Project and Portfolio Management Center

Software Version: 9.40

Installation and Administration Guide

Document Release Date: September 2016
Software Release Date: September 2016

Hewlett Packard Enterprise

Legal Notices

Warranty

The only warranties for Hewlett Packard Enterprise 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. Hewlett Packard Enterprise 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 Hewlett Packard Enterprise 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 Notice

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

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

UNIX® is a registered trademark of The Open Group.

Documentation Updates

To check for recent updates or to verify that you are using the most recent edition of a document, go to: https://softwaresupport.hpe.com/.

This site requires that you register for an HP Passport and to sign in. To register for an HP Passport ID, click **Register** on the HPE Software Support site or click **Create an Account** on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HPE sales representative for details.

The following table indicates changes made to this document since the last released edition.

Support

Visit the HPE Software Support site at: https://softwaresupport.hpe.com/.

Most of the support areas require that you register as an HP Passport user and to sign in. Many also require a support contract. To register for an HP Passport ID, click **Register** on the HPE Support site or click **Create an Account** on the HP Passport login page.

To find more information about access levels, go to: https://softwaresupport.hpe.com/web/softwaresupport/access-levels.

HPE Software Solutions Now accesses the HPSW Solution and Integration Portal website. This site enables you to explore HPE Product Solutions to meet your business needs, includes a full list of Integrations between HPE Products, as well as a listing of ITIL Processes. The URL for this website is https://softwaresupport.hpe.com/km/KM01702731.

Contents

Chapter 1: Administering Project and Portfolio Management Center

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

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

- Overview of PPM system architecture and of single-server and server cluster system configuration ("System Overview" on page 18)
- Information about product licensing and optional programs that you can install ("Installing PPM" on page 33)
- Instructions on how to create the required database schemas, verify installation, and install service packs and Deployment Management Extensions ("Installing PPM" on page 33)
- Details on how to configure all components of the PPM system and to start and stop the PPM Server ("Configuring the System" on page 76)
- Information that PPM users need to know in order to use the PPM Workbench ("Configuring the System" on page 76)
- Advanced configuration information, including details on how to configure an external Web server and PPM Server clusters ("Advanced System Configuration" on page 126)
- Information on how to integrate PPM with an LDAP server ("Advanced System Configuration" on page 126)
- Information about the kinds of performance issues that can arise, and how to identify and resolve them ("Improving System Performance" on page 223)
- Information on how to migrate entire instances of PPM, and on how to migrate just the database schemas ("Migrating Instances" on page 352)
- Details on how to maintain the PPM and the database after installation and configuration ("Maintaining the System" on page 264)
- Details on how to use the HPE entity migrators to migrate specific kinds of PPM entities and associated objects between instances of PPM ("Migrating Entities" on page 364)

- PPM Server configuration parameters ("PPM Configuration Parameters" on page 401)
- Details about PPM directories and the scripts and tools they contain ("Server Directory Structure and Server Tools" on page 490)

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, 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*

  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.

- Upgrade Guide

  If you plan to upgrade from an earlier version of PPM, 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 include:

- *Deployment Best Practices for PPM Operational Reporting*

- *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*
- *HPE-Supplied Entities Guide* (includes descriptions of all portlets, request types, and workflows in PPM)

Accessing PPM Documentation

To access the online help system of the PPM documentation, from the PPM menu bar, select **Help > Help Center**.

The PDF version of PPM documentation are also included with the product. This enables end users to access the PDF version of PPM documentation easily, especially for users who have no access to the HPE Manuals Site. To access PDF version of PPM documentation included with the product, from the **Help** menu, click **Help Center**, then in the **Get Started** section, click **Navigate PPM Documentation**. Locate a document of your interest and click the **PDF** link for the document.

> **Note:** Most of documents in the PPM documentation set are available in both online help and PDF format. Only some less used documents are still available in PDF format only.

> **Tip:** For PPM administrators who would like to access all PDF files, go to the `<PPM_Home>/itg/pdf/manual/Content/PDFs` directory.

Chapter 2: System Overview

This section provides an overview of PPM architecture.

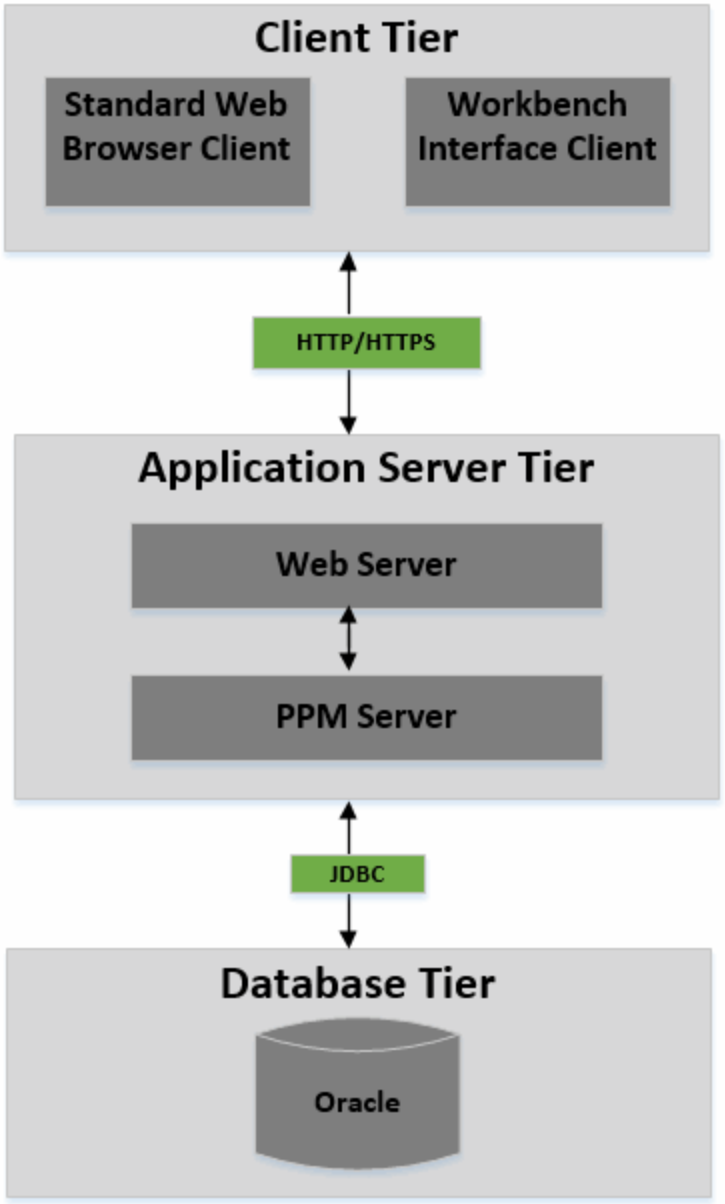
Overview of PPM Architecture

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

- Client browsers ("Client Tier " on page 20)
- One or more middle-tier J2EE servers ("Application Server Tier" on page 20)
- A single Oracle relational database ("Database Tier" on page 21)

This arrangement is shown in the following figure.

PPM architecture



Browser clients use HTTP or HTTPS (HTTPS requires an external Web server) to communicate with the PPM 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 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 Oracle Java plug-in to a web browser.
- The Administration Console. The Administration Console is embedded in the PPM standard interface.

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.

  > **Note:** 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 Remote Method Invocation over SSL (RMIS), which is optimized for use in PPM.

  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 standard and PPM Workbench interfaces, see the *Getting Started* guide.

Application Server Tier

The application server (PPM Server):

- Runs on the Microsoft® Windows®, Oracle Solaris, HPE-UX, IBM AIX, Red Hat Linux, and SUSE Linux platforms
- Uses the Tomcat Application Server
- Houses workflow, scheduling, notification, and execution engines that drive automated tasks such as 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 Oracle Java System Web Server, Microsoft IIS, Apache HTTP Server, Apache-based Web Server (from HPE), and IBM HTTP Server

  > **Note:** For detailed information on which Web server versions PPM 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 126.

Database Tier

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

> **Note:** 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.

> **Note:** Running PPM on Oracle RAC is supported. However, the Oracle RAC failover feature is not available due to that PPM uses Oracle Thin JDBC Driver.

