TYPE_MAX_ROWS	Maximum number of rows to retrieve from the database for short type auto-completion lists.	Default: `500`
AUTOCOMPLETE_STATUS_REFRESH_RATE	Interval at which the command status is refreshed to provide a list of values in an auto-complete list.	Default: `5` (seconds)
BACKGROUND_SERVICE_MONITOR_THRESHOLD	If *ENABLE_BACKGROUND_SERVICE_MONITOR* is enabled, this parameter determines the threshold value of the Background Services monitor.	Default: `900000` (milliseconds)
BASE_CURRENCY_ID	ID for the currency in which your organization maintains its accounting system.	Default: `97`

Table A-1. Server configuration parameters (page 7 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
BASE_LOG_DIR	Points to the “logs” directory directly under the directory specified for **BASE_PATH*. In a server cluster, all servers must be able to access and share the specified directory.	Example `com.kintana.core.server.BASE_LOG_DIR=C:/PPM/eon/logs`
*BASE_PATH	Full path to the directory where the PPM Server is installed.	Default: Based on the operating system platform. Example `C:/PPM/eon/`
*BASE_URL	Web location (top directory name) of the PPM Server.	Example `http://www.mydomain.com:8080`
BASE_URL_QC_INTEG	Used in the process of enabling the integration of PPM Center with Quality Center when PPM Center is set up with an external Web server with HTTPS enabled. For detailed information on how to use this parameter, see the *HP Solution Integrations Guide*.	Default: N/A Valid values: `http://<Instance_Host_Name>:<HTTP_Port>/itg/`

Table A-1. Server configuration parameters (page 8 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
BUDGET_IN_THOUSAND_SHOW_DECIMAL	Used with the `BUDGET_IN_WHOLE_DOLLARS` parameter as follows: • If `BUDGET_IN_WHOLE_DOLLARS` is set to `true`, the `BUDGET_IN_THOUSAND_SHOW_DECIMAL` parameter is ignored and values are displayed as whole numbers. • If `BUDGET_IN_WHOLE_DOLLARS` is set to `false`, and `BUDGET_IN_THOUSAND_SHOW_DECIMAL` is set to `false`, values are displayed as 1000s without decimals. For example, the value 1234567 is displayed as 1235. • If `BUDGET_IN_THOUSAND_SHOW_DECIMAL` is set to `true`, values are displayed as 1000s with decimals. For example, the value 1234567 is displayed as 1234.567.	Default: `false` Valid values: `true, false`
BUDGET_IN_WHOLE_DOLLARS	Determines whether forecast, approved funding, and financial summary values are expressed in whole dollars.	Default: `false` Valid values: `true, false`
CCM_MACHINE_URL [a]	URL of the Change Control Management server and port number used for integration with PPM Center.	Valid value format: `http://<Host>:<Port>/ccm/`
CHANGE_MANAGEMENT_LICENSE_KEY	License key for HP Demand Management.	Default: N/A

Table A-1. Server configuration parameters (page 9 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
CLIENT_TIMEOUT	Frequency (in minutes) with which the PPM Workbench interface session sends a message to the PPM Server that indicates the client is still active. Under normal operation, do not change this value.	Default: 5
CLOSE_BROWSER_ON_APPLET_EXIT	Determines whether the client browser closes after the user quits the PPM Workbench.	Default: `false` Valid values: `true, false`
**CONC_LOG_TRANSFER_PROTOCOL Required if ***ORACLE_APPS_ENABLED* = `true`	Transfer protocol used to transfer concurrent request logs and patching README files. If you use Object Migrator with PPM Center, you must specify value.	Default: `FTP` Valid values: FTP, SCP
**CONC_REQUEST_PASSWORD Required if ***ORACLE_APPS_ENABLED* = true	Encrypted password of the concurrent request user. If you use Object Migrator with PPM Center, you must specify value.	Default: (none) Example `fnd`
**CONC_REQUEST_USER Required if ***ORACLE_APPS_ENABLED* = true	Valid user on the Oracle system that can be used to retrieve concurrent request output files. If you use Object Migrator with PPM Center, you must specify value. Set the retrieval method (FTP or SCP). *See **CONC_LOG_TRANSFER_PROTOCOL* on page 401.	Example `applmgr`

Table A-1. Server configuration parameters (page 10 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
COST_ CAPITALIZATION_ ENABLED	Determines whether cost capitalization is allowed for costs incurred on projects. **Note:** HP strongly recommends that you not change the value of this parameter after PPM Center installation.	Default: `false` Valid values: `true, false`
DATE_ NOTIFICATION_ INTERVAL	Interval at which the PPM Server is to check to determine whether date-based notifications are pending, and to send them.	Default: `60` (minutes)
DAYS_TO_KEEP_ APPLET_KEYS	Number of days applet keys are retained in the `KNTA_APPLET_KEYS` table.	Default: `1`
DAYS_TO_KEEP_ COMMANDS_ROWS	Number of days records are kept in the prepared commands tables before they are cleaned up.	Default: `1`
DAYS_TO_KEEP_ INTERFACE_ROWS	Number of days to keep records of all interfaces.	Default: `5`
DAYS_TO_KEEP_ LOGON_ATTEMPT_ ROWS	Number of days to keep records of all logon attempts. **Note:** PPM Center keeps a record of the most recent logon attempt, regardless of when it occurred. So, for example, if the sole user only logs on once a month, PPM Center retains the record of the last logon, even if `DAYS_TO_KEEP_LOGON_ATTEMPT_ROWS` is set to 14 days.	Default: `14`
**DB_ CONNECTION_ STRING (Required if RAC is used)	Oracle RAC (Real Application Clusters) service name.	Example `K92RAC`

Table A-1. Server configuration parameters (page 11 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
DB_LOGIN_ TIMEOUT	Amount of time that the PPM Server is to continue to try to log on to the database (acquire the JDBC connections that make up the connection pool) before reporting that the database is unavailable.	Default: `30000` (milliseconds)
*DB_PASSWORD	Password for the database schema that contains the PPM Center tables.	Example `#!#<Password>#!#`
*DB_USERNAME	Name of the database schema that contains the PPM Center tables.	Example `knta`
DEFAULT_ COMMAND_ TIMEOUT	Number of seconds the PPM Server can try to run commands before it times out.	Default: `90`
*DEFAULT_PAGE_ SIZE	Number of work plan lines that can be loaded into the Work Plan page for all new users. This setting indicates whether to use the fast setting or the slow setting (rather than indicating a specific size). In new installations, this defaults to the slow connection setting. HP recommends that the system administrator review this setting after installation. If your system has mostly LAN users (fast connections), set this to use the fast setting. If your system has mostly WAN/VPN users (slow connections) or mixed usage, set this to use the slower setting.	Default: `50`

Table A-1. Server configuration parameters (page 12 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
*DEFAULT_PAGE_SIZE_OPTION	Initial type of page size (low, high, or custom) selected for the Edit My Profile page.	Default: `LOW_PAGE_SIZE` Valid values: `LOW_PAGE_SIZE`, `HIGH_PAGE_SIZE`, and `CUSTOM_PAGE_SIZE`
DEFAULT_REQUEST_SEARCH_ORDER_BY_ID	Affects the **Sort By** field on the Search Requests page. The default value is `true`, which sorts the search results based on Request ID. If set to `false`, search results are returned unsorted.	Default: `true` Valid values: `true, false`
*DEFAULT_TIME_SHEET_LINES_VIEW_MODE	Determines whether the time sheet items in HP Time Management are grouped under appropriate headings, or displayed in a flat list without headings. For detailed information about grouped and ungrouped display of time sheet items, see the *HP Time Management User's Guide*.	Default: `grouped` Valid values: `grouped, flat`
DEMAND_FIELDS_CACHE_SIZE	Specifies the size of the demand set fields cache in number of demand set.	Default: `10`
DEMAND_FIELDS_CACHE_TIMEOUT	Timeout for the demand set fields cache, expressed in seconds.	Default: `360000` (seconds)
DEPLOY_BASE_PATH	Deployment destination. **Note:** HP recommends that you leave the default value unless the PPM Server directory is renamed.	Default: `server/`

Table A-1. Server configuration parameters (page 13 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
DIST_ENGINE_ MONITOR_SLEEP_ TIME	Used in release distribution. Number of milliseconds the monitor waits between checking existing result listener. Use this parameter to adjust the amount of time the monitor sleeps between checks. **Note:** HP recommends that you not change this value. It does not affect performance.	Default: `5000` (milliseconds)
EMAIL_ NOTIFICATION_ SENDER	Email address of the default sender of email notifications. This sender receives any error messages associated with email notifications.	Example `mgr@ppm.com`
ENABLE_ BACKGROUND_ SERVICE_MONITOR	If set to `true`, enables the background services monitor.	Default: `true` Valid values: `true, false`
ENABLE_ CONCURRENT_ REQUEST_ UPDATES	Related to requests in HP Demand Management. If set to `true`, multiple users can change the same request simultaneously. Request data such as notes, new references and new table entries are always saved. Conflicting changes that cannot be saved are displayed to the user as differences.	Default: `true` Valid values: `true, false`
ENABLE_DB_ SESSION_ TRACKING	If set to `true`, enables a stack trace to be reported in the PPM Center DB Server Reports, which you can use to track the exact line of code used to request a database connection.	Default: `false` Valid values: `true, false`

Table A-1. Server configuration parameters (page 14 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
ENABLE_JDBC_LOGGING	Enables JDBC logging, which records SQL runs against the database, the time required to run the SQL, and the time to retrieve the results. This information is recorded in `jdbc.System_Name.log` in the server log directory. This parameter is useful in debugging system performance problems. You can set this parameter in the PPM Workbench interface without stopping the system (**Edit > Settings**).	Default: `false` Valid values: `true, false`
ENABLE_LOGIN_COOKIE	If set to `true,` the **Remember my logon** option is displayed on the logon page, and a cookie is placed on the client browser to maintain a record of the user logon information. **Remember my logon** sets a cookie on the local machine that lets a user log on to PPM Center later, without providing logon information. You can also view reports through notification links, and so on, without logging on. This cookie is removed only if the user clicks **Sign Out** (or clears cookies, or the cookie expires). If a user closes the browser window without signing off, the cookie is not cleared. To disable this function, change the parameter value to `false`. **NOTE:** If PPM Center is integrated with an SSO provider such as SiteMinder, then set this parameter to false. If PPM Center is *not* integrated with an SSO provider, HP recommends that you keep the parameter set to true.	Default: `true` Valid values: `true, false`

Table A-1. Server configuration parameters (page 15 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
ENABLE_MOBILITY_ACCESS_SERVICE	If set to `true`, enables the Mobility Access Service.	Default: `false` Valid values: `true, false`
ENABLE_OVERVIEW_PAGE_BUILDER	Provided for backward compatibility if you have customized “overview pages.” If you do not have customized “overview pages,” leave the default value (`false`).	Default: `false` Valid values: `true, false`
ENABLE_PORTLET_FULL_RESULTS_SORTING	If set to `true`, enables sorting of portlet results.	Default: `false` Valid values: `true, false`
ENABLE_PORTLET_MONITOR	If set to `true`, enables the portlet monitor, which gathers statistics on portlet performance. If activity exceeds the threshold value (determined by the parameter), the captured information is output to the `thresholdLog.txt` log file, which resides in same directory as the server log.	Default: `true` Valid values: `true, false`
ENABLE_PROJECT_LAUNCH_FROM_ACTION_MENU	If set to `true`, enables users with the required permission to open the PPM Workbench as a stand-alone application using Active X.	Default: `true` Valid values: `true, false`

Table A-1. Server configuration parameters (page 16 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
ENABLE_PROMISE_ RESOURCE_ ALLOCATION	If set to `true`, a user who has the Resource Management: Promise Unspecified Resources access grant can promise allocations and modify or remove previously promised allocations. A user who does not have the access grant cannot promise allocations or modify or remove previously promised allocations. In this case, the promise allocations for the resource pool (if any) are displayed (view-only) on the Staffing Profile and the Resource Allocation Management pages. The values are used in computing totals, regardless of whether the user has the Resource Management: Promise Unspecified Resources access grant.	Default: `false` Valid values: `true, false`
ENABLE_QUALITY_ CENTER_ INTEGRATION [a]	If no XML mapping file has been generated and deployed to both PPM Center and Quality Center, set this value to `false`. If a mapping has been deployed, to enable the integration, set the value to `true`.	Default: `true` Valid values: `true, false`
ENABLE_QUERY_ BUILDER	If set to `true`, enables the advanced “query builder” capability for searching HP Demand Management requests.	Default: `true` Valid values: `true, false`
ENABLE_ QUICKLIST_ UPDATE	Controls the visibility of the **Update** button on the Quick List.	Default: `true` Valid values: `true, false`

Table A-1. Server configuration parameters (page 17 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
ENABLE_SQL_TRACE	Determines whether performance statistics for all SQL statements run are placed into a trace file. The SQL trace facility generates the following statistics for each SQL statement: • Parse, run, and fetch counts • CPU and elapsed times • Physical reads and logical reads • Number of rows processed • Misses on the library cache • User name under which each parse occurred • Each commit and rollback This parameter corresponds to the **Enable DB Trace Mode** checkbox in the Server Settings dialog box.	Default: `false` Valid values: `true, false`
ENABLE_SSL_LOGIN	Enables flag for the SSL Login page. If set to `true` (default), then the following parameters must also be set: • *HTTPS_PORT* • *HTTPS_WEB_THREAD_MIN* • *HTTPS_WEB_THREAD_MAX* • *HTTPS_KEYSTORE_LOCATION* • *HTTPS_KEYPASSWORD*	Default: `false` Valid values: `true, false`

Table A-1. Server configuration parameters (page 18 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
ENABLE_ TIMESTAMP_ LOGGING	If set to `true`, specifies that a timestamp is written into the log for each line of debugging text that corresponds to actions you have performed. The timestamp can help you locate information in the server log files about events that occurred at a specific time, or to determine how much time elapsed between specific logged statements. **Note:** Including the timestamp adds text to each logged statement, which bloats the log file and can make it more difficult to read.	Default: `true` Valid values: `true, false`

Table A-1. Server configuration parameters (page 19 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values

ENABLE_TM_ALLOW_EMPTY_ITEM	Default setting used by time sheet policies to specify what occurs when users try to submit time sheets that contain empty lines (lines in which all values are 0). Time sheet policies have options that correspond to these settings. For the users assigned to a time sheet policy, the policy can override the default value set for this parameter. Following are the parameter settings and descriptions of their effects if the user's time sheet policy does not override the selected setting:
	• `ALLOW.` The user can submit time sheets that contain empty lines.
	• `WARNING.` PPM Center displays a warning that advises the user to consider entering data in the empty lines before submitting the time sheet. The user can, however, still submit the time sheet.
	• `RESTRICT.` PPM Center requires the user to enter data in the empty lines or delete those lines before submitting the time sheet.
	To improve PPM Center performance and to make it easier for approvers to review submitted time sheets, HP strongly recommends using the `RESTRICT` setting.
	For more information about time sheet policies, see the *HP Time Management Configuration Guide.*
		Default:
		• `RESTRICT` when used by new time sheets or new time sheet policies on an upgraded instance or a new installation.
		• `WARNING` for existing time sheets on an upgraded instance.
		Valid values:
		`ALLOW,`
		`WARNING,`
		`RESTRICT`