PPM supports the following Oracle database features:

- A relational data model
- Use of Oracle stored procedures to implement business logic (for example, workflow processing)

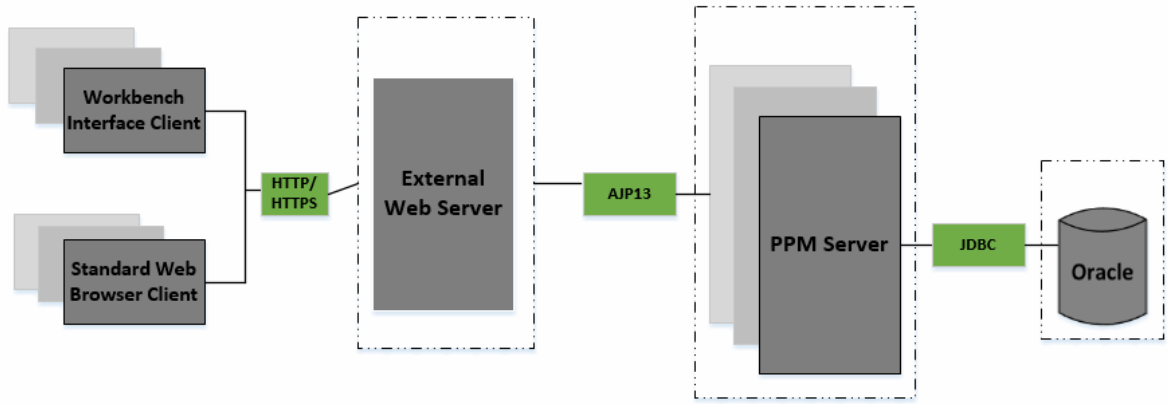
- Use of a database connection pool to eliminate the need to create a separate database session for each user or transaction
- Database caching of frequently-used data, programs, and procedures to improve performance
- A database schema (separate from the PPM database schema) for implementing operation reporting in PPM

System Configurations

The three-tier architecture of PPM 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 instance.

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 RMIS 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.

Server Cluster Configurations (Recommended)

PPM Servers are deployed in a server cluster configuration. Clustering enables you to run PPM on several parallel servers, or *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 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 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.

> **Caution:** To avoid problems with memory and performance, HPE strongly recommends that you isolate background services from user traffic. For more information, see "Services Isolation" on page 27.

In a server cluster configuration, you can enable multiple nodes to run background services.

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 system services remain operational.

> **Note:** 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 after a node shuts down see only changes that were saved on that node.

A PPM server cluster contains one or more nodes and an Oracle database. The first node 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.

> **Note:** A server cluster configuration 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 application code, only a single copy is required for each machine, regardless of the number of servers running on that machine.

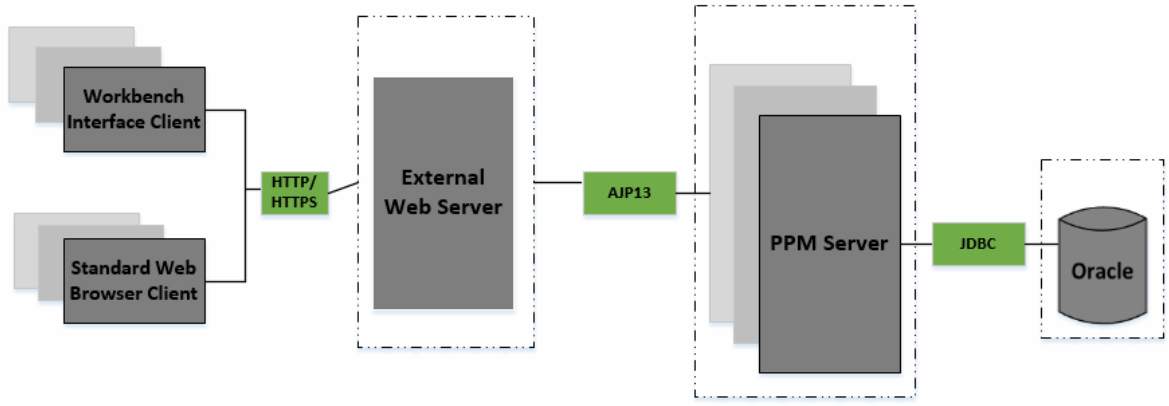
You can set up server clusters with an external Web server, or with a hardware load balancer. The following sections describe these two setups.

Server Cluster/External Web Server Configuration

The server cluster/external Web server configuration 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.

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 RMIS 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.

The nodes also accept TCP/UDP connections from other nodes for cache synchronization and cluster moniter.

Software Load Balancing

You can use the PPM Web server module as the software load balancer for a PPM Server cluster configuration. For this configuration, HPE recommends that PPM 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 HPE PPM Web server module.
3. The HPE 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, see "Implementing User

Authentication" on page 188.

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.

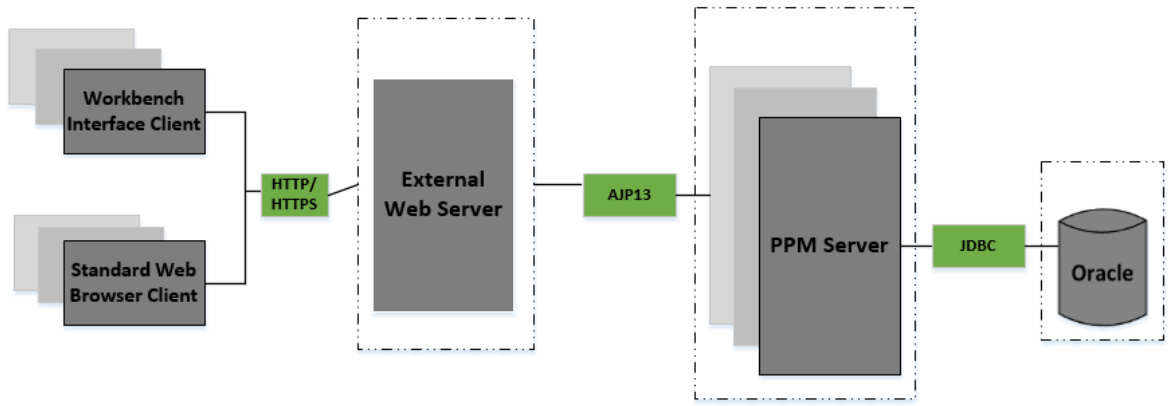
The external Web server and PPM Servers communicate using AJP13, a proprietary protocol that can be more efficient than HTTP for communicating with PPM Servers using an external Web server. For information about how to set up a server cluster with an external Web server, see "Advanced System Configuration" on page 126.

Server Cluster Hardware Load Balancer Configuration

The server cluster/hardware load balancer configuration 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.

> **Note:** HPE strongly advises to use cookie-based session persistence for load balancers. IP-based session persistence is known to result in issues when proxies or VPN are used by end users, and should be discouraged.

Figure. Server cluster/hardware load balancer configuration



In the server cluster/hardware load balancer configuration:

- 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.

Many hardware load balancers support handling HTTPS and forwarding plain HTTP. If you use hardware load balancer to forward HTTPS to PPM Server, you must also have an external Web server. 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, RMIS.
- The PPM Server and Oracle database reside on separate machines and communicate with each other using JDBC.

> **Note:** Although the above figure 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 "Advanced System Configuration" on page 126.

Services Isolation

PPM has many asynchronous background services that process data "behind the scenes" while the application is running. Depending on data characteristics of your PPM 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, HPE strongly recommends that you isolate the services on a separate JVM within the PPM 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 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.

> **Note:** 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.

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

Dedicating one PPM node to your services enables you to:

- Minimize the effect that running PPM 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 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.

When you install PPM; (see "Installing PPM" on page 33), you specify whether you want a server cluster configuration (required for a production instance) or a *stand-alone* configuration. A stand-alone PPM 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.