Table A-1. Server configuration parameters (page 20 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
ENABLE_TM_ WORK_ITEM_ PACKAGES	If set to `true`, the **Allowed work item types** list on the **Work Items** tab of a time sheet policy includes the **Packages** checkbox. If set to `false`, the **Packages** checkbox is not displayed. For information about the **Work Items** tab, see the *HP Time Management Configuration Guide.*	Default: `true` Valid values: `true, false`
ENABLE_TM_ WORK_ITEM_ PROJECTS	If set to `true`, the **Allowed work item types** list on the **Work Items** tab of a time sheet policy includes the **Projects** checkbox. If set to `false`, the **Projects** checkbox is not displayed. For information about the **Work Items** tab, see the *HP Time Management Configuration Guide.*	Default: `true` Valid values: `true, false`
ENABLE_TM_ WORK_ITEM_ REQUESTS	If set to `true`, the **Allowed work item types** list on the **Work Items** tab of a time sheet policy includes the **Requests** checkbox. If set to `false`, the **Requests** checkbox is not displayed. For information about the **Work Items** tab, see the *HP Time Management Configuration Guide.*	Default: `true` Valid values: `true, false`
ENABLE_TM_ WORK_ITEM_ TASKS	If set to `true`, the **Allowed work item types** list on the **Work Items** tab of a time sheet policy includes the **Tasks** checkbox. If set to `false`, the **Tasks** checkbox is not displayed. For information about the **Work Items** tab, see the *HP Time Management Configuration Guide.*	Default: `true` Valid values: `true, false`

Table A-1. Server configuration parameters (page 21 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
ENABLE_TMPM_ SYNC_SERVICE	If set to `true`, enables the TM-PM Sync Service, which synchronizes time sheet updates from HP Time Management to project work plan tasks in HP Project Management, at the interval specified by the TMPM_ SYNC_SERVICE_INTERVAL parameter.	Default: `false` Valid values: `true, false`
ENABLE_UI_ MONITOR	If set to `true`, enables the activity monitor, which captures UI activities (mainly URL requests). If activity exceeds the threshold value (determined by the *UI_MONITOR_ THRESHOLD* parameter), the captured information is output to the `thresholdLog.txt` log file. This file resides in same directory as the server log.	Default: `true` Valid values: `true, false`
ENABLE_UNICODE_ FOR_ TELNETCLIENT	If set to `true`, enables Unicode support in Telnet.	Default: N/A Valid values: `true, false`
ENABLE_ UNNAMED_ RESOURCE_ ALLOCATION	If set to `true`, enables allocation of unnamed resources.	Default: N/A Valid values: `true, false`
ENABLE_WEB_ ACCESS_LOGGING	If set to `true`, tells Tomcat (the Web server provided with JBoss) to log all http requests received. **Note:** If this is enabled on a busy system, Web access logging can generate many log files.	Default: `false` Valid values: `true, false`
ENABLE_WEB_ SERVICES	Enables the PPM Center Web services interface.	Default: `false` Valid values: `true, false`

Table A-1. Server configuration parameters (page 22 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
ENABLE_WORK_ ITEM_MISC	If set to `true`, the **Allowed work item types** list on the **Work Items** tab of a time sheet policy includes the **Miscellaneous Items** checkbox. If set to `false`, the **Miscellaneous Items** checkbox is not displayed. For information about the **Work Items** tab, see the *HP Time Management Configuration Guide.*	Default: `true` Valid values: `true, false`
ENABLE_ WORKBENCH_ NOTIFICATIONS	If set to `true`, the PPM Server receives notifications of completed concurrent requests from the Concurrent Request Watch background service and updates the status on PPM Workbench clients connected to that server. If set to `false`, PPM Workbench clients connected to the server do not receive automatic updates unless the Concurrent Request Watch background service is running on this node.	Default: `true` Valid values: `true, false`
ETL_END_DATE	If PPM Center's Operational Reporting solution is implemented, this determines the end date for the PPM Center data to extract, transform, and load into the Operational Reporting database schema.	Default: N/A Valid values: Calendar date in the format `mm-dd-yyyy.`
ETL_START_DATE	If PPM Center's Operational Reporting solution is implemented, this determines the start date for the PPM Center data to extract, transform, and load into the Operational Reporting database schema.	Default: N/A Valid values: Calendar date in the format `mm-dd-yyyy.`

Table A-1. Server configuration parameters (page 23 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
EV_ALLOW_ PRORATING	If set to `false`, indicates that if a task or project is less than 100 percent complete, the earned value is calculated to be 0. The EV calculation is 100 only if the task or project is 100% complete.	Default: `true` Valid values: `true, false`
EXCEPTION_ DETAIL_LEVEL	Determines the level of detail to include in internal error exceptions displayed in PPM Center. The exception message is included in server logs. This parameter uses a bitwise value combination, as follows: • If bit 1 is set to 1 (001), the message includes exception correlation information. • If bit 2 is set to 1 (010), the message includes the application server node name. • If bit 3 is set to 1 (100), the message includes the filtered stack trace. A value of 7 includes all levels of detail. for the error.	Default: 3 Valid Values: Integer using bitwise value combination (includes correlation information and server node name)
EXCEPTION_ ENGINE_WAKE_UP_ TIME	Determines the time at which the daily exception engine full calculation runs. A full calculation is needed for exceptions that occur as time elapses. The default value of 1 (1:00 AM) specifies that the daily exception calculation is performed once every day at 1:00 AM.	Default: 1 (1:00 AM) Valid Values: Integer between 1 and 24

Table A-1. Server configuration parameters (page 24 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
EXTERNAL_WEB_PORT	If you are using an external Web server to serve PPM Center clients, you must configure this parameter as an available port that can communicate with the PPM Server. This port receives AJP (Apache JServ Protocol) requests from the external Web server. AJP is the standard protocol used for communication between a Web server and an application server. **Note:** If you are using an external Web server, you must still configure the standard PPM Center **HTTP_PORT*. This port is used internally by PPM Center reports. There is no need to make it accessible to the network.	Valid value: Any available port number
FAIL_EXECUTIONS_ON_STARTUP	If the PPM Server stops while command executions are running, those executions are interrupted and the parent entities (package lines, releases, requests, and so on) are assigned the status “in progress.” This parameter tells the server that, after it starts, it must check for any entities that have “in progress” status and that have no executions running (that is, executions that were interrupted). The server sets the internal status of those entities to FAILED, with a visible status of “Failed (Interrupted).”	Default: `true` Valid values: `true, false`
FORECAST_PLANNING_PAGE_SIZE	Number of staffing profile positions displayed on a Forecast Planning page in HP Resource Management.	Default: `50`

Table A-1. Server configuration parameters (page 25 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
FS_QUEUE_ CONCURRENT_ CONSUMERS	Initial number of concurrent instances of the Financial Summary background service.	Default: `1` Valid values: Non-negative integer
FS_QUEUE_MAX_ CONCURRENT_ CONSUMERS	Maximum number of concurrent instances of the Financial Summary background service.	Default: `2` Valid values: Non-negative integer
FULL_NAME_ FORMAT	Format in which the full names are displayed for resources, contacts, and so on.	Default: `0` Valid values: 0, 1 0 denotes First Last. Example: John Smith. 1 denotes Last, First. Example: Smith, John
GRAPHICAL_WF_ ENABLED	If set to `true`, makes links to view graphical workflow available on submitted requests.	Default: `true` Valid values: `true, false`
GROUP_PRIVATE_ PUBLIC_PAGES	If set to `true`, after a user selects **Dashboard > Personalize Dashboard,** the PPM Dashboard displays two sections: **Private** and **Shared.** The **Private** section contains single pages and groups, the **Shared** section contains modules. If set to false, dashboard pages will appear as a flat list.	Default: `true` Valid values: `true, false`

Table A-1. Server configuration parameters (page 26 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
GZIP_ENCODING_ ENABLED	Determines whether HTTP responses are compressed before they are sent to PPM Center HTML clients. If set to `true`, then textual HTTP responses are compressed using GZIP compression (if the requesting browser supports GZIP). By default, this is set to `true` to improve the responsiveness of the PPM Center standard (HTML) interface, because less overall data is carried across the Internet between the client and the PPM Server. If all PPM Center clients have fast network access to the PPM Server, then consider setting this parameter to `false` to reduce the overhead of compressing and decompressing responses.	Default: `true` Valid values: `true, false`
*HEAVY_QUEUE_ CONCURRENT_ CONSUMERS	Number of listeners per node to execute heavy background services.	Default: 1 Valid values: Positive integer
*HEAVY_QUEUE_ MAX_ CONCURRENT_ CONSUMERS	Maximum number of listeners per node to execute heavy background services.	Default: 1 Valid values: Positive integer
*HEAVY_QUEUE_ MAX_DEPTH	Maximum depth of the heavy services queue.	Default: 10000 Valid values: Positive integer
*HEAVY_QUEUE_ REDELIVERY_ DELAY	Delay between redeliveries of messages to the heavy service queue.	Default: 60000 (milliseconds) Valid values: Positive integer

Table A-1. Server configuration parameters (page 27 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
*HEAVY_QUEUE_REDELIVERY_LIMIT	Number of times messages are to be redelivered to the heavy service queue.	Default: 5 Valid values: Positive integer
*HIGH_PAGE_SIZE	Recommended number of work plan lines to load into the Work Plan page if the user is connected through a fast connection such as a LAN.	Default: 100 Valid values: Positive integer
*HISTORY_MENU_SIZE	Number of links to display in the **History** menu in the PPM Center standard interface.	Default: 10 Valid values: Positive integer
*HOURS_TO_KEEP_DEBUG_MESSAGE_ROWS	Number of hours to keep rows in the KNTA_DEBUG_MESSAGES table. For high-volume PPM Center installations, a large number of rows may be generated in this table. For such installations, decrease this value accordingly.	Default: 48 Valid values: Positive integer

Table A-1. Server configuration parameters (page 28 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
*HTTP_PORT	Port to use to communicate with the built-in HTTP server. If PPM Center is in stand-alone mode (that is, it is not integrated with an external Web server), then PPM Center clients must have access to the HTTP_PORT. If PPM Center is integrated with an external Web server, then client HTTP traffic is routed through the *EXTERNAL_WEB_PORT*. However, even in that case, the PPM Server still uses the **HTTP_PORT* internally to run reports. In this case, it is not necessary to make the **HTTP_PORT* externally accessible to PPM Center clients (and thus, the port need not be exposed outside of the PPM Server). **Note:** If you are integrating PPM Center with Application Lifecycle Management, then you must set the parameter to a number less than 32767.	Default: `8080` Valid values: Unique port greater than 1024 and distinct from the Web server, SQL*Net, and RMI ports.
HTTPS_CIPHERS	Specifies the ciphers for the SSL/ TLS protocol with which the PPM Server negotiates. This parameter accepts a comma-separated list of ciphers that the server is to allow for SSL/TLS connections. You can use the parameter to limit the cipher suite to a set of specific strong ciphers.	Default: N/A Valid values: comma-separated list of ciphers
HTTPS_ KEYPASSWORD	Keystore password (encrypted). This setting is required if the ENABLE_ SSL_LOGIN parameter is set to `true`.	Default: N/A Valid values: N/A

Table A-1. Server configuration parameters (page 29 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
HTTPS_KEYSTORE_ LOCATION	Full path location of the keystore. This parameter setting is required if the *ENABLE_SSL_LOGIN* parameter is set to `true`.	Default: N/A Valid values: N/A
HTTPS_PORT	HTTPS port to use for SSL Login. This parameter setting is required if the ENABLE_SSL_LOGIN parameter is set to `true`.	Default: `none` Valid values: Must be the HTTP_PORT number + 363
HTTPS_PROTOCOL	Specifies the HTTPS protocol (SSL or TLS) the PPM Server uses.	Default: `TLS` Valid values: `SSL, TLS`
HTTPS_WEB_ THREAD_MAX	Maximum number of HTTPS threads. This parameter setting is required if the ENABLE_SSL_LOGIN parameter is set to `true`.	Default: 75 Valid values: Positive integer
HTTPS_WEB_ THREAD_MIN	Minimum number of HTTPS threads. This parameter setting is required if the ENABLE_SSL_LOGIN parameter is set to `true`.	Default: 5 Valid values: Positive integer
I18N_CARET_ DIRECTION	Caret position on input fields (for example, text fields). If unspecified, the system uses the value specified for *I18N_SECTION_ DIRECTION* on page 422.	Default: `ltr` Valid values: `ltr, rtl` (left to right, right to left)
I18N_ENCODING	Character encoding to be used on all HTML pages in the PPM Center standard interface.	Default: `UTF-8`

Table A-1. Server configuration parameters (page 30 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
I18N_LAYOUT_ DIRECTION	Default layout direction of HTML pages in the PPM Center standard interface.	Default: ltr Valid values: `ltr, rtl` (left to right, right to left)
I18N_REPORT_ HTML_CHARSET	HTML character set used to generate PL/SQL reports. Must map to the character set specified for *I18N_REPORTS_ ENCODING* on page 422.	Default: `WE8ISO8859P15` Valid values: Any character set names that Oracle recognizes.
I18N_REPORTS_ ENCODING	Character encoding to use to generate reports in PPM Center. HP recommends `IW8MSWIN1255` for Windows systems.	Default: `UTF-8` Valid values: Any encoding algorithm that Oracle can interpret.
I18N_SECTION_ DIRECTION	Layout direction of custom sections (for example, request detail sections). If unspecified, the system uses the value specified for *I18N_LAYOUT_ DIRECTION* on page 422.	Default: `ltr` Valid values: `ltr, rtl`
*INSTALLATION_ LOCALE	Language and country code of the PPM Center installation. The language code must match the PPM Center installation language.	Default: `en_US` Valid values: PPM Center installation language code
JAVA_CLASSES_ LOC	Location of the JRE classes.	Example `C:/Java/ j2sdk1.6/jre/ lib/ classes.zip`

Table A-1. Server configuration parameters (page 31 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
JAVA_PLUGIN_CLASSID	Class ID for the Java plugin used for the PPM Workbench.	Default: `CAFEEFAC-0016-0000-0007-ABCDEFFEDCBA`
JAVA_PLUGIN_PATH_IE	Web location for downloading the cross-platform Java plug-in bundle for Internet Explorer browsers.	Default: `http://java.sun.com/update/1.6.0/jinstall-1_6_0_7-windows-i586.cab`
JAVA_PLUGIN_PATH_NS	Web location for downloading the cross-platform Java plug-in installer for Netscape browsers.	Example `http://java.sun.com/products/archive/j2se/1.4.1_07/`
JAVA_PLUGIN_VERSION	Earliest version of the Sun Java plug-in used to start the PPM Workbench.	Default: `1.6.0_7`
JAVA_PLUGIN_XPI_PATH	Web location for downloading the cross-platform Java plug-in installer for Firefox browsers. For information about the Java plug-in supported for the current PPM Center version, see the *System Requirements and Compatibility Matrix.*	Example `http://java.sun.com/update/1.6.0/j2re-1_6_0_7-windows-i586.xpi`
JDBC_DEBUGGING	Enables debugging of the Java database calls.	Default: `false` Valid values: `true, false`