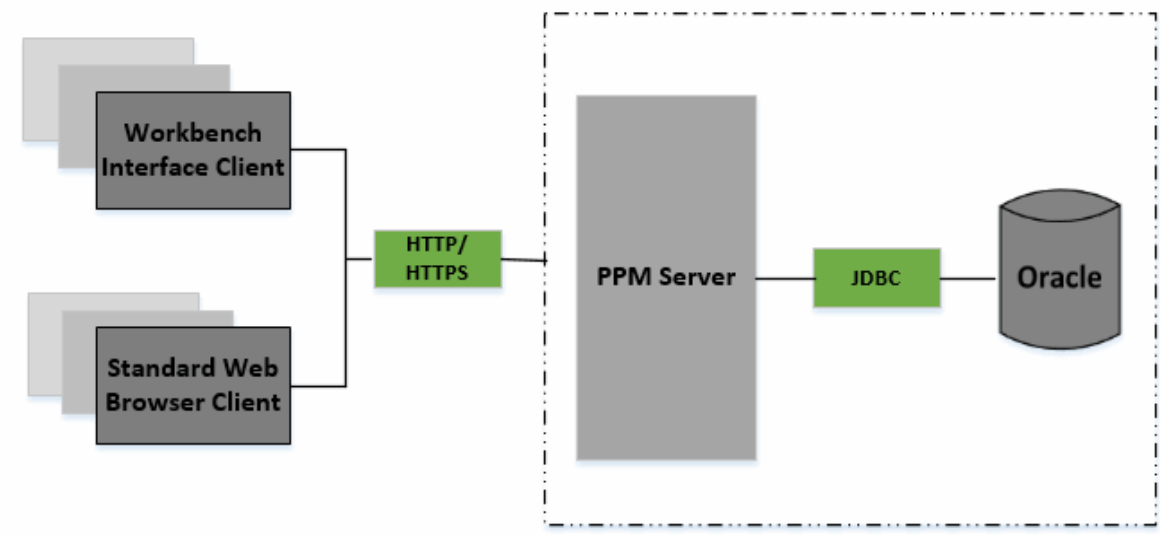
> **Note:** In a production environment, you must deploy a server cluster. A stand-alone configuration is adequate only in a development or testing environment.

Single-Server/Single-Machine Configuration

The single-server/single-machine configuration shown in the following figure consists of one machine that hosts both the PPM Server and the Oracle database.

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

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, RMIS.

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

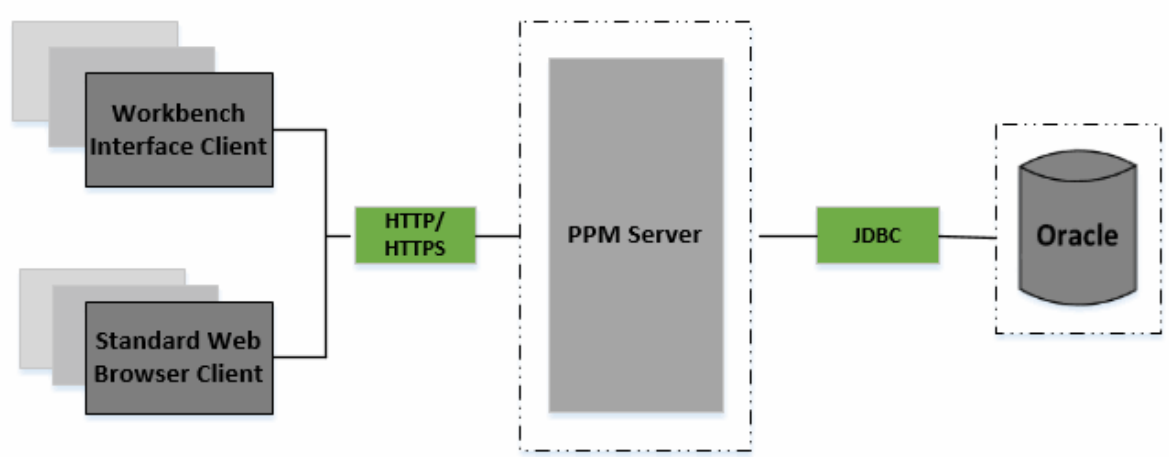
An organization typically uses this configuration if it requires a dedicated machine for all PPM services and database operations. User load, transaction capacity, and system performance depend on the available resources on a machine. This configuration does not support load balancing or server failover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

For information about how to set up a single-server/single-machine configuration, see "Installing PPM" on page 33.

Single-Server/Multiple-Machine Configuration

In the single-server/multiple-machine configuration shown in the following figure, 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.

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 RMIS for secure communication. The PPM Server and Oracle database use JDBC to communicate.

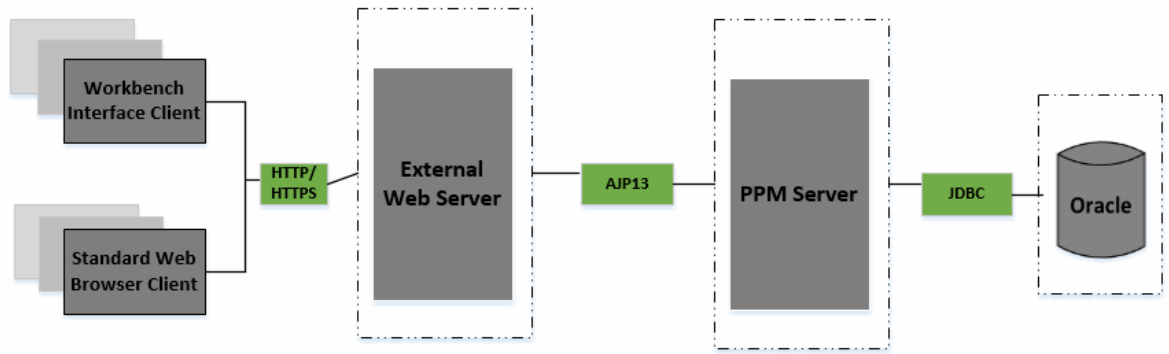
An organization typically uses the single-server/multiple-machine configuration if it requires a separate machine for database operations. User load, transaction capacity, and system performance depend on the resources available on the PPM Server machine. This configuration does not support load balancing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

For information about how to set up a single-server/multiple-machine configuration, see "Installing PPM" on page 33.

Single-Server/External Web Server Configuration

In the single-server/external Web server configuration shown in the following figure, Web traffic comes into the Web server and is then passed to PPM. 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.

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, RMIS.
- 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 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, HPE 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.

> **Note:** HPE 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 "Installing PPM" on page 33 and "Advanced System Configuration" on page 126.

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 documents, see "Accessing PPM Documentation" on page 17

Chapter 3: Installing PPM

This chapter contains the following topics:

Key Considerations

To prepare to install PPM, review the following topics:

> **Note:** For information about how to upgrade to PPM version 9.40 from an earlier version, see the *Upgrade Guide*.

Installing for the First Time

To prepare to install PPM version 9.40:

1. Make sure that your organization has obtained the installation software.
2. Read the rest of this section.
3. Read the *System Requirements and Compatibility Matrix,* which is described in "Related Documents" on page 16.
4. Read the *Release Notes,* which are described in "Related Documents" on page 16.

5. If you plan to install the PPM 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 Deployment Management Extensions, see the documentation for the product.

   > **Note:** For information on how to access documentation for HPE Migrators and Deployment Management Extensions, see "Accessing PPM Documentation" on page 17.

8. Make sure that you have generated Autopass license files for all of the products you already purchased and plan to install. For instructions, see "Activating and Generating Autopass License" on page 86.
9. Install PPM. For instructions on how to install PPM, see "Installing PPM" on page 60.
10. Configure the PPM Server and system environment.

    For information about how to configure PPM, see "Configuring the System" on page 76 and "Advanced System Configuration" on page 126.
11. Install and configure optional products you have purchased to work with PPM.

    > **Note:** After you install PPM, you can install Extensions, or the GL Migrator in any order you choose. For information about how to install and configure optional products, see "Optional Installations" on page 73.

PPMApplication Lifecycle ManagementBundle

The Application Lifecycle Management (ALM) bundle is included in PPM to facilitate integration of PPM with other HPE Software products. PPMALM bundle provides entities such as request types, workflows, portlets, reports, and special commands that you can install in PPM to support Information Technology Infrastructure Library (ITIL) processes for change management and release management.