Table A-1. Server configuration parameters (page 32 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
*JDBC_URL **Note:** For Oracle RAC (Real Application Clusters), this parameter must contain the host and port information for all databases to which the PPM Server connects.	Locator for the database that contains the PPM Center database schema. Must be specified correctly for PPM Server to communicate with the database. Format: `jdbc:oracle.thin:@<Host_Name>:<Port>:<SID>` where • *<Host_Name>* is the DNS name or IP address of the system running the database • *<Port>* is the port used by SQL*Net to connect to the database. Refer to the database entry in the `tnsnames.ora` file • *<SID>* is the database system ID.	Default: `1521` Example `jdbc:oracle:thin:@DBhost.domain.com:1521:SID`
JNDI_LOOKUP_PORT	For a server cluster deployment, the JNDI_LOOKUP_PORT parameter is the port to which the HA-JNDI server binds as it waits for JNP clients. Its value is taken from the ***APP_SERVER_HAJNDI_BINDING_PORT* parameter. For a stand-alone deployment, the JNDI_LOOKUP_PORT parameter is the JBoss Naming Service Binding port number, and its value is taken from the ***APP_SERVER_NAMING_SERVICE_BINDING_PORT* parameter.	N/A
*JOB_STATUS_CLEANUP_INTERVAL	Timeout (in minutes) for cleaning the `JOB_STATUS` table.	Default: `720` (minutes) Valid values:

Table A-1. Server configuration parameters (page 33 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
JVM_OPTIONS	For HP internal use only. Do not alter its value unless directed to do so by HP Software Support for PPM Center.	Default: N/A
JSP_RECOMPILE_ ENABLED	Determines whether changes to JSP files are picked up on a running server, thereby quickly making them visible. If set to `false`, JSP files are checked for changes only the first time they are accessed, with the result that changes are visible only after the server is restarted.	Default: `false` on production systems, `true` on development systems Valid values: `true, false`
KEEP_ALIVE_ INTERVAL	Frequency with which the client sends keep-alive messages to the PPM Server.	Default: 2
**KINTANA_LDAP_ID Required if *AUTHENTICATION_ MODE = LDAP	PPM Center account on the LDAP server. Used by the PPM Server to bind to the LDAP server.	Default: N/A Examples: `uid=admin, ou=dev, cn=Users`
**KINTANA_LDAP_ PASSWORD Required if *AUTHENTICATION_ MODE = LDAP	PPM Center password on the LDAP server. The PPM Server configuration utility automatically encrypts this password. To manually edit this value, surround the encrypted password with `#!#` delimiters.	Default: `#!##!#` Format: `#!#<Password>#!#`
KINTANA_LOGON_ FILENAME	Used in non-HTML notification, this parameter value is specified with the filename (to be appended to the URL), which points to the logon page. **Note:** HP recommends that you not reset this parameter.	Example: `kintanaHome.html`

Table A-1. Server configuration parameters (page 34 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
KINTANA_SERVER_DIRECTORY	Server directory location. Set this value if you have a clustered server setup.	Default: `PPM_910_ALL`
KINTANA_SERVER_LIST	The server sets the (read-only) value of this parameter at runtime.	Example: `aeon!rmi://ice:27099/KintanaServer`
*KINTANA_SERVER_NAME	Name of the PPM Server instance. If multiple PPM Servers are running on the same machine, this name must be unique for each server. If the server is running Windows, this name must match the name of the Windows service name.	Default: `PPM_910_ALL`
*KINTANA_SESSION_TIMEOUT	Time set to elapse before the PPM Server terminates a user session (in the PPM Workbench or standard interface) because of inactivity. A value of 0 denotes no timeout.	Default: `120` (minutes) Valid values: `10` through `720`
LDAP_BASE_DN	Base distinguished name on the LDAP server. Each LDAP URL must specify a base distinguished name (DN), which is used in place of the LDAP_BASE_DN server configuration parameter. If the URLs provided for *LDAP_URL_FULL* do not have a DN value, PPM Center uses the value set for LDAP_BASE_DN.	Default: N/A Examples: `CN=Users,DC=PPMAD,DC=com`

Table A-1. Server configuration parameters (page 35 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
LDAP_ENABLE_ DEREFERENCING	If set to `yes`, enables LDAP dereferencing. For more details about LDAP dereferencing, see "Dereferencing Aliases" on Oracle's Sun Developer Web site (java.sun.com/products/jndi/tutorial/ldap/misc/aliases.html).	Default: `yes` Valid values: `yes, no`
**LDAP_GROUP_ RECURSION_LIMIT Required if *AUTHENTICATION_ MODE = LDAP	Number of levels of subgroups to traverse when importing users from groups.	Default: 15
LDAP_KEYSTORE	LDAP keystore.	Default: N/A
LDAP_KEYSTORE_ PASSWORD	LDAP keystore password.	Default: N/A
LDAP_LAST_ SYNCH_ TIMESTAMP	Last time the LDAP import was run.	Default: N/A
LDAP_REFERRAL_ CHASE	If set to `true`, enables the LDAP server to follow referrals.	Default: `false` Valid values: `true, false`
LDAP_REFERRAL_ HOP_LIMIT	Maximum number of referral hops that the LDAP libraries can follow.	Default: 10
LDAP_SERVER_ TYPE	Type of LDAP server used.	Default: N/A
**LDAP_SSL_PORT Required if *AUTHENTICATION_ MODE = LDAP	SSL port number on the LDAP server. If not specified, all transactions are carried over the port specified for **LDAP_URL on page 428.	Default: 636

Table A-1. Server configuration parameters (page 36 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values

**LDAP_URL
Required if *AUTHENTICATION_MODE = LDAP
	Comma-delimited list of LDAP URLs, which the PPM Server queries in the order specified.
	If no port number is specified, the default port number 389 is used.
	Note: The `LDAP_URL_FULL` parameter supersedes the `LDAP_URL` parameter. That is, if a value is set for both in the `server.conf` file, `LDAP_URL_FULL` is used. If the URLs specified for `LDAP_URL_FULL` do not have a DN value, the value set for `LDAP_BASE_DN` is used.
		Format
		`ldap://ldap.<URL>.com:389`
		Example
		`ldap://10.100.102.199:389`

LDAP_URL_FULL	PPM Center uses this parameter to handle multiple domains during LDAP authentication. The values for the parameter include a space-separated (not comma-separated) list of full LDAP URLs. Each LDAP URL must specify a base DN. **Notes:** • To specify a space character inside a URL, use the URL encoding scheme, and replace the space with “%20.” For example, if you have an organizational unit called “My Org Unit,” then specify”My%20Org%20Unit” in the LDAP URL. • The `LDAP_URL_FULL` parameter supersedes the `LDAP_URL` parameter. That is, if a value is set for both in the `server.conf` file, `LDAP_URL_FULL` is used. If URLs specified for `LDAP_URL_FULL` do not have a DN value, the value set for `LDAP_BASE_DN` is used.	Example `com.kintana.core.server.LDAP_URL_FULL=ldap://host.yourdomain.com/CN=Users,DC=yourdomain,DC=com ldap://host.yourdomain.com/OU=Users2,DC=yourdomain,DC=com`

Table A-1. Server configuration parameters (page 37 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
LICENSE_KEY	License key required to use PPM Center core functionality.	Example `36ha5b993c1776kc6g03gjct5k7hv5c3`
*LIGHT_QUEUE_CONCURRENT_CONSUMERS	Number of listeners per node to execute light-weight background services.	Default: `1` Valid values: Positive integer
*LIGHT_QUEUE_MAX_CONCURRENT_CONSUMERS	Maximum number of listeners per node to execute light background services.	Default: `3` Valid values: Positive integer
*LIGHT_QUEUE_MAX_DEPTH	Maximum depth of the light services queue.	Default: `10000` Valid values: Positive integer
*LIGHT_QUEUE_REDELIVERY_DELAY	Delay between redeliveries of a message to the light service queue.	Default: `60000` (milliseconds) Valid values: Positive integer
*LIGHT_QUEUE_REDELIVERY_LIMIT	Number of times a message can be redelivered to the light service queue.	Default: `5` Valid values: Positive integer
LOAD_FACTOR	Determines how much load to place on a node in the server cluster.	Default: `1.0` Valid values:

Table A-1. Server configuration parameters (page 38 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
LOCAL_IP	Name of the machine running the firewall. This parameter applies only to RMI traffic for the PPM Workbench. Before you set this parameter, register the external IP address on the external DNS server, and then specify the name of the machine running the firewall as the LOCAL_IP value. If you set this up correctly: • Client A running inside the firewall connects to the internal DNS server and the machine name resolves to an IP address. • Client B running outside the firewall connects to an external DNS server and the machine name resolves to a different IP address. Both clients can then connect, each to a different IP address.	Example `10.1.101.64`
	Note: Setting the `LOCAL_IP` parameter resolves the following potential problems: • If set to the IP address of the machine running the firewall, clients inside the firewall can connect, but clients outside cannot, because they have no route to the host. • If set to the name of the machine running the firewall, clients inside the firewall can connect, but clients outside cannot, because they cannot resolve the hostname. • If set to an IP address that is different from the machine running the firewall, clients outside the firewall can connect, but clients inside the firewall cannot, because the address is not translated between a different IP address to the IP address on the machine running the firewall.	

Table A-1. Server configuration parameters (page 39 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
LOGIN_COOKIE_ MAX_AGE	Maximum age (and thus the expiration) of cookies used to start a PPM Center session.	Default: `180` (days) Valid values:
LOGON_METHOD	Method used to log on to PPM Center.	Default: `USER_NAME` Valid values:
LOGON_PAGE	URL for the PPM Center logon page.	Default: `/web/knta/global/Logon.jsp`
*LOGON_TRIES_ INTERVAL	Interval (in minutes) during which logon attempts are monitored.	Default: `1` (minutes) Valid values: Positive integer
*LOW_PAGE_SIZE	Number of work plan lines to load into the Work Plan page if the user is connected through a slow connection such as a WAN.	Default: `20` Valid values: Positive integer
MAC_LOG_ SEVERITY [a]	Logging level to use. If set to 0, only integration exceptions and a summary are logged. If set to 1, events other than errors related to processing changes are also logged.	Default: `1` Valid values: `0` and `1`
MAX_BATCH_ TIMESHEET_ FREEZE_CLOSE	Maximum number of time sheets that can be frozen or closed at one time is the *lower* of: • Value of this parameter, which has no default value, and a recommended value of 50. • Value of the **Results Displayed Per Page** field on the Search for a Time Sheet to Freeze/Close page (which has a default value of 50).	Default: None Valid values: Integer

Table A-1. Server configuration parameters (page 40 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
MAX_BATCH_ TIMESHEET_LINE_ APPROVE	Maximum number of time sheets that can be approved at one time is the *lower* of: • Value of this parameter, which has no default value, and a recommended value of 50. • Value of the Results **Displayed Per Page** field if using the Approve Time Sheets page (which has a default value of 50), or **Rows Displayed** field if using the Approve Time Sheets portlet (which has a default value of 5).	Default: None Valid values: Integer
MAX_BUBBLE_ CHART_RESULT	Maximum number of results to display in bubble charts.	Default: `500`
MAX_BUBBLE_ CHART_SIZE	Maximum number of bubbles (entities) that can be displayed in a bubble chart.	Default: `500` Valid values: Integer
MAX_DB_ CONNECTION_ IDLE_TIME	Amount of time (in minutes) that an unused database connection stays open before it is closed and removed from the pool.	Default: `60` (minutes)
MAX_DB_ CONNECTION_ LIFE_TIME	Amount of time that a database session is held open before it is closed and removed from the pool. Some Oracle cleanup operations that should be run periodically occur only at the end of database sessions. Therefore, do not keep database sessions open for the life of the PPM Server.	Default: `1440` (minutes)
MAX_DB_ CONNECTION_ WAIT_TIME	Amount of time that the system waits before it times out a request for a database connection.	Default: `180` (seconds)

Table A-1. Server configuration parameters (page 41 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
MAX_DB_ CONNECTIONS	Maximum size of each of the two database connection pools that the PPM Server creates. Each user does not get a dedicated connection. The server uses connection pooling, so it opens a new database connection only if no connections are available in the pool. After this number is reached, user sessions queue for the next available database connection.	Default: `60`
*MAX_EXECUTION_ MANAGERS	Maximum number of concurrent executions allowed to run on the server. If your system is heavily loaded, decreasing this may help reduce load, but may also delay execution of tasks. If your organization processes a high volume of packages, you may require more execution managers.	Default: `15`
*MAX_LOGON_ TRIES	Maximum number of logon attempts in the time interval specified for **LOGON_TRIES_INTERVAL* on page 431.	Default: `0`
*MAX_PAGE_SIZE	Absolute maximum number of work plan lines that can be loaded into the Work Plan page. Use this parameter to prevent excessive load on the server from excessive queries, and to prevent users from getting themselves into low performance situations.	Default: `500`

Table A-1. Server configuration parameters (page 42 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
*MAX_RELEASE_ EXECUTION_ MANAGERS	Number of command executions that can run in a release distribution simultaneously. Organizations that process a high volume of packages may require a larger number of release execution managers.	Default: `15` Valid values: Integer greater than 1
MAX_SERVER_ CONF_BACKUPS	Number of `server.conf` file backups to be maintained when application configuration parameters are updated through the Administration Console.	Default: `1` Positive integer
MAX_STATEMENT_ CACHE_SIZE	Maximum number of prepared statements cached per database connection. Part of the database connection pool settings.	Default: `50` Valid values: Integer greater than 0
MOBILITY_ ACCESS_BATCH_ SIZE	Number of emails that the Mobility Access service fetches in a single batch.	Default: `100`
MOBILITY_ ACCESS_FETCH_ TIMEOUT	Amount of time the PPM Server tries to connect to the email account before it times out.	Default: `3`
MOBILITY_ ACCESS_SERVICE_ INTERVAL	Number of minutes the Mobility Access service is to wait after the start time or the last batch sent, before sending out the next batch of email notifications.	Default: `5`
MOBILITY_ ACCESS_THREAD_ COUNT	Number of threads that the Mobility Access service uses to process emails.	Default: `1`

Table A-1. Server configuration parameters (page 43 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
MSP_NOTES_SIZE_LIMIT	Maximum size of Microsoft Project notes in PPM Center. Notes larger than the size specified for this parameter are truncated when MSP data is synchronized with PPM Center.	Default: `2 MB`
MSP_PROJECT_CUSTOM_FIELD	Microsoft Project includes a set of pre-defined text fields (Text1, Text2, Text3, and so on) that users can use to store whatever they want. By default, PPM Center uses the Text30 field at the project level in Microsoft Project to store information about the PPM Center project with which the Microsoft Project file is associated. If you already use the Text30 field in Microsoft Project, you can use this parameter to specify the Microsoft Project field for PPM Center to use. **Warning:** Do not change this parameter value after the Microsoft Project integration has been used.	Default: `Text30`
MSPS_RESOURCE_CUSTOM_FIELD	Used to specify which enterprise custom resource field name is to be used by the Plug-in for PPM during enterprise resource mapping. Make sure that you specify a value for this parameter for each resource on all MSP Servers, and that the value uniquely identifies a resource across all MSP Servers. Use this parameter *only* when working with multiple MSP servers.	Default: none

Table A-1. Server configuration parameters (page 44 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
MULTICAST_ CLUSTER_NAME	Unique name of a PPM Server cluster. This parameter, along with the APP_ SERVER_MULTICAST_PORT, MULTICAST_IP, and MULTICAST_ PORT parameters, determines whether PPM Center starts in server cluster mode, or stand-alone mode. **Warning:** Do not configure two clusters with the same name running on the same subnet. **Warning:** The IP address you specify for MULTICAST_CLUSTER_NAME must not include the text string `"http://"`.	Example `server. mydomain.com/ ppm`
MULTICAST_DEBUG	Determines whether or not incoming and outgoing multicast messages are logged to the PPM Server log.	Default: `false` Valid values: `true, false`
MULTICAST_IP	Multicast IP address. This parameter, along with the APP_SERVER_ MULTICAST_PORT, MULTICAST_ CLUSTER_NAME, and MULTICAST_ PORT parameters, determines whether PPM Center starts in server cluster mode, or stand-alone mode. **Warning:** The IP address you specify for MULTICAST_IP must not include the text string `"http://"`.	Default: N/A Valid values: `224.0.0.0` through `239.255.255.2 55`
MULTICAST_ LEASE_MILLIS	Amount of time that must elapse after the PPM Server heartbeat stops, before the PPM Server is considered terminated.	Default: `60000` (milliseconds)
MULTICAST_PORT	Multicast port used by PPM Center's Cluster Monitor. You can specify any unused port number that does not conflict with other multicast ports.	Default: `9000` Valid values: Number for any unused port

Table A-1. Server configuration parameters (page 45 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
MY_LINKS_MAX_COUNT	Maximum number of links a user can add to the **My Links** menu in the standard interface.	Default: `100` Valid values: Positive integer

**NLS_DATE_<*NLS_LANGUAGE*>	Language used to display dates, by locale, on a multilingual PPM Center system. You can specify NLS_DATE_<*NLS_LANGUAGE*> using all languages installed on a PPM Center instance.
	Although values are set during installation when the administrator selects the languages to install, the administrator can also add these values to the `server.conf` file manually.
	For example, if you install Korean and Brazilian Portuguese languages after you install PPM Center, you would add the following to the `server.conf` file:
	`com.kintana.core.server.NLS_DATE_KOREAN=GENERIC_M`
	and
	`com.kintana.core.server.NLS_DATE_BRAZILIAN_PORTUGUESE=GENERIC_M`
	For guidance on what values to set for this parameter, see the *Oracle Database Globalization Support Guide* (http://www.oracle.com/technology/software/index.html).
		Valid values: Any Oracle-supported values

Table A-1. Server configuration parameters (page 46 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values

NLS_SORT_<*NLS_LANGUAGE*>	Sort order of search results displayed on a multilingual PPM Center system. You can specify NLS_SORT_<*NLS_LANGUAGE*> values for all of the languages installed on a PPM Center instance.
	Although values are set during installation when the administrator selects the languages to install, the administrator can also add these values to the `server.conf` file manually.
	For example, if you install Korean and Brazilian Portuguese languages after you install PPM Center, you would add the following to the `server.conf` file:
	`com.kintana.core.server.NLS_SORT_KOREAN=GENERIC_M`
	and
	`com.kintana.core.server.NLS_SORT_BRAZILIAN_PORTUGUESE=GENERIC_M`
	For guidance on what values to set for this parameter, see the *Oracle Database Globalization Support Guide* (http://www.oracle.com/technology/software/index.html).
		Valid values: Any Oracle-supported values

NLS_TERRITORY<*NLS_LANGUAGE*>	Oracle defaults for a territory by Java locale.	Valid values: Any Oracle-supported values
NOTIFICATIONS_CLEANUP_PERIOD	Number of days that notifications remain in the system before the Notifications Cleanup Service removes them.	Default values: 7 Valid values: Integer greater than 1.

Table A-1. Server configuration parameters (page 47 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
OPTIMIZATION_ ITERATION_ MULTIPLIER	Number of algorithmic iterations that the optimization engine is to run. The more iterations, the more time is given to finding an optimal portfolio. Although the default is adequate in most instances, complex cases can benefit from more iterations. **Note:** This parameter also affects generation of the Efficient Frontier curve.	Default: `100` (iterations)
OPTIMIZER_ NUMBER_OF_ TIMESHIFTS	Maximum number of periods the optimizer can shift start dates forward. This does not affect manually-shifted HP Portfolio Management entities. If you allow a new start date for a project, the optimizer can start the project any time between the original start date and six months beyond that date.	Default: 6 (months)
**ORACLE_APPS_ ENABLED	Determines whether PPM Center is to be integrated with Oracle Apps. You must set this parameter to `true` for installations using HP Deployment Management to integrate with Oracle Apps through HP Object Migrator or HP GL Migrator. If you use Object Migrator with PPM Center, you must specify a value.	Default: `false` Valid values: `true, false`
ORACLE_APPS_ VERSION	Version of Oracle Apps used. For releases R11, R11i, and R12, specify R11.	Default: `R11` Valid values: `R11` for any Oracle Apps release 11 or later.

Table A-1. Server configuration parameters (page 48 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
ORACLE_DB_ VERSION	The server sets this read-only parameter value during startup.	Example `10.2.0.4.0` Valid values: Any supported Oracle database software version
*ORACLE_HOME	Full path to the Oracle home directory on the PPM Server. The Oracle_Home/network/admin directory must contain the correct TNS names (or a file containing the names such as `tnsnames.ora`) required to connect to the PPM Center database schema.	Example `d:/orant`
PACKAGE_LOG_DIR	Directory to which PPM Center writes package output. In a server cluster, if you have overridden the default value for this parameter to refer to a different directory, then all servers in the cluster must be able to access and share the directory.	Default: Same default value as the *BASE_LOG_ DIR* parameter
PACKAGE_LOG_ EXT	Extension used for package log files.	Default: `html`
PACKAGE_LOG_ HEADER	Prefix used for package log file names.	Default: `PKG_`
PAGE_PDF_ EXPORT_DISABLED	To disable the Export to PDF feature, add this parameter to the `server.conf` file, and set its value to `true.`	Default: N/A Valid values: `true, false`

Table A-1. Server configuration parameters (page 49 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
PARTITION_NAME	Logical group name assigned to a cluster of JBOSS application servers. Any node in the server cluster that is started with the same partition name becomes part of the cluster. The value for this parameter is set during installation. The startup script pulls the specified partition name from the `server.conf` file and uses it to start the PPM Server.	Default: N/A
*PASSWORD_ EXPIRATION_DAYS	Expiration period of passwords for new users. A value of 0 indicates no expiration.	Default: `0` (days) Valid values: 0 through 366
*PASSWORD_ REUSE_ RESTRICTION_ DAYS	Number of days to restrict the reuse of an old password from the last date the password changed. A value of 0 indicates no restriction.	Default: `0` Valid values: 0 through 2192
PDF_FONT_FILE	Absolute file path to the font used in exporting work plans in PDF file format.	Default: N/A Valid values: *<Absolute_ Font_File_ Path>*
PENDING_STATUS_ CHANGE_SERVICE_ POOL_SIZE	Size of the thread pool for the Request Status Change service.	Default: 5 Valid values:
PGA_AGGREGATE_ TARGET	Maximum physical memory Oracle can use for working areas for all processes together. See also *WORKAREA_SIZE_POLICY* on page 465.	Maximum number of MB to dedicate to working Oracle processes
PLUGINS	Semicolon-separated list of plugins used by the PPM Server.	Default: N/A

Table A-1. Server configuration parameters (page 50 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
*PM_CAN_ROLLUP_ACTUALS_ON_SAVE	If set to `true`, enables the user to change the mode on HP Project Management's Enter Actuals page. The resulting options are either rollups calculated during save processing, or rollups are deferred (and rolled up using the associated service).	Default: `true` Valid values: `true, false`
*PM_NUM_EDIT_ASGMTS	Maximum number of assignments that can be displayed for editing on the Enter Actuals page in HP Project Management.	Default: `200`
PORTFOLIO_MANAGEMENT_LICENSE_KEY	License key required to use HP Portfolio Management. This key is delivered in the `license.conf` file, which you can find in the `<PPM_Home>/conf` directory after PPM Center installation.	Default: N/A Example `9g54th33f7510 8l96d88fe7d16 390c52`
PORTLET_EXEC_TIMEOUT	Amount of time (in seconds) a portlet's SQL statement is to run before it is automatically disconnected from the database. This parameter is used to limit long-running queries in portlets, which may be caused by adding portlets without filtering criteria. Used to avoid excessive database CPU processing when users end their sessions before processing has completed. **Note:** Increase the value of `PORTLET_EXEC_TIMEOUT` only as a last resort. This setting has system-wide performance impact. If you must increase the value, specify a value such as 30, and not a high value such as 200.	Default: `20` (seconds)

Table A-1. Server configuration parameters (page 51 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
PORTLET_MAX_ ROWS_RETURNED	Maximum number of rows to display in portlets.	Default: `200`
PORTLET_ MONITOR_ THRESHOLD	If the `ENABLE_PORTLET_MONITOR` parameter is set to `true`, then this parameter determines the portlet load time threshold (in milliseconds) above which the portlet monitor logs portlet load information.	Default: (milliseconds)
PROJ_COST_ROLL_ UP_DURATION_IN_ DAYS	Determines the maximum duration of a project (based on start and finish dates of root task in the assigned work plan), on which cost roll-up calculations are performed. The default is optimized for performance.	Default: `3650` (days)

Table A-1. Server configuration parameters (page 52 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
PV_USE_ACTIVE_BASELINE_DATES	If set to true, enables users to use the Planned Value (PV) calculation algorithm, which uses projects' active baseline dates instead of their scheduled dates. When the Project Planned Value Update service runs, and the `PV_USE_ACTIVE_BASELINE_DATES` flag is set to `true`, PPM Center checks for and uses active baseline dates instead of scheduled dates for planned value calculations on all projects. The PV calculation formula is: `PV = Baseline Cost * {MIN (Today's Date, Baseline Finish Date) - Baseline Start Date} / (Baseline Finish Date - Baseline Start Date)` Where `Baseline Cost = Planned Labor + Planned Non-Labor` When the Project Planned Value Update service runs the first time after this flag is turned on, projects with scheduled dates and active baseline dates that are past are added to the queue for PV recalculation.	Default: `false` Valid values: `true, false`
*QUARTZ_WORKER_THREADS	Number of threads to be invoked per PPM Server node to send messages.	Default: `10` Valid values: Integer greater than zero

Table A-1. Server configuration parameters (page 53 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
REMOTE_ADMIN_REQUIRE_AUTH	Determines whether user authentication is required for remote administration of the PPM Center instance. If set to `true`, users running `kStop.sh` to shut down the PPM Server must supply a valid PPM Center user name and password. If set to `false`, any user with access to `kStop.sh` can shut down the server.	Default: `true` Valid values: `true, false`
REPORT_DIR	If you want report output to be written to a location other than the default directory (outside of the PPM Server directory structure), use this parameter to specify a different directory. Make sure that the PPM Server has access to the directory so that the report output HTML files can be written here.	Example `D:/<PPM_Home>/910/aeon/reports/`
REPORT_LOG_DIR	Directory in which the PPM Center report logs are stored. **Note:** In a server cluster, if you have overridden the default value for this parameter to refer to a different directory, then all nodes in the cluster must be able to access and share the directory.	Same default value as the *BASE_LOG_DIR* parameter Example `D:/<PPM_Home>/910/aeon/logs/reports/`
REPORTING_STATUS_REFRESH_RATE	Frequency with which report status is refreshed and displayed to the user.	Default: 5 (seconds)

Table A-1. Server configuration parameters (page 54 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
REQUEST_LOG_DIR	Location for Request execution log output. (Logs directory directly under the directory specified by the *BASE_PATH parameter.) **Note:** In a server cluster, if you have overridden the default value for this parameter to refer to a different directory, then all nodes in the cluster must be able to access and share the directory.	Example `D:/PPM/910/aeon/logs/`
REQUEST_SEARCH_RESULTS_MAX_ROWS	Maximum number of results returned by a search. The value is displayed as the default in the **Limit Rows Returned To** field.	Default: `1000`
REQUEST_TYPE_CACHE_TIMEOUT	Stale check timeout for the cache that maintains mappings between parameters and tokens for Request Type and Request Header Type. **Note:** HP strongly recommends that you not change the value of this parameter.	Default: `3600` (seconds)
RESOURCE_FINDER_ROLE_WEIGHT	Value used to calculate the suitability score for items returned on the Resource Finder results page.	Default: `25` Valid values: `0` through `100`
RESOURCE_FINDER_SEARCH_MAX_USERS	Maximum number of resources that can be targeted in a user search. If the targeted number exceeds this value, the Resource Finder displays the message that the number of resources targeted is too large.	Default: `100`
RESOURCE_FINDER_SKILL_WEIGHT	Value used to calculate the suitability score for items returned on the resource finder results page.	Default: `25` Valid values: `0` through `100`

Table A-1. Server configuration parameters (page 55 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
RESTRICT_ BYPASS_ EXECUTION_TO_ MANAGERS	Determines whether only managers can bypass execution of workflow steps in packages. If set to `true`, only users with an access grant of Package Manager or Request Manager access can bypass executions. If set to `false`, all users eligible to act on executions can bypass them.	Default: `false` Valid values: `true, false`
RESTRICT_ BYPASS_REQ_ EXEC_TO_ MANAGERS	Restricts bypass execution to request managers. If set to `true`, only a user with the Manage Request access grant can bypass an execution step on a request.	Default: `false` Valid values: `true, false`
RM_ALLOWED_ EFFORT_TYPES	Determines the effort types allowed for HP Resource Management staffing profile and other modules. You can specify a combination of up to three comma-delimited values. Examples • `fte, person_days, hours` • `fte, person_days` • `fte, hours` • `person_days, hours` The order does not matter.	Default: `fte, person_days` Valid values: `hours, fte, person_days`
*RM_DEFAULT_ EFFORT_TYPE	Effort type used to display staffing profiles and resource pool information.	Default: `fte` Valid values: `hours, fte, person_days`

Table A-1. Server configuration parameters (page 56 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
*RM_DEFAULT_ PERIOD_TYPE	Default period type used to display staffing profiles and resource pool information.	Default: `month` Valid values: `quarter, month, week, year`
*RM_MAX_ RESOURCE_IN_ POOL	Maximum number of resources in a resource pool. If the number of resources exceeds this value, some features are unavailable on the Resource Pool overview page. The **View Resource Load** button is not available if the number of resources in that resource pool (or its hierarchy if the “Include children resource pools when calculating load for this resource pool” flag is selected) exceeds the value set for this parameter. The **View Forecasted Demand** and **Manage Pool Capacity** button are also unavailable if the number of resources in the resource pool exceeds the default. However, you can still use the Resource page Manage Participation feature to add or remove resources. Values greater than the default (250) may increase response times and memory footprint when the above operations are performed.	Default: `250`

Table A-1. Server configuration parameters (page 57 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
RM_OVERRIDE_ ASSIGNMENT_ CONTROLS	If set to `true`, this parameter turns off security during allocation of a resource to a staffing profile or during assignment of a resource to a work plan. A project manager can then directly assign any resource (in a resource pool) to the staffing profile or the work plan, or to both. The project manager can also use the resource finder to locate and assign resources in all resource pools.	Default: `false` Valid values: `true, false`
*RMI_URL	Port on which the PPM Server listens to initiate RMI client/server communication. Must be a unique port, distinct from the Web server, SQL*Net, and the HTTP or HTTPS ports. Format: `rmi://<Host_Name>:<Port>/ KintanaServer`	Default: `1099` Valid values: Port numbers higher than 1024 Example `rmi:// gold.ppm.com: 1099/ PPMServer`
RMI_VALIDATE_ SERVER_ CERTIFICATE	Used if PPM Server is running in secure RMI mode. If set to `true`, the client PPM Workbench validates the server certificate against the Certificate Authorizer's to verify server identity. If set to `false`, the certificate is not validated.	Default: `false` Valid values: `true, false`
*RML_PASSWORD	Password for the Oracle schema name specified for **RML_ USERNAME*.	Valid values: Encrypted password in the format `#!#<Password> #!#`