ALM entities enhance PPM functionality, and facilitate PPM integration with HPEService Manager, HPE Universal Configuration Management Database (Universal CMDB), HPEQuality Center and HPE Application Lifecycle Management, and HPE Release Control. For detailed information about ALM bundle and how to install it, see the *Solution Integrations Guide.*

MSP Plug-in for PPM

You can integration Microsoft Project with PPM Project Management by using either the MSP plug-in for PPM or the mpp-file integration method. Ne aware of the following only if you plan to integrate Microsoft Project with Project Management by installing the Plug-in for PPM:

- The Plug-in for PPM is a Visual Studio Tool for Office (VSTO). Net Office add-in that adds a menu to Microsoft Project. You can use the menu to synchronize Microsoft Project and 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 (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 *Project Management User's Guide.*

Installing Object Migrator or GL Migrator

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

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

Installing an Deployment Management Extension

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

Obtaining License Key Files

Make sure that you have purchased the HPE products you intend to install (you can purchase and install additional products later) and that you have Autopass license key file(s) for the purchased version. Customers with valid entitlement order number can go to HPE Licensing for Software portal at http://www.hpe.com/software/entitlements by entering your Entitlement Order Number (EON) to activate your license. For instructions, see "Activating and Generating Autopass License" on page 86.

Checking System Requirements

Before you start to install PPM, 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 documents, see "Accessing PPM Documentation" on page 17.

Key Decisions

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

Table 3-1. Decisions to make before you install

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 88.
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, 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 52. 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 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.
What privileges do I grant the database schema user?	To rebuild the statistics, the PPM 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>/ppm930/sys`

Table 3-1. Decisions to make before you install, continued

Decision	What to Consider
	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 Best Practices Installation" below.
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 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 Portfolio Management and Program Management.

Best Practices Content

Project Management and Portfolio Management access the request type content installed as part of Best Practices. This includes HPE-supplied menu items that access these request types. "Table 3-2. Best Practices request types " on the next page lists the Best Practices request types and their associated workflows, and provides descriptions of the product functionality they provide. "Table 3-3. Best Practices workflows" on page 40 lists the functionality that Best Practices workflows content enabled in this version of PPM.

Demand Management and 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 database schema name
- PPM database username and password

- PPM administrator username and password
- You have licenses for both the Portfolio Management and Program Management.

For detailed instructions on how to install Best Practices separately, see "Installing HPE Project and Portfolio Management Best Practices" on page 73.

The Best Practices content supplied with this version of PPM includes the request types and workflows listed in "Table 3-2. Best Practices request types " below and "Table 3-3. Best Practices workflows" on page 40.

> **Caution:** In order for PPM 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

Request Type and Description	Menu Selections	Field Groups
PFM - Proposal Represents a project proposal within the 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 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
Program Details	• Create > Program • Open > Program Management > Create Program	PFM Program
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 Reference • Project Issue

Table 3-2. Best Practices request types , continued

Request Type and Description	Menu Selections	Field Groups
	Project Controls > Search Project Issues Also associated with a project type to enable creating from within the Project Overview page.	
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
Program Risk Represents risks associated with a	• Search > Program Risks • Create > Program Risk	• Program Reference

Table 3-2. Best Practices request types , continued

Request Type and Description	Menu Selections	Field Groups
program	• Open > Program Management > Risks > Search Program Risks • Open > Program Management > Risks > Submit Program Risk	• 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).

The following table lists the functionality that Best Practices workflows content enabled in this version of PPM.

Table 3-3. Best Practices workflows

Best Practices Workflow	Description
PFM - Proposal	Portfolio Management process for requesting a new project.

Table 3-3. Best Practices workflows, continued

Best Practices Workflow	Description
PFM - Asset	Portfolio Management process for an asset life cycle.
PFM - Project	Portfolio Management process for a project life cycle.
Program Process	Portfolio Management process for a program life cycle.
DEM - Enhancement Request Process	Demand Management process for enhancement requests for new functionality in applications.
DEM - Database Refresh	Demand Management process for database refresh requests.
DEM - Bug Request Workflow	Demand Management process for application bug requests.
DEM - Project Initiative Process	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 Best Practices content is available for all supported language packs. HPE recommends that you deploy language packs soon after you install Best Practices so that the its content matches in all languages.

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

Preparing to Install PPM

Before you start to install PPM, 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.

> **Caution:** 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.)
>
> 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.

> **Note:** Total and free physical memory and operating system swap file space are logged during PPM Server startup. The exception to this is AIX systems, where this information is not available.

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 PPM Server" on page 88.

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

   PPM 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.

   > **Note:** HPE 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 44.

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 47.

8. Verify that the `JAVA_HOME` environment variable is set.
9. (Windows systems only) On each Windows server with which PPM 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.

    To install PPM 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 " on page 51.
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 Oracle database parameters.
    - Set Oracle database parameter `NLS_LENGTH_SEMANTICS` value to `CHAR;`
    - Set Oracle database parameter `NLS_CHARACTERSET` value to `AL32UTF8.`

    > **Note:** For new PPM installations, make sure you ALWAYS use `AL32UTF8`.

    For information about setting Oracle database parameters, see "Configuring or Reconfiguring the PPM Server" on page 88.
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 59.)

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

> **Note:** The placeholder <*PPM_Home*>, which is used throughout this document, refers to the root directory where PPM 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*>.

Collecting Required Information

The PPM 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. The following table lists the information required for installing PPM on either a single PPM Server or the primary node for a server cluster.

> **Note:** 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 165.

Table 3-4. Required installation information

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. HPE recommends that you not share this directory.
Path to the Autopass license key file	The Autopass license key file contains valid PPM license keys that you generated from the HPE Licensing for Software portal. If you do not have a valid Autopass license key file, see "Activating and Generating Autopass License" on page 86.
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. Windows example

Table 3-4. Required installation information, continued

	`C:\jdk1.7` **Note:** Make sure that the value specified for `JAVA_HOME` contains no spaces.
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. **Note:** Do not use a string like "hpswDemo09" as a password during the installation, because the InstallAnywhere treats "09" as a parameter and consequently turns "hpswDemo09" into "hpswDemo".

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)))`

Table 3-4. Required installation information

Prompt	Description
ORACLE_HOME	Home directory for the Oracle client tools on the PPM Server machine. The directory path cannot contain spaces.
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 Schema page (Displayed if you created the database schemas before installation)	
Username	PPM database schema user name.
Password	PPM database schema password.
RML Schema page	
Username	Username for the PPM Reporting Meta Layer (RML) schema.
Password	Password for the PPM 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.
Regional Settings	
Holiday schedule	Holiday schedule on which to base the PPM 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 *Financial Management User's Guide.*

Table 3-4. Required installation information , continued

Prompt	Description
	Caution: 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.

Unzipping the Installation Files

Before you run the installation driver script, extract your installation files for the PPM software to the *<PPM_Extract>* directory. The extraction procedure creates a new subdirectory named `ppm940`. Run the extraction command in a directory other than the *<PPM_Home>* directory.

Note: Do not change the `ppm940` 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 are available from either Oracle or from the operating system vendor.

Note: 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*.

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.

> **Note:** 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.

   > **Note:** 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.

   > **Note:** For information about how to check for and set the `JAVA_HOME` environment variable, see "Verifying that the JAVA_HOME Environment Variable Is Set" on the next page.

(Optional) Configure JDK to Use the Unlimited Strength Java Cryptography Jars

PPM supports control over the encryption suites used by its SSL (TLS) sockets. This can be specified by the server configuration parameter `SSL_ENCRYPTION_SUITES`.

The value for this parameter should contain a comma-separated list of the encryption suites to be made available to PPM Centre. These should be specified using the standard SSL/TLS cipher suite names.

For example, to specify that PPM should only establish connections using the TLS_DHE_RSA_WITH_AES_256_CBC_SHA cipher suite:

```
com.kintana.core.server.SSL_ENCRYPTION_SUITES=TLS_DHE_RSA_WITH_AES_256_CBC_SHA
```

If using AES256 or similarly strong encryption, the JDK used by both PPM and the client must be configured to use the unlimited strength Java cryptography jars, if this is permissible in your jurisdiction and under US export laws.

> **Note:** The `SSL_ENCRYPTION_SUITES` parameter only impacts the encryption algorithm used for RMIS traffic. There is no impact on HTTPS (SSL) encryption, nor on how the passwords and

sensitive data are encrypted in PPM.