Table A-1. Server configuration parameters (page 58 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
*RML_USERNAME	Oracle schema name for the meta layer schema. Must be the same as the database schema name used during installation.	Valid values: Any user name format that Oracle supports
SCPCLIENT_ TIMEOUT	Amount of time after which SCP clients must provide feedback after a file transfer has initiated, else a timeout occurs. Set to the maximum expected time for file transfer.	Default: `10000` (milliseconds)
SDI_LOG_SEVERITY	Level of detail included in Service Desk Integration (SDI) error logs. To log only errors, specify the value 0. To log both errors and information, specify the value 1.	Valid values: `0` and `1`
SDI_SERVICE_ INTERVAL	Frequency (in seconds) with which the SDI service is run.	Default: `900` (seconds)
SEARCH_TIMEOUT	Number of seconds after which searches time out. Used to limit long-running queries in searches, which may be caused by submitting a search without specifying selective data. Avoids taking up database CPU if a user ends a session before the search is completed.	Default: `60` (seconds)
SECURE_RMI	If set to `true`, enables encryption of RMI network traffic between PPM Workbench clients and the PPM Server.	Default: `false` Valid values: `true, false`

Table A-1. Server configuration parameters (page 59 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
SERVER_ENV_ NAME	Name of the PPM Center environment that contains information about the PPM Server machine (for example, host name, user name, and password). Must be set before PPM Center entity migrators or commands involving secure copy can run.	Default: `KINTANA_ SERVER`
SERVER_MAX_ PERM_SIZE	For HP internal use only. Do not change its value unless directed to do so by HP Software Support for PPM Center.	Default: N/A
SERVER_MODE	Server mode to use in case you want exclusive access to a running server.	Default: `NORMAL` Valid values: `Normal, Restricted, Disabled`
*SERVER_NAME	DNS name or IP address of the machine hosting the PPM Server.	Default: `kintana` Valid values: Any valid machine name
SERVER_TYPE_ CODE	Operating system on which the PPM Server is installed.	Valid values: `UNIX, WINDOWS`

Table A-1. Server configuration parameters (page 60 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
SERVICE_LIST_ SOURCE	Source of the available list of services that users can associate with a request (through HP Demand Management) or a project task (through HP Project Management): • If set to `uCMDB`, PPM Center retrieves the list of services for requests from the integrated Universal CMDB application in real time. • If set to `lookup`, PPM Center retrieves the list of services for requests and tasks from the PPM Server.	Default: `Validation` Valid values: `uCMDB, lookup`
SERVICE_LIST_ UCMDB_CACHE_ TIMEOUT	Used for integration with Universal CMDB for service list retrieval. Length of time (in seconds) that the service list remains in PPM Center cache before it is retrieved again.	Default: `300` Valid values: Integer
SERVICE_LIST_ UCMDB_CI_ MAPPINGS	Used for integration with Universal CMDB for service list retrieval. Service list mappings between PPM Center and Universal CMDB CIs. For more information, see the *HP Solution Integrations Guide.*	Default: N/A Example: `name: data_name, description: service_ description`
SERVICE_LIST_ UCMDB_CI_TYPE	Used for integration with Universal CMDB for service list retrieval. Name of the configuration item (CI) type used to store the service list. **Note:** You must create this CI type on the Universal CMDB server. For more information about creating a CI type, see the documentation for Universal CMDB.	Default: `Service` Valid Values:

Table A-1. Server configuration parameters (page 61 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
SERVICE_LIST_ UCMDB_MAX_CI_ NUMBER	Used for integration with Universal CMDB for service list retrieval. Maximum number of Universal CMDB configuration items (CIs) allowed in the service list.	Default: `1000` Valid values: Integer
SERVICE_ PROVIDER_ SECURITY_GROUP	For HP internal use only. Do not change its value unless directed to do so by HP Software Support for PPM Center.	Default: N/A
SERVICES_ ENABLED	Services, if any, run on a node in a server cluster. This parameter is set for every server in a cluster. Valid values are: • `false` - This node does not process light or heavy services • `light` - This node only processes light services • `heavy` - This node only processes heavy services • `true` - This node processes all (light and heavy) service types For a description of PPM Center and instructions on configuring these, see *PPM Center Background Services* on page 232	Default: `true` Valid values: `true, false, light, heavy`
SHOW_BASE_URL_ ON_NOTIFICATIONS	Determines whether the URL for the PPM Center logon window is displayed at the top of each email notification.	Default: `true` Valid values: `true, false`
SHOW_ PERSONALIZE_ FIRST	If set to `true`, **Personalize Dashboard** is the first PPM Dashboard menu item listed on the menu bar.	Default: `false` Valid values: `true, false`

Table A-1. Server configuration parameters (page 62 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
SINGLE_SIGN_ON_ PLUGIN	In single sign-on configuration, this parameter is used to specify the SSO method. You must manually add this parameter to the `server.conf` file. For more information, see Chapter 6, *Implementing User Authentication,* on page 203.	Example `com.kintana.sc.security.auth.SiteMinderSingleSignOn`
SKIP_CHECK_ REQUIRED_FIELD_ WHEN_IMPORT_ PROJECT	If set to `true`, the import utility skips the check for required fields during project creation.	Default: `false` Valid values: `true, false`
SM_RFC_ INTEGRATION_ ENABLED	Enables the integration of PPM Center tasks and Service Manager requests for change (RFCs).	Default: `true` Valid values: `true, false`
SM_PASSWORD	Encrypted password that PPM Center uses to access Service Manager. **Note:** You must use `kEncrypt.sh` to encrypt the password, and then remove the `#!#` character string from the beginning and the end of the encrypted password.	Default: N/A Valid values: Encrypted password
SM_URL	Host name or IP address of Service Manager.	Default: N/A Example: `http://<Host_Name>:13080`
SM_USERNAME	User name that PPM Center uses to access Service Manager. This user name must include only single-byte characters.	Default: N/A Example: **`admin`**

Table A-1. Server configuration parameters (page 63 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
SM_WEB_URL	Address of Service Manager Web tier.	Example: `http://<Host_Name>:<Port>/<WebTier_Package_File_Name>/index.do`
SMTP_ADD_PERIOD	Enables an SMTP client to prepend a dot to each line in the body of an email message.	Default: `true` Valid values: `true, false`
SMTP_PORT	Port used to connect to the SMTP server when sending notifications.	Default: 25 Valid values: Any available port number.
SMTP_RFC_COMPLIANCE	If set to `true,` formats PPM Center email notifications with line-feed <LF> and carriage-return <CR> characters appropriate for restrictive Global 9 security SMTP servers.	Default: `false` Valid values: `true, false`
**SMTP_SERVER Required if notifications are used	Host name of the SMTP-compliant mail server that acts as the gateway for email notifications.	Example `mailserver.mydomain.com`
SMTP_WINDOWS_ADD_PERIOD	If set to `true`, and if a Windows SMTP server is detected, PPM Center appends a period '.' to email notifications.	Default: `true` Valid values: `true, false`
SOCKS_PROXY_HOST	Host name of the SOCKS proxy server.	Host name of the SOCKS proxy server
SOCKS_PROXY_PORT	Port on the SOCKS proxy host that accepts proxy connections.	Any available port on the SOCKS proxy host

Table A-1. Server configuration parameters (page 64 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
*SQLPLUS	Name of the command-line SQL*Plus executable, which must be in the `<Oracle_Home>/bin` directory.	Default: `sqlplus.exe`
SQLPLUS_ CMDLINE_HANDLER	SQL*Plus version and operating system that use the command line to pass arguments.	Default: N/A Example: `101030: WINDOWS`
SQLPLUS_ESCAPE_ CHARACTER	Specifies the SQL*Plus escape character.	Default: (none)
**SQLPLUS_ VERSION	Oracle SQL*Plus version installed on the machine that hosts the PPM Server. You must set this for some PPM Center reports that run from command-line SQL*Plus calls. If you encounter problems running PL/ SQL-based reports in PPM Center, set this parameter.	Example `com.kintana.core.server.SQLPLUS_ VERSION=10.1.0.2`
SYNC_EXEC_INIT_ WAIT_TIME	Amount of time after which the intermediate Request Working page opens. For more information, see the *HP Demand Management User's Guide.*	Default: `4` (seconds)
SYNC_EXEC_MAX_ POLL_TRIES	Maximum number of times to poll for completion of a request before a final message is sent to the user. For more information, see the *HP Demand Management User's Guide.*	Default: `4`
SYNC_EXEC_POLL_ INTERVAL	Time interval (in minutes) at which to poll for completion of a request after the intermediate Request Working page opens. For more information, see the *HP Demand Management User's Guide.*	Default: `15`

Table A-1. Server configuration parameters (page 65 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
SYNCH_PFM_PROJECT_HEALTH_INTERVAL	Determines how often (in seconds) the Synch PFM Project Health service runs.	Default: `600` (seconds) Valid values: N/A
THREAD_POOL_MAX_THREADS	Maximum number of packages to run simultaneously within a release distribution. If a large number of packages in a distribution are processing, increasing this value can improve performance.	Default: `10`
THREAD_POOL_MIN_THREADS	Minimum number of packages to be run simultaneously within a release distribution. See also *THREAD_POOL_MAX_THREADS* on page 457.	Default: `5`

Table A-1. Server configuration parameters (page 66 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
**TIME_ZONE Required if the PPM Server and the Oracle database are in different time zones	Used to set the time zone of the Oracle database. Leave the parameter blank if the PPM Server and the Oracle database host are in the same time zone. If they are in different time zones, set this to the time zone of the Oracle database host. Use a “standard” time zone setting, and not a daylight savings setting (such as EDT or PDT). You can use a fully-qualified time zone name (you are not restricted to three digits), such as “America/Los_Angeles” or “Australia/LHI.” For a list of fully-qualified names, run the Client Time Zone Report in the Admin Tools window of the PPM Workbench. For details on how to run the report, see *Running Server Reports from the Admin Tools Window* on page 263. If you do not a specify a value for this parameter, the value defaults to the time zone in which the PPM Server is running.	Default: Time zone in which the PPM Server is running Valid values: Any fully-qualified time zone designation such as “America/Los_Angeles” or “Australia/LHI.” Do not use daylight savings-modified time zones such as “EDT” or “PDT.”
TM_DISABLE_INCLUDING_LAST_TS_ITEMS	If set to `true`, disables the option for users to include (copy) items from a time sheet for the most recent previous period when they create a new time sheet. If set to `false`, enables the option. HP recommends setting this parameter to `true`.	Default: `true` Valid values: `true`, `false`

Table A-1. Server configuration parameters (page 67 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
TM_DYNAMIC_ DESC_CHARS	If set to `true`, enables the width allotted to the **Item** column on each time sheet to vary as items are added or removed, according to the number of characters in the longest work item data in all the rows on the time sheet. The maximum width of the column is determined by the *TM_DYNAMIC_ DESC_CHARS_MAX* parameter.	Default: `false` Valid values: `true, false`
TM_DYNAMIC_ DESC_CHARS_MAX	If the `TM_DYNAMIC_DESC_CHARS` parameter is set to `true`, this parameter determines the maximum width (in number of characters) of the **Item** column on time sheets. The allotted column width is calculated from this number as an approximation, and the data for particular items might be truncated. If this is an issue, you can increase the value of this parameter. **Note:** The default value of 80 characters is the recommended setting.	Default: 80 Valid values: Integer
TM_MAX_ PREVIOUS_TIME_ PERIODS	Determines the maximum number of time periods that can be displayed in the **Previous Time Periods to Show** list on the Approve Time Sheets page. The default value (4) supplied is for optimal performance. If you increase the value, search performance may suffer.	Default: 4 Valid values: Integer

Table A-1. Server configuration parameters (page 68 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
TM_MAX_ TIMESHEET_ QUERY_RESULTS	Determines the maximum number of result rows retrieved from the database for queries from the Search Time Sheets, Search for a Time Sheet to Freeze, Search for a Time Sheet to Close, and Approve Time Sheet pages.	Default: `500` Valid values: Integer
TMG_ CONFIGURABLE_ FILTERS_REF_ CODE	Stores the reference code for the HP Time Management request type to use to override the default request type used in the Add Work Item to Timesheet window.	Default: N/A Valid values: Reference code of the request type
TMG_FUTURE_ PERIODS_TO_ ALLOW	Specifies the number future periods for which users can specify time on time sheets.	Default: `10`
TMG_PAST_ PERIODS_TO_ ALLOW	Specifies the number of previous periods for which users can specify time.	Default: `10`
TMPM_SYNC_ SERVICE_INTERVAL	Determined the frequency (in milliseconds) with which the TM-PM Sync Service runs. **NOTE:** HP strongly recommends that you keep the default value (equal to three hours).	Default: 10800000 (in ms)
TRANSFER_PATH	Specifies the default temporary directory that PPM Center uses. The main purpose of this directory is to temporarily hold files as they are migrated from a source environment to a destination environment with HP Deployment Management. In a server cluster, all servers must be able to access and share the specified directory.	Example `D:/<PPM_Home>/910/ionia/transfers/`

Table A-1. Server configuration parameters (page 69 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
TURN_ON_ CONCURRENT_ REQUEST_WATCH_ DOG	If set to `true`, enables the Watchdog tool.	Default: `true` Valid values: `true, false`
UCMDB_GATEWAY_ URL	Used for integration with Universal CMDB for CI selection. Web location of the Probe Gateway component of the Discovery and Dependency Mapping (DDM) Probe. The Probe Gateway provides communication (HTTP or HTTPS) between the Probe Manager and the Universal CMDB server for processes such as downloading tasks and returning task results. For more information, see the *HP Solution Integrations Guide.*	Valid values:
UCMDB_MAX_CI_ NUMBER	Used for integration with Universal CMDB for CI selection. Maximum number of configuration items (CIs) to display on the Universal CMDB section of the request details page. For more information, see the *HP Solution Integrations Guide.*	Default: `20` Valid values: Integer between 1 and 100 **Note:** If you specify a value greater than 100, the Universal CMDB server does not restart, and instead displays an error message.