To configure your JDK to use the unlimited strength Java cryptography jars,

1. Go to http://www.oracle.com/technetwork/java/javase/downloads/index.html.
2. Scroll down to the end of the page and download the unlimited strength Java cryptography jars that match your JDK version.

   For JDK 1.7.0, download Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 7 (`UnlimitedJCEPolicyJDK7.zip`).
3. Extract the downloaded zip package.
4. Copy the `local_policy.jar` and `US_export_policy.jar` files to the `<JDK_HOME>/jre/lib/security` directory on both your server side and client side to replace the existing files.

   Note: If you enabled secure RMI and are using a high strength encryption suite, such as AES256, make sure to follow the steps above to install the unlimited jars on machines which will run workbench.

Verifying that the JAVA_HOME Environment Variable Is Set

PPM 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.

Caution: 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 7. 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 window.
3. Click **Advanced system settings**in the navigation pane.

   The System Properties dialog box opens.
4. Click **Environment Variables** on the Advanced tab.
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, and click **OK**.
8. Locate the **Path** variable under **System Variables**, click **Edit**.

   The Edit System Variable dialog box opens.
9. In the **Variable Value** box, add `%JAVA_HOME%\bin` to the existing value and click **OK**.

Setting the JAVA_HOME value in Cygwin

To set the JAVA_HOME value in Cygwin,

1. Add the the JAVA_HOME parameter to System Variables.

   For detailed instruction, see Setting the JAVA_HOME Value in Windows section above.
2. (Optional, but recommended) Set the `.bash_profile` file.
   a. Open the `<Cygwin_HOME>/.bash_profile` file in a text editor.
   b. Add the following to the end of the file:

      ```
      export JAVA_HOME=/cygdrive/<Drive_Letter>/<JDK_HOME>
      export PATH=$JAVA_HOME/bin:$PATH
      ```
   c. Save the `.bash_profile` file and quit.

Setting the JAVA_HOME Value in DOS

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

```
set JAVA_HOME=<JDK_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=<JDK_Install_Directory>; export JAVA_HOME
```

Creating a System Account for PPM

To install PPM 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 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 and all of its subdirectories. Provide the Administrators screen group with at least read access to these directories.

Configuring a PPM User for PPM in UNIX

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

Configure your PPM user with the following:

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

Installing a UNIX Emulator (Windows)

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

> **Note:** To configure private key authentication with secure shell (see "Configuring Private Key Authentication with Secure Shell" on page 92), 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 database schema.

   > **Tip:** Even though a fresh installation of PPM typically requires less than 1 GB of database space, its size could sharply increase with time, especially if you choose to store attachments in the database. For an accurate estimation of your DB space requirement on the long term, contact HPE Software Support.

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

   > **Note:** 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;
   ```

> **Note:** 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 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 installation bundle as outlined in "Installing PPM" on page 60.

   The `<PPM_Extract>/ppm930` directory is created. The `<PPM_Extract>/ppm930/sys` and `<PPM_Extract>/ppm930/system` directories contain the scripts required to create the database schemas.

2. Run the script `CreateKintanaUser.sql` against the database into which you plan to install PPM Center

   The script is located in `<PPM_Extract>/ppm930/system`. It prompts for a user name and password, and the tablespaces that the PPM 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 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 database schema user name. The script creates the RML schema and establishes the permissions between the RML and the PPM database schema.

> **Note:** 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>/ppm930/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 Schemas

By default, the PPM database schema and RML database schema accounts (PPM_USER and RML_USER) are granted Oracle database privileges from an overall PPM perspective, which works for all customer scenarios and environments. Some of these privileges are required, but some are not, and can be revoked without affecting the PPM system.

The PPM 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 database schema account.

Although revoking the CONNECT role privilege does not affect the PPM system, the PPM 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
- grant create job
- grant execute on ctxsys.ctx_ddl

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

Other Permissions Needed or Not Needed for PPM Schemas Accounts

This section provides more information about other permissions needed or not needed for PPM database schema and RML database schema accounts. You can decide whether you want to have them revoked after reading this section.

HPE has not tested every access grant and its impact, because every access grant has a different impact on each customer, as every customer is using PPM differently and for different scenarios. If you are looking for specific access grant impact, you may need to test that out.

- SELECT_CATALOG_ROLE and SELECT ANY DICTIONARY

  If only SELECT_CATALOG_ROLE is enabled then it provides access to all SYS views only.

  If only SELECT ANY DICTIONARY privilege is enabled then it provides access to SYS schema objects only.

  If both SELECT ANY TABLE privilege and SELECT ANY DICTIONARY privilege are enabled then it allow access to all SYS and non-SYS objects.

PPM needs to access many SYS/DBA views and objects, thus needs SELECT_CATALOG_ROLE role and SELECT ANY DICTIONARY privilege.

To generate AWR/ADDM/ASH reports, PPM needs SELECT_CATALOG_ROLE role too.

- PPM need access to the following packages:
  - DBMS_JAVA
  - DBMS_JAVA_TEST
  - DBMS_LOB
  - DBMS_SCHEDULER
  - DBMS_SQL

  PPM does not need access to the following packages:
  - UTL_FILE
  - UTL_HTTP
  - UTL_TCP

  For other packages, read the information below to decide whether your PPM still needs access to them:
  - DBMS_JAVA — PPM uses a stored procedure written in Java, to generate a hash that is used as REFERENCE_CODEs for various PPM entities. Generating this hash using pure PL/SQL is cumbersome and unreliable, if not impossible. So, PPM used Java for it and the application code needs access to this package.
  - DBMS_JAVA_TEST — Allows you to test Java Stored Procedures. PPM might not need to access this package. So far there seems no harm or implications of revoking the access.
  - DBMS_LOB — PPM stores a lot of data in BLOB and CLOB columns and the application code might need to parse or modify the contents.
  - DBMS_SCHEDULER — Not needed for core PPM. This is required for Operational Reporting. If your organization does not use Operational Reporting, you may revoke access to this package.
  - DBMS_SQL — PPM generates dynamic SQL and executes it during installation and upgrade. This package is also used to create triggers as part of the application functionality.
- EXECUTE ANY PROCEDURE and EXECUTE ANY PROGRAM

  PPM does not need the EXECUTE ANY PROCEDURE privilege.

  PPM needs the EXECUTE ANY PROGRAM privilege. The definition of EXECUTE ANY PROGRAM is as follows:
  Use any program in a job in the grantee's schema.

- CREATE ANY VIEW, CREATE ANY TABLE, SELECT ANY TABLE

  RML_USER needs these three privileges:

  - CREATE ANY TABLE — Create tables in any schema. The owner of the schema containing the table must have space quota on the tablespace to contain the table.
  - SELECT ANY TABLE — Query tables, views, or materialized views in any schema.
  - CREATE ANY VIEW — Create views in any schema.

- Privileges required for RML database schema

```
grant create session to &RML_USERNAME;
grant create table to &RML_USERNAME;
grant create view to &RML_USERNAME;
grant create synonym to &RML_USERNAME;
grant resource to &RML_USERNAME;
```

For information about other Oracle database privileges, see Oracle documentation.

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.

> **Note:** The shared folder described in this section is also required to give users access to the Administration Console interface after you deploy your PPM Center instance. For information about the Administration Console, see "Tools in the Administration Console" on page 272.

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

1. Create a shared folder on a file server.

   > **Note:** 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.
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>/<Local_Drive_Letter>$/<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 nodes on Windows. The value of the environment variable is the location of the shared folder.

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

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

1. Create a shared folder on a file server.

> **Note:** 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.
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, add the following line:

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

Verifying Port Availability

To successfully install and configure PPM, 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. The following table contains summary information about the ports and protocols that PPM system components use to communicate.

> **Note:** 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. PPM Center ports and protocols

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 / RMIS	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	SSH	22
App Server and External System	FTP (control)	21

Table 3-6. PPM Center ports and protocols, continued

Communication Channel	Protocols	Ports
App Server and External System	FTP Data	Dynamic
App Server and External System	SCP (Secure Copy)	22
App Server nodes	TCP/UDP	Customized in `<PPM_Home>/conf/integrity.conf`, `cache.conf`, and `<PPM_Home>/conf/jboss/clusterservices.xls`. It depends on your requirement.

Assigning Ports Outside of the Ephemeral Port Range

PPM 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

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 database objects are to be installed.

> **Caution:** 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 on a Windows System" on the next page
- "Installing PPM on UNIX Systems" on page 66

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

> **Caution:** You cannot install PPM unless you have SYS DBA privileges or a DBA has already created the required schemas. For more information, see "Key Decisions" on page 36.

To install the PPM Server on Windows:

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

   > **Note:** For a list of supported UNIX emulators, 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 92), 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-940-install.zip` to the file system.

   The extraction creates the `<PPM_Extract>/ppm940` directory, which includes the `install.sh` file.

3. From the `ppm940` directory, use Windows Command or Cygwin to run the `install.sh` script.

   The PPM installer starts.

   > **Note:** (Windows 2012 R2 platform only) If you encounter an error and fail to launch the PPM installer, you need to modify the properties of the `ppm_install.exe` file for the installer to run properly. To do so,
   >
   > a. Locate and right click the `ppm_install.exe` file in the `<PPM_Extract>/ppm940/Disk1/InstData/Windows/NoVM` directory.

b. Select **Properties** from the popup menu.

c. In the ppm_install Properties window, go to the **Compatibility** tab.

d. In the **Compatibility mode** section, select the **Run this program in compatibility mode for:** checkbox and select a desired option from the drop-down list.

e. Then in the **Privilege level** section, select the **Run this program as an administrator** checkbox.

f. Click **Apply**.

g. Click **OK**.

h. Double click the `ppm_install.exe` file to launch the PPM installer.

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) Swedish	English Spanish Italian French Dutch German Portuguese (Brazilian) Swedish
Russian	Russian English
Chinese	Chinese English
Korean	Korean English
Japanese	Japanese English
Turkish	Turkish English

Operating System Language	Languages Available for Displaying Installation Wizard Steps
Arabic	Arabic English

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

Limitation: If the Windows system language is Chinese, Japanese, or Korean, and the system has JDK 8 installed, PPM installer wizard does not display language list.

Workaround: This is the limitation of the third-party software InstallAnywhere. To work around this issue, run the following command to set the locale for the PPM installer wizard:

```
sh install.sh -l <language_code>
```

where language codes for Chinese, Japanese, and Korean are `zh_CN`, `ja`, and `ko` respectively.

> **Note:** 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 Choose License page, select **Yes, I have a license and want to install it now** to install an Autopass license file that is sent to you by HPE. Or, select **No, thanks. I will install the license later.** to generate and activate a 60-day trial license.
7. On the next several pages, provide the information you collected to prepare for the installation (see "Collecting Required Information" on page 44 in "Collecting Required Information" on page 44).
8. From the **Select the system language** list, select the PPM 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*.
9. Select the check boxes for any languages you want to deploy in addition to the system language on your instance.

   > **Note:** 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, see the *Multilingual User Interface Guide*.
10. On the server mode step, do one of the following.

    > **Note:** 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 77.

    > **Note:** 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 77.

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

PPM Server, leave **Stand-alone** selected.

The Configuration Option page opens next.

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

> **Note:** For instructions on configuring the server later, see "Configuring the System" on page 76.

12. 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 "PPM Configuration Parameters" on page 401.

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

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

> **Note:** 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>/ppm940/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).

14. After successful installation, PPM 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 service from the Control Panel (**Start > Administrative Tools > Services.**)

HPE 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 user, specify this user name for the "Log On As" value.

> **Note:** PPM comes with an Administrator user with the default username/password combination admin/admin. HPE recommends that you change the password for the administrator user after you install PPM.

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 documentation and an uninstall program.

If you did not configure the PPM Server during installation, see "Configuring or Reconfiguring the PPM Server" on page 88.

Installing PPM on UNIX Systems

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-940-install.zip`

   Alternatively,

   `jar xvf ppm-940-install.zip`

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

   The `<PPM_Extract>/ppm940` 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) 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.
`-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_$PARAMETER to <PPM_Schema>`
- `grant execute on DBMS_STATS to <PPM_Schema>`
- `grant select on v_$MYSTAT to <PPM_Schema>`
- `grant select on v_$PROCESS to <PPM_Schema>`
- `grant select on v_$SESSION to <PPM_Schema>`

The `GrantSysPrivs.sql` script (located in the `<PPM_Extract>/ppm940/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).

> **Note:** PPM comes with an Administrator user with the default username/password combination admin/admin. HPE recommends that you change the password for the administrator user after you install PPM.

Configuring the FTP Server on Windows

PPM 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.

Before you configure the FTP server on a machine, make sure that the Windows user account (which PPM 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).

> **Note:** On Windows, most FTP servers, including Microsoft IIS, do not support drive letters. If you use FTP in PPM, the drive letter is removed from the base path. If your base path is `d:\ppm940`, 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 `/ppm940` to `D:\ppm940`.)

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 can do this too.

Contacting Support

If you encounter problems with your installation or if have questions, contact HPE Software Support Web site (https://softwaresupport.hpe.com). Before you contact HPE Software Support, have the following information ready:

1. Open the `ppm_install.log` file (located in the `<PPM_Home>/install_940/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_940/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

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

> **Note:** 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 36.

PPM service packs are distributed as deployments (software bundles that contain files and data). You can get service packs from the HPE support Web site.

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

1. Go to the Patches search page (https://softwaresupport.hpe.com/group/softwaresupport) on the HPE Software Support Web site.
2. In the select search filters section on the left pane, 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.

3. Provide any optional search 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 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.

> **Note:** If multiple languages are deployed on your PPM 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.

   > **Note:** 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 a Single-Server System" on page 77.

3. Run the following command:

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

4. As the script runs, respond to the prompts.
5. Start the PPM Server.

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

For more information about the `kDeploy.sh` script, see "kDeploy.sh" on page 495

Contacting Support

If problems occur during service pack installation, go to the HPE Software Support Web site (https://softwaresupport.hp.com).

Protecting Backed-Up Data

Because the backups (both server host and database backup) you create may contain sensitive information such as cryptographic keys and payload data, HPE strongly advises that you protect the backups themselves. Oracle Advanced Security provides transparent data encryption of data stored in the database, the encryption of disk-based backups of the database, and network encryption for data traveling across the network between the database and client or mid-tier applications. In addition, it provides a complete suite of strong authentication services to Oracle Database.

If you want to use Enterprise User Security in Oracle Database Enterprise Edition, you must license Oracle Internet Directory (OID). If you want to use stronger authentication alternatives (such as Kerberos or PKI) for enterprise user security, you must license Oracle Advanced Security and the Oracle Internet Directory (OID). For more information, see the release notes for your Oracle software.

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.

  For 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
```

Uninstalling a Service Pack

When you deploy a service pack bundle, a basic full backup of your PPM system is performed, including the following system folders: `bin`, `conf`, `icons`, `integration`, `lib`, `pdf`, `rml`, `scripts`, `server`, `sql`, `utilities`, `work`, `security`.

The Patch Uninstaller included in this version allows you to uninstall the service pack files and roll back your PPM file system to the status before the service pack was deployed, using the full backup files. Note that only the File System changes are rolled back. After rollback of the service pack, make sure you manually restore PPM Center database to the status before the service pack was deployed, otherwise the system will NOT work properly.

To uninstall a service pack and roll back the file system to its previous status,

1. Stop PPM Server.
2. Navigate to the `<PPM_Home>/bin` directory.
3. Run the following command:

   ```
   sh ./kDeploy.sh -rollback
   ```
4. Follow the on-screen instructions when prompted.
5. Restart PPM Server.

> **Note:** The rollback operation does not roll back the `kDeploy.sh` script itself.

Verifying PPM Installation

To verify the installation:

1. Check the logs produced during installation. (Located in the `<PPM_Home>/install_940/logs` directory)
2. Complete the PPM configuration and perform all post-installation tasks (covered in later sections).
3. Start the PPM Server.

4. Log on to PPM.

   > **Note:** All PPM 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 307.)
7. Create a request and test the graphical view of the request. (For instructions, see the *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.)

   > **Note:** Before you can export a page in PDF format, you must enable that functionality. For information, see "Enabling Export to PDF" on page 104.)