Table A-1. Server configuration parameters (page 70 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
UCMDB_ PASSWORD	Used for integration with Universal CMDB for CI selection. Encrypted password for the Universal CMDB user. For more information, see the *HP Solution Integrations Guide.*	Valid values: Encrypted password

UCMDB_SERVER_ URL	URL of the Universal CMDB server.
	`http://<UCMDB_Host>:<Port>/ucmdb/`
	or
	`https://<UCMDB_Host>:<Port>/ucmdb`
	where `<UCMDB_Host>` represents the machine on which Universal CMDB is running.
	Note: If the Universal CMDB server is configured to support HTTPS, make sure that you configure the `UCMDB_SSL_KEYSTORE_PATH` parameter.
	For more information, see the *HP Solution Integrations Guide.*
		Default: N/A

UCMDB_SSL_ KEYSTORE_PATH	Used for integration with Universal CMDB for CI selection or service list retrieval. SSL keystore path. Required only if *UCMDB_SERVER_URL* parameter uses `HTTPS`.	Default: N/A Example: `/home/release/Instances/ucmdb80.keystore`
UCMDB_USER	Used for integration with Universal CMDB for CI selection. Universal CMDB user name for the integration. For more information, see the *HP Solution Integrations Guide.*	Default: N/A Example: `Admin`

Table A-1. Server configuration parameters (page 71 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
UCMDB_WS_MAX_CONNECTION_NUMBER	Determines the maximum number of connections to the Universal CMDB server through the Web Service API.	Default: 10
UCMDB_WS_PASSWORD	Used for integration with Universal CMDB for service list retrieval. Universal CMDB user password for logging in through Web service. **Note:** This password must be encrypted.	Default: N/A Valid values: Encrypted password
UCMDB_WS_USER	Used for integration with Universal CMDB for service list retrieval. Universal CMDB user name for logging in through Web service.	Default: N/A Example: `Admin`
UI_MONITOR_THRESHOLD	If *ENABLE_UI_MONITOR* on page 413 is set to `true`, this parameter determines the threshold value of the activity monitor.	Default: `4000` (milliseconds) Valid values: Integer greater than 0
USE_HTTPONLY	If set to `true`, enables the HTTPOnly flag for selected cookies used by the PPM Server.	Default: `false` Valid values: `true, false`
USER_PASSWORD_MAX_LENGTH	Maximum number of characters in user passwords.	Default: `16`
USER_PASSWORD_MIN_DIGITS	Minimum number of digits in user passwords.	Default: `0`
USER_PASSWORD_MIN_LENGTH	Minimum number of characters in a user password.	Default: `1`
USER_PASSWORD_MIN_SPECIAL	Determines the minimum number of non-alphanumeric characters that user passwords must contain.	Default: `0`

Table A-1. Server configuration parameters (page 72 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
VALIDATION_LOG_DIR	In a server cluster, if you have overridden the default value for this parameters to refer to a different directory, then all servers in the cluster must be able to access and share the directory.	Same default value as the *BASE_LOG_DIR* parameter Example `D:/<PPM_Home>/910/aeon/logs`
VISUALIZATION_EXEC_TIMEOUT	Length of time (in seconds) that resource management visualizations can run before they time out.	Default: `180`
WF_SCHEDULED_TASK_INTERVAL	Frequency with which the PPM Server checks for pending scheduled tasks, and starts the tasks if worker threads are available.	Default: `60` (seconds)
WF_SCHEDULED_TASK_PRIORITY	Determines the priority of scheduled tasks. Because scheduled tasks run in the background, it may be useful to run them at a lower priority than the threads servicing user-oriented interactive tasks.	Default: `10`

Table A-1. Server configuration parameters (page 73 of 73)

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
WORKAREA_SIZE_POLICY	Controls how the memory for SQL working areas is allocated for intensive operations as sort, group by, hash join, and so on. If this parameter is set to `AUTO` (the default), Oracle manages the allocation and de-allocation of the memory area for each process, and these need not be set separately or manually. Oracle calculates memory allocation based on the load and characteristic of the system. **Note:** HP strongly recommends setting `WORKAREA_SIZE_POLICY` to `AUTO`. This parameter must be set concurrently with *PGA_AGGREGATE_TARGET* on page 441.	GIL: Default: If you are using Oracle 10g, the default is `AUTO`. In Oracle 9i, the default is `AUTO` only if *PGA_AGGREGATE_TARGET* is set.
WORKBENCH_PLUGIN_VERSION	Earliest Java plug-in version used to access the PPM Workbench interface. Use this parameter to use a specific version (other than the default version) of the Java plug-in to open the PPM Workbench.	Example `com.kintana.core.server.WORKBENCH_PLUGIN_VERSION=1.6.0_07`
WS_UPDATE_CLOSED_AND_CANCELED_REQUESTS	If set to `true`, lets Web services update closed and canceled requests.	Default: `false` Valid values: `true, false`

a. For details about this parameter, see the *HP Solution Integrations Guide.*

b. For details about this parameter, see *HP Center Management for Quality Center Guide.*

Server Configuration Parameters Related to the PPM Dashboard

Table A-4 lists and provides descriptions of the PPM Server configuration parameters related to the PPM Dashboard in the PPM Center standard interface. These parameters, like those listed in *Figure A-1*, are located in the `server.conf` file.

The parameter names listed in the table are shortened versions of the actual names, all of which start with the string `com.kintana.core.server.dashboard.` For example, the full name of the `Favorites-Disabled` parameter is `com.kintana.core.server.dashboard.Favorites-Disabled`.

Table A-2. PPM Dashboard-related server configuration parameters (page 1 of 3)

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
Application-Server	Specifies the application server for the PPM Dashboard.	Default: `JBoss 4.0`
BaseURL	PPM Dashboard base URL.	
Character-Encoding	Specifies the coding to use for text displayed in the PPM Dashboard.	Default: `UTF-8`
Chart-Width-Restriction-Enabled	If set to `true`, enables chart width restriction in PPM Dashboard.	Default: `true` Valid values: `true, false`
DSH - Center Name	PPM Dashboard center name.	Default: HP Project and Portfolio Management Center
DSH - Org Units Supported	If set to `true`, enables org units in PPM Dashboard.	Default: `true` Valid values: `true, false`
DSH - Version	Specifies the PPM Dashboard version.	1

Table A-2. PPM Dashboard-related server configuration parameters (page 2 of 3)

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
Dashboard-MLU-Operational	Used with *Supported-MLU- Languages*, determines whether MLU is enabled in the PPM Dashboard.	Default: `false` Valid values: `true, false`
Dashboard-Page-Auto-Refresh-Disabled	Auto-refresh option for the PPM Dashboard.	Default: `false` Valid values: `true, false`
Data-Source	Specifies the data source for the PPM Dashboard.	Default: `java:/ItgDS`
Database-Type	Database used by the PPM Dashboard.	Default: `oracle`
Favorites-Disabled	Used to turn off the PPM Dashboard Favorites option.	Default: `false` Valid values: `true, false`
Fonts-Directory-Path	Directory path of the fonts used by PPM Dashboard.	
Footer	Absolute URL path of the PPM Dashboard footer page.	Default: `/web/knta/dsh/DashboardFooter.jsp`
Header	Absolute URL path of the PPM Dashboard header page.	Default: `/web/knta/dsh/DashboardHeader.jsp`
Layout-Direction	PPM Dashboard layout direction.	
List-Column-Header-Size	Number of characters displayed in the column heading of list portlets. If the number of characters in a column heading exceeds this value, the heading is truncated. The default is 20 characters.	Default: 20 Valid values: Positive integer
Non-SSL-Port	Non-SSL port used by the PPM Dashboard.	Default: N/A

Table A-2. PPM Dashboard-related server configuration parameters (page 3 of 3)

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
openDataSource	Open data source used by the PPM Dashboard.	Default:
PDF-URL	PPM Dashboard URL for PDF files.	Default: N/A
PDF-Unicode-Font-File-Path	Unicode font file path used by PPM Dashboard.	Default: N/A
Page-Auto-Refresh-Disabled	If set to `true`, disables the auto-refresh capability in the PPM Dashboard.	Default: `false` Valid values: `true, false`
Portlet-Thread-Batch-Size	Specifies the batch size of threads used to process prefetch portlets simultaneously.	Default: 4 Valid values: Any positive integer
SQL-Debug	If set to `true`, enables SQL debugging in the PPM Dashboard.	Default: `false` Valid values: `true, false`
Smtp-Port	Specifies an SMTP port for the PPM Dashboard.	Default: N/A
Smtp-Server	Specifies an SMTP server for the PPM Dashboard.	Default: N/A
Supported-MLU-Languages	Specifies the languages supported for the PPM Dashboard. Use commas to separate multiple values.	Default: N/A Valid values: All languages supported by the current PPM Center version
Time-Zone	Specifies the time zone used in the PPM Dashboard.	Valid values:
Use-Full-Screen-Width	Indicates that the PPM Dashboard page view occupies the full width of the screen.	Default: `false` Valid values: `true, false`

Logging Parameters

Table A-3 lists the PPM Server configuration parameters located in the `logging.conf` (logging properties) file, and provides a description of each. The `logging.conf` file is located in the `<PPM_Home>/conf` directory.

Changes to the `logging.conf` file are picked up dynamically by the application (it takes about one minute), so there is no need to restart the application.

The logging parameter names listed in the *Table A-3* are shortened versions of the actual names, all of which start with the string `com.kintana.core.logging.` For example, the full name of the `LOG_LAYOUT` parameter is `com.kintana.core.logging.LOG_LAYOUT.`

Table A-3. Logging parameters (page 1 of 9)

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
CATCH_SYSTEM_ERR	Used to determine whether to redirect `System.err` to the server log.	Default: `true` Valid values: `true, false`
CATCH_SYSTEM_OUT	Used to determine whether to redirect `System.out` to the server log.	Default: `true` Valid values: `true, false`

Table A-3. Logging parameters (page 2 of 9)

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
class.filters	Specifies the class names to include in the stack trace (substring of stack trace classname, including packages). This parameters is used to filter out the classes in stack traces that are of no interest to reduce the log size. If you specify multiple classes or packages, use commas to separate them. If the full class name in a stack trace contains one of the specified classes or package names, then that line is preserved. For example, if the value is set to `com.kintana,com.mercury,` then any class names that contain “com.kintana” or “com.mercury” are kept. The number of traces filtered out is added to server logs after the stack trace.The system default is EMPTY, do not filter out any classes in stack trace.	Default: `None`
CLASS_LOGGING_ LEVEL	Overrides the logging level for a specific Java class to enable logging of a specific piece of code while leaving the rest of the application at a lower logging level. The format of this parameter is the fully-qualified class name followed by a "," then the level. Warning: Use of class filtering degrades performance.	Default: `DEBUG`

Table A-3. Logging parameters (page 3 of 9)

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
CONF_FILE_ RECHECK_INTERVAL	Determines the regular interval (in seconds) at which the `logging.conf` file is checked for changes while the server is running.	Default: `30` (seconds)
context.option	Number that specifies the levels to include context in exception logging, as follows: • 3 includes all • 0 include bitwise combination value in binary (default) • 001 Include context information, if provided • 010 Include stack trace for log messages without exception	Default: `0` Valid values: `3`, `0`, `001`, and `010`

Table A-3. Logging parameters (page 4 of 9)

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
DEFAULT_SERVER_LOGGING_LEVEL	Determines the default debug logging level of the PPM Server. It controls the verbosity of logs generated by the PPM Server. The values, which you can set dynamically at runtime in the Workbench Server Settings window, map as follows: • `ERROR` maps to None in the Server Settings window • `INFO` maps to Normal • `DEBUG` maps to Max For more information about the Server Settings window, see *Setting Debugging and Tracing Parameters* on page 269.	Default: `ERROR` Valid values: • `NONE` - No information, (including errors) is logged • `ERROR` - Only errors are logged • `INFO` - Errors and additional information is logged • `DEBUG` - Includes verbose debugging messages • `ALL` - Displays all log messages generated

Table A-3. Logging parameters (page 5 of 9)

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
DEFAULT_USER_DEBUG_LEVEL	Specifies the default debug level of a user's client session. Controls the verbosity of users' logs on the client, application server, and database. Can be different for different client sessions, and can be changed in the standard interface as a user preference. The values, which can also be set in the Workbench Server Settings window dynamically at runtime, map as follows: • ERROR maps to None in the Server Settings window • INFO maps to Normal • DEBUG maps to Max For more information about the Server Settings window, see *Setting Debugging and Tracing Parameters* on page 269.	Default: `ERROR` Valid values: • `NONE` - No information, (including errors) is logged • `ERROR` - Only errors are logged • `INFO` - Errors and additional information is logged • `DEBUG` - Includes verbose debugging messages • `ALL` - Displays all log messages generated

Table A-3. Logging parameters (page 6 of 9)

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
DEFAULT_USER_LOGGING_LEVEL	Specifies the default logging level for users client sessions. It controls the verbosity of a users logs on a client, application server, and database.	Default: `ERROR` Valid values: `OFF:` No information is logged (even errors are excluded) `ERROR:` Only errors are logged. `INFO:` Errors and additional information are logged. `DEBUG:` Verbose logging of debugging messages. `ALL:` Display all log messages being produced
ENABLE_CONSOLE_LOGGING	If set to true, all logs that are written to the `serverLog.txt` file are also printed to the console that started the PPM Server.	Default: `true` Valid values: `true, false`
ENABLE_EXECUTION_CONSOLE_LOGGING	If set to `true,` during an execution, all logs written to the `serverLog.txt` file are also printed to the console that started the server.	Default: `false` Valid values: `true, false`
ENABLE_WEB_ACCESS_LOGGING	Determines whether or not information sent to the internal PPM Center Web server (Tomcat) is logged.	Default: `false` Valid values: `true, false`

Table A-3. Logging parameters (page 7 of 9)

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
ENCODING	Specifies the default character set for logged messages. This overrides the default operating system character set.	Default: `UTF-8`
FILE_RECHECK_ INTERVAL	Time interval (in seconds) at which the `logging.conf` file is checked for changes. The file keeps being checked as long as the PPM Server is running.	Default: `30`
LOG_LAYOUT	Determines the layout format of the log files.	Default: `XML` Valid values: `TEXT, XML`

Table A-3. Logging parameters (page 8 of 9)

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
LOG_PATTERN	Controls the formatting of the generated logs. You can use the following variables in the log output: • %c - Product function hierarchy of the logged item • %d - Date of logging • %m - Logging message • %n - Line separator • %p - Priority level for the log message • %r - Number of milliseconds elapsed since the application started • %t - Thread name in which the log message created • %x - Username for the user who triggered the log message Because the following log pattern variables can significantly slow performance, HP recommends that you use them only during debugging. • %C - Fully-qualified class name of the class creating the log message • %F - Name of the file in which the log message was created • %l - Class and line number of the log message source • %L - Line number where the log message was created • %M - Method name producing the log message	Default: `%x:%t:%c:%d{yyyy/MM/dd-HH: mm:ss.SSSz}: %m%n` Valid values: See Description, Usage
MAX_BACKUP_INDEX	Limits the number of backup logs kept in the system.	Default: `20`

Table A-3. Logging parameters (page 9 of 9)

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
PRODUCT_ FUNCTION_ LOGGING_LEVEL	Specifies the logging level for a specific product function while leaving the rest of the application at a different logging level. The function areas are hierarchical. Parameter value format: *<Functional_Hierarchy_ Name>, <Logging_Level>*	Default: `com.kintana. crt.request, DEBUG`
ROTATE_LOG_SIZE	As the PPM Server logs information into the `serverLog.txt` file, the file can grow quite large. This parameter determines how large (in KB or MB) it can grow before the server creates a new log file. When the `serverLog.txt` file reaches the size specified by this parameter, the PPM Server renames it (to `serverLog_` *<Timestamp>*`.txt`), and starts a new serverLog.txt file.	Default: `250KB` You can specify the size in either kilobytes or megabytes by appending `KB` or `MB` to the number
SERVER_DEBUG_ LEVEL	Debug level of the PPM Server. Controls the verbosity of logs generated by independent server processes (for example, EmailNotificationAgent). Corresponds to the **Debug Leve**l list in the **Server** section of the Server Settings page.	Valid values: `NONE, LOW, HIGH`
SYSTEM_THRESHOLD	Determines the system-wide logging threshold. IF statements wrapping the log requests check this setting.	Default: `ERROR` Valid values: `ERROR, DEBUG`

LDAP Attribute Parameters

Table A-4 lists and provides descriptions of the PPM Server configuration parameters in the `LdapAttribute.conf` file, which is located in the `<PPM_Home>/conf` directory. Use the `LdapAttribute.conf` file to map the attributes of the LDAP server with the attributes used by the PPM Server.