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

Installing HPE Project and Portfolio Management Best Practices

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

For more information, see "About PPM Best Practices Installation" on page 37. Before you start to install Best Practices, make sure that *all* of the conditions described in "Requirements for Installing Best Practices" on page 37 have been met.

To install Best Practices:

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

   > **Note:** Although setting your server to RESTRICTED mode is optional, HPE 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 77.)

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

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

   > **Note:** For more information about the `kDeploy.sh` script, see "kDeploy.sh" on page 495.

Verifying HPE 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 Deployment Management Extensions

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

You are not required to stop the PPM Server(s) before you install an Extension. However, HPE 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.

> **Note:** Although setting your server to RESTRICTED mode is optional, HPE 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 77.)

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

> **Note:** 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 36.

What to Do Next

After you have successfully installed PPM, delete all subdirectories of the `install_930` directory, except for the `logs` subdirectory.

Proceed to "Configuring the System" on page 76.

Chapter 4: Configuring the System

This chapter provides detailed information about how to configure the basic components of the PPM 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.

> **Note:** For advanced PPM system configuration information, including how to configure an external Web server and PPM Server clusters, see "Advanced System Configuration" on page 126.

This section contains the following topics:

(UNIX only) Setting the ulimit Value

On UNIX systems, PPM (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.

> **Tip:** You can simply remove the setting mentioned in step 2 from the `kStart.sh` file to allow the UNIX operating system to control the number of open file descriptors.

Starting and Stopping the PPM Server on a Single-Server System

This section provides information about how to start the PPM Server on a single-server system.

> **Note:** 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 "System Configurations" on page 22 and "Configuring a Server Cluster" on page 165.

Setting the Server Mode

PPM 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 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 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 the previous page. For more information about the `kConfig.sh` script, see "kConfig.sh" on page 493.

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

This section covers starting and stopping the server on a Windows system.

> **Caution:** If your PPM 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.

If your instance is configured as a server cluster *and* the `PPM_SERVER_CONF_DIR` environment variable is set, then before you start the PPM Server, you must configure user account logon options for the PPM service. If you do not, the service will not start. (It is only necessary to do this once.)

Configuring user account logon options for the PPM service

1. From the Control Panel, select **Administrative Tools > Services.**
2. Right-click the PPM service, and then select **Properties** from the shortcut menu.
3. In the properties dialog box, click the **Log On** tab.
4. Select the **This account** option, and then type `.\Administrator` in the text box.
5. Type your administrator password in the **Password** and **Confirm password** boxes.

Starting the PPM Server on Windows

1. From the Control Panel, select **Administrative Tools > Services.**
2. Right-click the PPM service, and then click **Start** on the shortcut menu.

> **Note:** If you prefer to use the Windows shell command line to start servers, you can use the kStart.sh script. For information about the kStart.sh script, see "Server Directory Structure and Server Tools" on page 490.

Stopping the server on a Windows system:

1. From the Control Panel, select **Administrative Tools > Services.**
2. In the Services window, right-click the PPM service, and then click **Stop** on the shortcut menu.

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 "Server Directory Structure and Server Tools" on page 490.

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 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 452.

Starting and Stopping the PPM Server on UNIX

To start the server on UNIX.

> **Caution:** If your PPM 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 509. For information about how to start servers in a cluster, see "Starting and Stopping the PPM Server on a Single-Server System" on page 77.

To stop the server on UNIX.

> **Note:** 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 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 452.

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 510. For information about how to stop servers in a cluster, see "Starting and Stopping Servers in a Cluster" on page 179.

Startup Checks

To help catch common configuration and deployment issues, PPM Center performs basic startup checks before starting. These checks include:

- Making sure that essential `server.conf` parameters are present.
- Making sure that none of the ports required by PPM Center are already bound by other processes.
- Making sure that all ports used by the nodes on a given physical server have unique values.

> **Note:** Because of these startup checks, you may encounter more configuration errors in 9.40 (and beyond) that prevent a PPM Server from starting (whereas a successful start was allowed on a previous version).

A summary of all issues found during the startup checks is traced out into the server log of the node being started. If PPM Center is running as a Windows service, an error is added to the Windows event log asking you to refer to the server log when errors are found. In some rare cases that the server logs could not provide useful details, we recommend that you start PPM Center in the console mode by using the `kStart.sh` script, and then troubleshoot the configuration and deployment issues. After you fix these issues, you can start PPM Center as a Windows service again.

Errors fall into two categories:

- **Critical errors (Error messages)**: These errors relate to the node being started. PPM Center will refuse to start if there are any critical errors.
- **Non-critical errors (Warning messages)**: These errors relate to other nodes in server.conf than the one being started. It is highly recommended to fix such errors, however they will not prevent the startup. All nodes in the `server.conf` file are checked when any one node is started.

Bypassing the Startup Checks

By default, the `com.kintana.core.server.BYPASS_STARTUP_CHECKS` parameter in the `server.conf` file is set to `false`.

You can bypass the startup checks by setting this parameter to `true`:

```
com.kintana.core.server.BYPASS_STARTUP_CHECKS=true
```

> **Caution:** This parameter should only be used under advisement from PPM customer support engineers, and if there is a clear and well-understood reason for doing so.

Comprehensive Logging During PPM Server Startup

PPM includes a significant number of improvements to the application startup process, from both a logging and configuration validation standpoint. One of the goals of the improvements is to provide customers with as much information as possible about the validity of the configuration of their clustered environments.

Displaying Failed Executions at Startup

To enable the display of failed server executions at startup, make sure that the `FAIL_EXECUTIONS_ON_STARTUP` server configuration parameter is set to `true` in the `server.conf` file. All executions that were interrupted during the last PPM shut-down are marked as failed.

Displaying Configuration Parameters at Startup

You can enable the display (and logging) of all PPM server configuration parameters used during startup. To enable this feature, set the `SHOW_PARAMETERS_AT_STARTUP` server configuration parameter to `true.`

Adjusting the Server Log Level

Server startup logs indicate what the server is doing at each step of the startup process (including the successful start of each web context). By default, when the startup check catches a configuration or deployment issue, only a error or warning message will be generated in the server logs. You can modify the logging level to get more details by adding the following entries in the logging.conf file:

```
com.kintana.core.logging.SYSTEM_THRESHOLD = INFO
```

```
com.kintana.core.logging.PRODUCT_FUNCTION_LOGGING_LEVEL =
com.kintana.core.server.ServerStartupSanity, INFO
```

> **Caution:** These parameters are designed for advanced troubleshooting. We do not suggest modifying these parameters without advisement from PPM customer support engineers.

Installing Autopass Licenses for PPM Product

This section contains the following topics:

Overview of the Autopass Licensing Solution

Since version 9.30, PPM implemented Autopass integration to replace PPM's original licensing mechanism with HPE's Autopass licensing mechanism. The new Autopass licensing solution simplifies the license generation and validation process, supports more product license types, offers flexible license installation options, and simplifies product license management.

- Simplified license activation and validation process.

  To activate and generate an Autopass license, simply go to the HPE Licensing for Software portal (https://www.hpe.com/software/entitlements). See "Activating and Generating Autopass License" on page 86.

  When installing an Autopass license key file, PPM validates the IP address of the PPM Server against that assigned in the license file. For a clustered environment, IP address assigned in the license file shall match that of the primary node in the cluster, otherwise you may receive a "0 license key(s) installed successfully" message.

  Invalid licenses (expired license, or a license with IP address not matching that of the machine where you plan to install the license) will not be installed.

- More product license types available.

  Starting from PPM version 9.30, in addition to the perpetual product license (or term license), trial

license and evaluation license are also available with PPM. This allows new customers to try and evaluate PPM features and functionalities.

- **Trial license.** For fresh install of PPM, a trial license is always automatically generated and activated for you right away, which allows you to try and use PPM modules for 60 days with limited number of users for different modules. Trial license has the lowest priority.

  > **Tip:** To view the features and capacity available with a trial license, you can check the PPM Workbench License Administration window (**PPM Workbench > System Admin > License**).