The default mapping uses the standard LDAP attributes. All values are case-sensitive. Do not add spaces between tokens.

Do not map the `ORG_UNIT_NAME` and `PARENT_ORG_UNIT_NAME` parameters in `LdapAttribute.conf`. These attributes are specified in the `KRSC_ORG_UNITS_INT` table.

Table A-4. LDAP attribute parameters

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
KNTA_USERS_INT	Target table for the import. Can be mapped to any LDAP attribute. Always map both `VISIBLE_USER_DATA` and `USER_DATA`. To disable default mapping: 1. Either comment out or delete the mapping line. Mappings: • USERNAME = sAMAccountName • EMAIL_ADDRESS = mail • PHONE_NUMBER = telephoneNumber • DEPARTMENT_MEANING = departmentNumber • LOCATION_MEANING = locality • MANAGER_USERNAME = manager • USER_DATA1 = mail • VISIBLE_USER_DATA1 = mail 2. Add a placeholder parameter to the `LdapAttribute.conf` file that will add a value to the **FIRST_NAME** and **LAST_NAME** fields.	Format: `ColumnName = LDAPAttribute`

LDAP_TIME_FORMAT	Attribute that keeps track of the time format that the LDAP server uses.	Format for Active Directory servers:
		`yyyyMMddHHmmss'.0Z'`
		Format for Sun Java System Active Server Pages LDAP server:
		`yyyyMMddHHmmss'Z'`

LDAP_USER_OBJECTCLASS	Object class attribute for a user on the LDAP server.	Default: `person`

B Server Directory Structure and Server Tools

Overview of Directory Structure

This appendix addresses the `ppm910` and *`<PPM_Home>`* directories and the scripts and tools they contain. The `ppm910` directory (the installation directory) contains two subdirectories that relate to the Oracle database schemas: `ppm910/sys` and `ppm910/system`.

The *`<PPM_Home>`* directory (the install directory for PPM Center) holds several subdirectories (`bin, docs, logs, reports,` and so on) that contain server and system information, and administrative tools used to perform tasks such as starting, stopping, and reporting on the PPM Server or system.

ppm910/system

The `ppm910/system` directory contains the `CreateKintanaUser.sql` and `CreateRMLUser.sql` scripts.

The `CreateKintanaUser.sql` script variables are:

`<PPM_Username>`	represents the username of the new database schema.
`<PPM_Password>`	represents the password of the new database schema
`<Data_Tablespace>`	represents the tablespace used to store PPM Center tables

<Index_Tablespace>	represents the tablespace used to store PPM Center indexes
<Temp_Tablespace>	represents temporary tablespace
<Clob_Tablespace>	represents the tablespace used to store large data (CLOB).

The `CreateRMLUser.sql` script variables are:

<RML_Username>	represents the username for the new RML database schema.
<RML_Password>	represents the password for the new RML database schema
<RML_Data_Tablespace>	represents the tablespace used to store PPM Center database tables
<RML_Temp_Tablespace>	represents temporary tablespace.

<PPM_Home>/bin

The *<PPM_Home>*`/bin` directory contains all of the scripts required to configure and administer the PPM Server. This section provides descriptions of these scripts.

kBudgetBenefitImport.sh

The `kBudgetBenefitImport.sh` script is used to seed data in the KTMG_PERIODS table.

Run the script as follows:

```
sh ./kBudgetBenefitImport.sh <Num_Periods> <Start_Date>
```

where *<Num_Periods>* represents the number of periods required and *<Start_Date>* represents the start date from which the periods are to be seeded.

The KTMG_PERIOD_TYPES table determines time period duration. The `kBudgetBenefitImport.sh` script presents a list of all the enabled period types that you can use to seed time periods.

If you run `kBudgetBenefitImport.sh` with no arguments, the number of time periods defaults to 24.

kBuildStats.sh

The `kBuildStats.sh` script instructs Oracle to gather statistics about the PPM Center database schema. This information can be very important in improving the overall performance of PPM Center. For information on how to use this script, see *Using the dbms_stats Package to Collect Additional Statistics* on page 292.

kCalculateHealth.sh

The `kCalculateHealth.sh` script computes the classpath for accessing the PPM Center logging libraries.

kCancelStop.sh

The `kCancelStop.sh` script is used to cancel a scheduled shutdown of the running PPM Server. If a command such as `kStop.sh-delay` is being used to stop the server, you can run `kCancelStop.sh` to cancel the stop request. Authentication may be required for this, which works in the same way as for `kStop.sh`. Use the `-user` user name flag.

Example

```
sh ./kCancelStop.sh -user <Admin_Username>
Password: <Admin_Password>
```

kChangeNameDisplay.sh

The `kChangeNameDisplay.sh` script is used to change the display format of PPM Center users' names.

Run the script as follows:

```
sh ./kChangeNameDisplay.sh [-h] <Full_Name_Format>
```

where <*Full_Name_Format*> is 0, 1, or 2.

During the script run, specify the format to use to display a user's full name, as follows:

- To use the format First Last (for example, John Smith), type **`0`**.
- To use the format Last, First (for example, Smith, John), type **`1`**.
- To use the format LastFirst (for example, SmithJohn), type **`2`**.

  The LastFirst format (option 2) is specifically for Korean language users.
- To cancel the operation, type **`a`**.

Running `kChangeNameDisplay.sh` with no arguments defaults the full name format to 0, where first name and last name are separated by one space.

kCharConverter.sh

The **kCharConverter.sh** script is used to convert the character set of a file to a different character set. If no source encoding has been specified. he script uses the default character set of the system It will convert it to the character set specified by destEnc.

Examples

```
sh ./kCharConverter.sh [-p] [-escape <Source_File [Dest_File]>
[Dest_Enc]]
sh ./kCharConverter.sh [-p] [-escape <Source_File [Dest_File]>
[Dest_Enc]]
sh ./kCharConverter.sh [-p] [-escape <Source_File <Dest_File>>
<Source_Enc> <Dest_Enc>]
```

where

`Source_File`	represents the original character file name
`Dest_File`	represents the new file name
`Source_Enc`	represents the original character set encoding for the file
`Dest_Enc`	represents the new character set encoding you are setting for the file

If you do not specify the source encoding, the script uses the default character set of the system and converts that to the destination character set you specify.

kClusterDeploy.sh

If you install PPM Center for the first time as a single-server, stand-alone instance (the default installation), and you later determine that a server cluster better meets your organization's needs, you can convert your system to a cluster that includes additional nodes. The steps required to transition to a clustered configuration include running the `kClusterDeploy.sh` script. For detailed information about how to run this script as you convert PPM Center from a stand-alone instance to a multiple-node cluster, see *Switching from Stand-Alone to a Server Cluster Configuration* on page 200.

kConfig.sh

The `kConfig.sh` script launches the server configuration tool. Because you cannot use `kConfig.sh` to update parameters in a cluster node (that is, anything that comes after an @node), HP recommends that, for a server cluster environment, you use a text editor to edit (or add) parameter values directly in the `server.conf` file, or use the Administration Console to modify parameter values. Regardless of how you modify configuration parameters, you must run the `kUpdateHtml.sh` script (see *kUpdateHtml.sh* on page 505) afterward to implement your changes.

On a Windows system, run the script as follows:

- To run the script in console mode:

  ```
  ash ./kConfig.sh -console
  ```

- To run the script in graphic mode:

  ```
  ash ./kConfig.sh -swing
  ```

  On a UNIX system, run the script as follows:

- To run the script in console mode:

  ```
  sh ./kConfig.sh -console
  ```

- To run the script in graphic mode:

  ```
  sh ./kConfig.sh -swing
  ```

kConvertProject.sh

The `kConvertProject.sh` script converts project effort data from days to hours.

kConvertToLog4j.sh

The `kConvertToLog4j.sh` script converts the JDBC log, Web log, or server log to the log4j XML format. You can view logs in this format with a tool such as Chainsaw (a GUI-based log viewer available at the Web site logging.apache.org/log4j/docs/chainsaw.html).

Examples

To convert a Web log to the log4j XML format.

```
sh ./kConvertToLog4j.sh -webLog apacheLog.txt
```

To convert a JDBC log to the log4j XML format.

```
sh ./kConvertToLog4j.sh -jdbcLog jdbc.kintana.log
```

To convert a `serverLog.txt` file in text format to the log4j XML format.

```
sh ./kConvertToLog4j.sh -serverLog serverLog.txt
```

To convert a server log, JDBC log, and Web log, and then concatenate them in a result log.

```
sh ./kConvertToLog4j.sh -serverLog serverLog.txt -jdbcLog
jdbc.kintana.log -webLogiisLog.txt
```

For information about usage type.

```
sh ./kConvertToLog4j.sh -help
```

where

<Server_Log>	represents the server log to be converted
<JDBC_Log>	represents the jdbc log to be converted
<Web_Log>	represents the Web log to be converted
<Merged_Log>	represents the name of the result log

<Web_Date_Format>	represents the format of date for the Web log Example `DD/MMM/YYYY:HH:MM:SS`
<Web_Date_Regex>	represents a regular expression to match the date in the Web log
<Srv_Date_Format>	represents the date format for the server log Example `DD/MMM/YYYY:HH:MM:SS`
<Srv_Date_Regex>	represents a regular expression to match the date in the server log
<Append>	Determines whether the existing merged log is appended. The default is `No`.
<Help>	represents a message to display

kConvertUserPasswords.sh

The `kConvertUserPasswords.sh` script is used to convert the user password storage algorithm between a one-way hash and a reversible encryption scheme. Converting to the hashing algorithm ensures the security of your saved user passwords, but disables the [USR.PASSWORD] token in any commands, notifications, and so on where it is used. The standard encryption option (the default) saves passwords securely encrypted with the El Gamal public/private key algorithm, which enables the server to decrypt the passwords for uses such as the [USR.PASSWORD] token.

kDeploy.sh

The `kDeploy.sh` script is used to install HP Deployment Management Extensions, PPM Center Best Practices, language packs, and PPM Center product service packs. This software is distributed as a deployment (a software bundle that contains files) in the following format.

`ppm-<Ver>-<ID>[.#'].jar`

where

<Ver>	represents the PPM Center version for which you can install the Extension, Best Practices, or service pack
<ID>	represents the unique identifier for service pack
(Optional) `.#'`	represents the revision number for the deployment.

Example

To install a product service pack SP1:

1. Make sure that the deployment JAR file is in the *<PPM_Home>* directory.

   There is no need to extract anything. The `kdeploy.sh` script does that for you.

2. To apply the SP1 service pack, run the command:

   ```
   sh ./kDeploy.sh -i SP1
   ```

   Table B-1 displays the key command-line options for `kDeploy.sh`. To generate a list of options, run the command `sh ./kDeploy.sh -h`.

Table B-1. Key command-line options for kDeploy.sh

Option	Description
-i	Installs deployments. Example To install a PPM Center service pack (SP) 14, run the command: **`sh ./kDeploy.sh -i SP14`**
-l	Lists the deployments installed on an instance.
-D	Searches for bundles in a given directory. Example To search for a file in the `DIR` directory, run the command: **`sh ./kDeploy.sh -D DIR`**
-h	Provides help for `kDeploy.sh.` Lists all the command-line options.
-f	Reinstalls an existing deployment.
-lang	Installs a language pack after PPM Center installation or upgrade. For information about how to install a language pack, see the *Multilingual User Interface Guide.*
-k	Includes the PPM Center database schema password in the command. Automates command execution but may be a security risk.
-u	Includes the PPM Center user name in the command.
-p	Includes the password for the PPM Center user name in the command. Automates command execution but may be a security risk.
-tidy	Cleans up unnecessary deployment files.
-skip -database	Specifies that database changes are not to be applied if they already exist.
-update-deploy	Extracts the new `kDeploy.sh,` if a new version exists.

kDevMigratorExtract.sh

The `kDevMigratorExtract.sh` script uses the content migrator to extract a content bundle from the PPM Center instance.

You must run the `kDevMigratorExtract.sh` script from the `<PPM_Home>/bin` directory.

You can use the following command-line options with the script:

Option	Description
-username	PPM Center administrator user name
-password	PPM Center administrator user password
-dbpassword	System database password
-itghome	Directory where PPM Center is installed (`<PPM_Home>`)
-action	Action to perform. Specify either `search` or `bundle`.
-entityId	Entity ID
-keyword	Search keywords
-filename	Content bundle filename
-delimiter	Delimiter string
-help	Help text
-quiet	Suppresses output from the `kVariables.sh` script.

In the following example, the script searches for validations on the PPM Server.

```
sh ./kDevMigratorExtract.sh -username admin -password pwd
-action search -entityId 13 -keyword “Reference”
```

kDevMigratorImport.sh

The `kDevMigratorImport.sh` script uses the content migrator to import a content bundle into the PPM Center instance.

kEnableTimeMgmtPeriodType.sh

Use the `kEnableTimeMgmtPeriodType.sh` script to enable or disable period types in the KTMG_PERIOD_TYPES table. During the script run, you are presented with a list of all of the available period types. To disable an enabled period type, select it from the list. To enable a disabled period type, select it from the list.

kEncrypt.sh

In some cases you may need to generate encrypted strings in accordance with the encryption scheme of your PPM Server installation. To do this, you use the `kEncrypt.sh.` script.

Run the command:

```
sh ./kEncrypt.sh <String_To_Encrypt>
```

The `kEncrypt.sh` script run generates an encrypted string that starts and ends with the characters `#!#,` which the system uses to mark encrypted data. Copy only the text string between these markers.

kExportAttributes.sh

You can use the `kExportAttributes.sh` script to export the translatable attributes of entities for a PPM Center system that supports multiple languages. If you export a specific entity, the attributes of child entities defined in the same language are included.

To export the all attribute definitions, run the command:

```
sh ./kExportAttributes.sh -username <username> -password
<password>
```

To export the translations for all entities, run the command:

```
sh ./kExportAttributes.sh -username <username> -password
<password> -t
```

To export the translations for a single entity, run the command:

```
sh ./kExportAttributes.sh -username <username> -password
<password> -entityId <entityId> -t
```

For detailed information about how to use the `kExportAttributes.sh` script (and the `kImportAttributes.sh` script), see the *Multilingual User Interface Guide*.

kGenFiscalPeriods.sh

You can use the `kGenFiscalPeriods.sh` script to:

- Generate sets of periods to be used by HP Portfolio Management, HP Financial Management, and HP Resource Management, for a range of years you specify
- Customize the formats of periods (years, quarters, and months) as they appear in financial summaries and elsewhere
- Customize the format of weeks as they appear in the project cumulative cost page in HP Financial Management
- Customize the names of months as they appear in financial summaries and elsewhere

- Change the month in which fiscal years start, to match your fiscal calendar
- Support fiscal years with more than twelve periods (not twelve months) to represent standard and non-standard retail calendars in financial summaries, scenario comparisons, some portlets, and some reports (but not in time sheets)
- Change the start day of the week, used in HP Financial Management

For more detailed information and instructions on how to set up these functions, see the *Generating Fiscal Periods* document.

kGenTimeMgmtPeriods.sh

The `kGenTimeMgmtPeriods.sh` script is used in HP Time Management to populate the `KTMG_PERIODS` table with data. The script takes the number of periods to be populated and the start date from which the periods are to be populated.

Run the command:

```
sh ./kGenTimeMgmtPeriods.sh <Num> <Start_Date>
```

where

<Num>	represents the number of time periods required
<Start_Date>	represents the date from which the periods are to be populated.

For a new installation, running `kGenTimeMgmtPeriods.sh` is optional. If you run the script with no arguments, the number of time periods defaults to 24.

kHash.sh

The user name and password required to access the JMX console are encrypted to prevent unauthorized access to the information that the JMX console makes available. They are both stored as SHA-1 hash output in the `jmx-console-users.properties` file, which is located in the `<PPM_Home>/conf/props` directory.

You can run the `kHash.sh` script to output the hashed password required to access the JBoss JMX console, as follows:

```
sh ./kHash.sh -t <Password_Text>
```

kImportAttributes.sh

Use the `kImportAttributes.sh` script to import the translatable attributes of PPM Center entities that were exported using the `kExportAttributes.sh` script.

To import all of the files in the `<PPM_Home>/mlu/translations/` directory, run the command:

```
sh ./kImportAttributes.sh -username <user_name> -password
<password>
```

For detailed information about how to use the `kImportAttributes.sh` script, see the *Multilingual User Interface Guide*.

kJSPCompiler.sh

The first time a user requests a page in the PPM Center standard interface, the server must compile the page. To eliminate this initial performance drag, run the `kJSPCompiler.sh` script to precompile all of the JSP pages before users request them. This gives first-time users faster access to the standard PPM Center interface.

kKeygen.sh

The `kKeygen.sh` script generates new security keys.

kLdap.sh

This script is used to add the `kntaUser` attribute to specified entries in the LDAP schema. You can specify the entries using the standard LDAP search filter. If you do not specify a filter, the attribute is added to all the entries, starting from the base DN.

Example

```
sh ./kLdap.sh -s
```

where -s indicates that the LDAP server parameters are to be read from the `server.conf` file.

kLicenseReader.sh

Use the `kLicenseReader.sh` to run the license reader tool The license reader reads the encrypted license file and provides the following information:

- PPM Center products are licensed for use on your instance
- IP address of the licenses machine
- Expiration dates for licenses
- Number of licenses available for different PPM Center modules.

You can only use the script to read the license information. You cannot use the script to modify the license information.

Run the following command as follows:

```
kLicenseReader.sh [-filename <License_File_Name>]
[-filepath <License_File_Path>] [-help]
```

If you do not specify the file name, the script uses `license.conf` by default. If you do not specify the file path, the script uses `<PPM_Home>/conf` as the default path.

kMigratorExtract.sh

The script `kMigratorExtract.sh` is used in PPM Center entity migration.

It requires the following parameters:

-username *<Username>*

-action *<Search>*, *<Bundle>*, *<Test>*

-referenceCode *<Reference_Code>*

-entityId *<Entity_Id>*

13 Validation

11 Special Command

26 Object Type

17 Report Type

9 Workflow

4 Environment

58 Environment Group

39 Request Header Type

19 Request Type

522 Workplan Template

61 Overview Page Section

37 User Data Context

509 Portlet Definition

470 Module

505 Data Source

521 Project Type

The following parameters are optional:

-url *<URL>*

-password *<Password>*

-delimiter *<Delimiter>*

-quiet

-keyword *<Keyword>*

-primaryKey *<Primary_Key>*

-primaryKeyName *<Primary_Key_Name>*

-filename *<File_Name>*

-uncompressed

kMigratorImport.sh

Use the `kMigratorImport.sh` script to migrate PPM Center entities. Type only **Y** or **N** for the 19 flags listed.

Example

To import a file, run the command:

```
sh ./kMigratorImport.sh -username <Username> -password
<Password> -action import -filename <'Full_File_Path'> -i18n
none -refdata nochange -flags NNNNNNNNNNYYNNNNN
```

Make sure that the full file path is enclosed in single quotes.

The following parameters are required for this script:

```
-username <username>
```

```
-action <import, trial>
```

```
-filename <filename>
```

```
-i18n <none, charset, locale>
```

none: Require same language and character set

charset: Ignore language and character set warnings

locale: Ignore all warnings

-refdata <nochange, install, update>

nochange: Do not change reference data

install: Install reference data

update: Create new or replace existing reference data

-flags <flags>

Flag 1: Replace existing Object Type

Flag 2: Replace existing Request Type

Flag 3: Replace existing Request Header Type

Flag 4: Replace existing Special Command

Flag 5: Replace existing Validation

Flag 6: Replace existing Workflow

Flag 7: Replace existing Report Type

Flag 8: Replace existing Workplan Template

Flag 9: Replace existing Workflow Step Sources

Flag 10: Add missing Environment

Flag 11: Add missing Security Group

Flag 12: Add missing Request Status

Flag 13: Replace existing Overview Page Section

Flag 14: Replace existing User Data Context

Flag 15: Replace existing Portlet Definition

Flag 16: Replace existing Module

Flag 17: Replace existing Data Source

Flag 18: Replace existing Project Type

Flag 19: Replace existing Sub workflow

The following parameters are optional:

- `-url <URL>`
- `-password <Password>`
- `-report <Report>`

kPMTMSync.sh

Use the `kPMTMSync.sh` script to run a synchronization script that copies actuals with matching tasks and resources from HP Time Management to HP Project Management. Any actuals not entered into HP Project Management using HP Time Management are replaced by actuals from HP Time Management.

Run the script as follows:

```
sh ./kPMTMSync.sh [-projectno <Project_Number> | -projectname
<Project_Name>]
[-username <User_Name>]
[-password <Password>]
```

To run this script, you must have the Edit All Projects access grant. For complete details about what the synchronization script does and how to run it, see the *HP Project Management Configuration Guide*.

kRunCacheManager.sh

Use the `kRunCacheManager.sh` script to manage your cache from the command line and without having to restart the PPM Server. You can specify a report number and a url for the server manager or for a server. If you do not specify the url, the command parses the `server.conf` file for the server manager url.

Run the script as follows:

```
sh ./kRunCacheManager.sh
```

You can script this to run after your database changes have been committed.

kRunServerAdminReport.sh

You can use the `kRunServerAdminReport.sh` script to run diagnostic reports on the PPM Server. This utility provides a summary of current activity on the system and the number of database connections made.

You can also access this functionality through the PPM Workbench. To access and run these diagnostic reports from the PPM Workbench, on the shortcut bar, select **Sys Admin > Server Tools.**

The reports listed in the Admin Tools window are the same reports you can use the `kRunServerAdminReport.sh` script to run.

kStandaloneDeploy.sh

If you install PPM Center for the first time as a server cluster deployment, and you later determine that a stand-alone instance better meets your organization's needs, you can convert your system to a stand-alone configuration. The steps required to transition to a stand-alone configuration include running the `kStandaloneDeploy.sh` script. For detailed information about how to run this script, see *Switching from Stand-Alone to a Server Cluster Configuration* on page 200.

kStart.sh

The `kStart.sh` script is used only on UNIX systems to start the PPM Server as a background process. For more details about starting the server, see *Starting and Stopping the PPM Server* on page 95.

For PPM Servers participating in the same cluster, HP recommends the following:

- Start the servers one at a time.
- Start each server with an explicit partition name to avoid inadvertent cluster participation.

For server clustering, the `kStart.sh` script accepts `partition` as a parameter, as follows:

```
sh ./kStart.sh [-name <PPM_Server_Name>][-partition <Partition_
Name>]
```

If no value is specified, the default partition name `ppm_p` is used.

kStatus.sh

Run the `kStatus.sh` script to check the state of the PPM Server. This script returns the server status whether the server is running or not. If it is running, the script returns the current load value, which refers to the number of active user sessions.

kStop.sh

Use the `kStop.sh` script to stop the PPM Server. This script requires some arguments. You can use the `-now` flag to quickly stop the server, or use the `-delay <#minutes>` flag to stop it after a delay of a specified number of minutes.

If you are using the `-delay` option, you can use the `kCancelStop.sh` script to cancel the stop request.

Using the `-delay` option automatically issues a message to advise all connected PPM Center users that the server will stop after the specified delay. This script requires authentication if the server parameter `REMOTE_ADMIN_`

`REQUIRE_AUTH` is set to `true`. In this case, you must also specify the flag `-user <Username>`.

For more information on available flags, run `kStop.sh` without any options. For information about how to stop the server, see *Starting and Stopping the PPM Server* on page 95.

kSupport.sh

Use the `kSupport.sh` script to gather information useful to HP Software Support in diagnosing system problems, and create a Zip file with a timestamp in the `support/zipfiles` directory.

The `kSupport.sh` script gathers information from the following:

- Install logs
- Server logs (with the option for a date range)
- JDBC logs
- Deploy logs (for the installation of patches and HP Deployment Management Extensions)
- Configuration files
- Server reports
- Database information
- File system information

As it collects server logs or JDBC logs, the script concatenates all the files into one server `Log.txt` file.

You can run `kSupport.sh` in GUI, console, or silent mode. Silent mode automatically captures a default set of information without prompting for user input.

To run in GUI mode:

```
sh ./kSupport.sh
```

To run in console mode:

```
sh ./kSupport.sh -console
```

To run in silent mode:

```
sh ./kSupport.sh -silent -k <Password> -customer <Company_Name>
-sr <Service_Request_Number>
```

kTestSiteMinder.sh

Use the `kTestSiteMinder.sh` script to determine whether SiteMinder is correctly configured with your PPM Center instance and can authenticate your account. If your `siteminder.conf` file is not configured correctly, then the file is not parsed, and the error “Unable to connect to SiteMinder Agent” is displayed. If you provide an invalid username or password, the message “User is failed for the authentication: 2” is displayed.

After a `kTestSiteMinder.sh` script run, the SiteMinder Agent API displays one of the following values:

```
NOCONNECTION = -3;
TIMEOUT = -2;
FAILURE = -1;
SUCCESS = 0;
YES = 1;
NO = 2;
CHALLENGE = 3;
UNRESOLVED = 4;
```

kTMDataConversion.sh

Use this time sheet data loader script to import large volumes of time sheet data from an external application into the PPM Center database. You can run the script either before or after you start to use HP Time Management. The time sheets that the script creates from data you specify in `.xml` files have the functionality as the time sheets created using HP Time Management in PPM Center.

To run this script, you must have the Edit Time Sheets access grant and the Edit All Projects access grant. For complete details about the script, the

requirements for running it, and how to run it, see the *Tracking and Managing IT Demand Configuration Guide*.

kUpdateHtml.sh

The `kUpdateHtml.sh` script is a key script used to update the PPM Server configuration. Always run the `kUpdateHtml.sh` script after you make changes to the `server.conf` file from a text editor or by running the `kConfig.sh` script (see *kConfig.sh* on page 486). If you change the configuration through the Administration Console, run the `kUpdateHtml.sh` script afterward *only* if the Administration Console prompts you to restart the PPM Server.

kVariables.sh

The `kVariables.sh` script automatically generates the JAVA class search path, validates the JAVA_HOME environment variable, and sets the KNTA_HOME environment variable. Many scripts, including `kStart.sh`, `kStatus.sh`, and `kStop.sh`, call the `kVariables.sh` script to set the JAVA classpath.

kWall.sh

Use the `kWall.sh` script to send a message to all users logged on to the PPM Workbench. When you run the script, it prompts you for your PPM Center user name and password, for the message text, and for a list of recipients.

setServerMode.sh

The `setServerMode.sh` script, located in the `<PPM_Home>/bin` directory, sets the server mode in case you want exclusive access to a running server.

The following are valid server mode values:

- **Normal.** In normal mode, all enabled users can log on, and all services are available, subject to restrictions set in `server.conf` parameters.
- **Restricted.** In restricted mode, the server lets users with Administrator access grant log on. The server cannot run scheduled executions, notifications, or the concurrent request manager while in this mode.

  Before you can install an HP Deployment Management Extension, you must set the server to restricted mode.
- **Disabled.** Disabled mode prevents server startup. A server enters disabled mode only after a PPM Center upgrade exits before the upgrade is completed.

To set the server mode using the `setServerMode.sh` script:

1. On the desktop, select **Start > Run.**

   The Run dialog box opens.

2. In the **Open** field, type:

   ```
   sh ./setServerMode.sh <Mode_Value>
   ```

For more information about server modes, see *Setting the Server Mode* on page 95.

<PPM_Home>/pdf

The `pdf` subdirectory contains all documentation files for PPM Center (to view them, you need Adobe Reader).

You can also access product documentation:

- From **Product Information > Documentation** in either the PPM Center standard interface or the PPM Workbench interface
- The HP Software Product Manuals Web site (h20230.www2.hp.com/selfsolve/manuals)

<PPM_Home>/integration

The `integration` subdirectory contains information or examples for various common integrations between the PPM Server and external systems. For example, the *<PPM_Home>*`/integration/webserver` directory contains information about each external Web server that you can integrate with the PPM Server. Files used to perform the integration are located in these folders. For more information on using the folders and files in the `integration` subdirectory, see the relevant document that pertains to the integration involved.

<PPM_Home>/logs

The server directory structure has two log directories. The *<PPM_Home>*/logs directory contains the reports subdirectory, which contains a log file for each PPM Server report that is run, and directories named PKG_number and REQ_number. These subdirectories contain execution logs for HP Deployment Management packages and HP Demand Management requests. The *<Number>* placeholder in the directory name corresponds to the ID of the package or request being run.

The other log directory, *<PPM_Home>*/server/*<PPM_Server_Name>*/log contains all PPM Server-generated logs. As the server runs, it generates logging messages and writes them to the serverLog.txt file. When this file reaches the size indicated by the ROTATE_LOG_SIZE server parameter, it is renamed to serverLog_timestamp.txt, and a new serverLog.txt is started.

The Java servlets used to serve the Web pages generate their own log files, named servletLog.txt. The amount of information in the server log files depends on the debugging level set in the server configuration. The server parameters SERVER_DEBUG_LEVEL and DEFAULT_USER_DEBUG_LEVEL control the debugging level. If a problem arises and you require more information in the logs, log on to the PPM Workbench as Administrator and reset the server debug level to Maximum debugging information (select **Edit > Debug Settings**).

<PPM_Home>/reports

The reports subdirectory contains the HTML files for all reports that PPM Center clients have run.

<*PPM_Home*>/server

The `<PPM_Home>/server` directory contains the deployed PPM Server. Typically, administrators are not required to make any changes in this directory. Server configurations are handled through the provided admin scripts in the `<PPM_Home>/bin` directory.

<*PPM_Home*>/sql

The `sql` subdirectory contains source code for the built-in PPM Center reports and core PL/SQL packages. This is provided for convenience and for customization needs.

<*PPM_Home*>/transfers

The `transfers` subdirectory serves as temporary storage for files transferred between the server and remote computers. For more information about how the transfers directory is used in entity migration, see *Basic Parameters* on page 336.

<*PPM_Home*>/utilities

The `utilities` subdirectory contains scripts for granting the SYS-level privileges to the PPM Center database schema, and for creating the PPM Center users required for installing or upgrading PPM Center. It also contains diagnostic tools for troubleshooting PPM Center installation, and the Watchdog tool.

kWatchdog.sh

This script is used to run the Watchdog tool. Watchdog is a stand-alone tool that issues a command to generate a thread dump whenever memory exceeds the configured threshold after a full garbage collection (GC). This tool requires that the Java garbage collection log be turned on at startup.

Run the command:

```
sh ./kWatchdog.sh pid
```

For more detailed information about the watchdog tool and the requirements for using it, see *Using the Watchdog Tool* on page 321.

Other Directories

Directories other than those described in this appendix contain reference files, as indicated by their names. You are not likely to require access to these directories.

Index

Symbols

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W