- **Evaluation license.** An Autopass license key file that you generate from the HPE Licensing for Software portal with a specified start date and end date. The evaluation license allows you to use an authorized set of PPM modules for an authorized period of time. When you install an evaluation license, it overrides the system-default trial license.
- **Perpetual license (or term license).** An Autopass license key file that you generate from the HPE Licensing for Software portal. The perpetual or term license allows you to use an authorized set of PPM modules for an authorized period of time. When you install a perpetual or term license, it overrides the trial license and evaluation license (if available), regardless of how long the valid period for the perpetual or term license is. For example, if your evaluation license expires in three months, but your term license expires in two months, when you install the term license, it overrides the evaluation license and the end date is displayed as that of the term license.

If your evaluation license or term license expires within 60 days of the product installation, or you remove the evaluation license or term license within 60 days of the product installation, the system-default trial license will become effective, until it expires.

The evaluation licenses and term licenses are additive. For example, if different term licenses are installed with different dates, the capacity of different licenses are added together, and the expiration dates for all current licenses are displayed the same in the License Administration window of PPM Workbench, that is, the expiration date for the license that will expire the earliest.

- More flexible license installation options available.

  For new PPM customers, when installing PPM for the first time, you have the following options:

  - If you do not have an evaluation license or a perpetual product license, you can select the **No thanks. I will install the license later.** option in the Choose License page of the installation wizard. A trial license will be generated automatically and activated for you right away, which allows you to use limited features of PPM for 60 days.

Before the trial license expires, if you wish to experience the full features of PPM, you can obtain an Autopass license file from the HPE Licensing for Software portal and install the license file by using either the Administration Console or the newly introduced `kLicenseInstall.sh` tool.

- If you have an evaluation license or perpetual product license , you can install the Autopass license key file by using one of the following ways:
  - In the Choose License window during PPM Center installation process, or
  - After the installation of PPM Center 9.40, before the trial license expires, install the license key file by using either the Administration Console the `kLicenseInstall.sh` tool.

For existing PPM Center customers with active support contract, you can go to the My software updates portal, select **My software updates**, and enter your SAID (Service Agreement ID) number to get PPM Center version 9.40 Software Updates and new license key required for the updates. After successful upgrade, to start PPM Server properly, you must use the `kLicenseInstall.sh` tool to install the Autopass license you received from HPE.

For more information, see "Installing Autopass License Key File and Viewing License Summary in Administration Console" on page 275 and "Installing Autopass License Key File Using the kLicenseInstall Tool" on the next page.

- Simplified product license management.

  A new Administration Console tool (**Administration Console > Administration Task > License**) is added to allow you manage product license easily. Using the Administration Console tool, you can,

  - Install an Autopass license file without having to stop and restart the PPM Server
  - View a summary of licenses installed on the PPM Server, including license capacity and expiration dates
  - Remove an Autopass license file easily

  For more information, see "Installing Autopass License Key File and Viewing License Summary in Administration Console" on page 275.

- Licenses installed are stored in the PPM database, instead of the file system. This makes it possible for the installed licenses to become effective right away after installation. There is no need to stop and restart the PPM Server.
- In a clustered environment, you can install an Autopass license on any node of the cluster. The installed license becomes effective for the entire cluster right away, no need to stop and restart the node. See "Installing An Autopass License in a Clustered Environment" on page 87.
- After starting PPM Server, the system checks whether either of the following foundation licenses is enabled:

- Foundation License Up To 25 Users
- Foundation License Over 25 Users

Activating and Generating Autopass License

To activate and generate your Autopass license for the version of PPM you purchased,

1. Go to the HPE Licensing for Software portal at https://www.hpe.com/software/entitlements.
2. Click **Sign In**.
3. Provide your HP Passport credentials and click **Sign in**.

   > **Note:** If you do not have an HP Passport, click **Create an account**.

4. On the Enter Entitlement Order Number page, enter the Order number found on your Entitlement Certificate and click **Go**.
5. Complete the activation process to generate an Autopass license.

   The generated Autopass license (a .dat file or several .dat files) will be sent to you by HPE.

Installing Autopass License Key File Using the kLicenseInstall Tool

After you have successfully installed PPM, you can install an Autopass license key file by using the kLicenseInstall tool, with PPM Server started or not started.

To install an Autopass license key file,

1. Obtain and save the license file somewhere on your computer.

   For information about obtaining the license key file, see "Key Considerations" on page 33.
2. Open a command prompt.
3. Navigate to the `<PPM_Home>/bin` directory and run the following command:

   `sh ./kLicenseInstall.sh <Autopass_License_File_Path>`

where *Autopass_License_File_Path* is the location of the Autopass license key file that you saved.

The license file is installed and becomes effective right away, with a message popping up showing how many licenses are installed.

> **Note:** You can also install the Autopass license key file using the Administration Console tool. For more information, see "Installing Autopass License Key File and Viewing License Summary in Administration Console" on page 275

Installing Autopass Licenses and Viewing License Summary in Administration Console

The Install License page in the Administration Console allows you to,

- Install an Autopass license key file without having to stop and restart the PPM Server
- View a summary of licenses installed on the PPM Center instance
- Remove a license key

For more information, see "Installing Autopass License Key File and Viewing License Summary in Administration Console" on page 275.

Installing An Autopass License in a Clustered Environment

You can install an Autopass license file on any node of a cluster without having to copy the license file to each of the remaining nodes. The licenses you installed are stored in the database instead of the `license.conf` file, which means that licenses are centralized.

However, always make sure that the IP address assigned in the license file matches that of the primary node in the cluster, otherwise you will receive a "`0 license key(s) installed successfully`" message.

To install licenses in a clustered environment,

1. Obtain and save the Autopass license key file somewhere on your computer.
2. Install the license key file using the `kLicenseInstall.sh` tool or the Administration Console on any of the nodes in the cluster.

   The license file is installed and becomes effective right away, with a message popping up showing how many licenses are installed.

> **Note:** For instructions about installing licenses using the Administration Console, see "Installing Autopass License Key File and Viewing License Summary in Administration Console" on page 275

Reading Licenses Information Using kLicenseReader

You can use the license reader tool to obtain the information of licenses that are installed on your machine.

To use the license reader, you should run the following command:

`kLicenseReader.sh [-filename <Autopass_License_File_Name>] [-filepath <Autopass_License_File_Path>] [-help]`

For more information about this command, see "kLicenseReader.sh" on page 504.

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" on the next page. 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 90.

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 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
     ```

     > **Note:** (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 "PPM Configuration Parameters" on page 401.

   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.

   > **Note:** Always use forward slashes (/) as a path separator, regardless of your operating system environment. PPM 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.

3. If you have no custom parameters to add, leave **Custom Parameters** empty. If you require custom parameters, see "Defining Custom and Special Parameters" on the next page 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 a Single-Server System" on page 77.

   > **Note:** You can also modify parameters directly in the server configuration file, which is described in "PPM Configuration Parameters" on page 401.
   >
   > 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 PPM supplies, PPM 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

  PPM has created configuration parameters that you can use in special situations after you add them to the custom parameters folder. The following table 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 493.

For information about the `kUpdateHtml.sh` script, see "kUpdateHtml.sh" on page 514.

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 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 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 HPE Software Support Web site (https://softwaresupport.hpe.com).	WAR-FTPD
TEMP_DIR	This parameter defines a PPM 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

> **Note:** PPM does not enable SSL by default, for enabling it requires other user information. However, HPE recommends that you enable it, especially in production environment, to make sure data being transmitted is encrypted. The use of SSL protects sensitive information from the risk of eavesdropping, data tampering, or message forgery in the process of transmitting.

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`.

2. In the `server.conf` file, specify values for the following 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
```

> **Note:** HPE does not recommend using self-signed certificates in production environments as they may negate the benefits of end-to-end security by decreasing the ability of a user to detect a man-in-the-middle (MITM) attack.

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 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.

Adding 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 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.

Setting 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 user.
2. Change directory to the home directory defined for the PPM user on the operating system.
3. Run the following SSH utility.

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

   > **Note:** PPM 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.`

   > **Note:** 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.

   > **Note:** 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.

Reconfiguring 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

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

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

   > **Note:** 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.
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 () button.

      The Enter or Change Password dialog box opens.

      > **Note:** The PPM Workbench requires that you provide a password, regardless of whether the authentication uses RSA.
      >
      > If authentication with RSA fails, the password you provide here will be used instead to connect to the remote host.

   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.

> **Note:** 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

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

To configure your instance to use HTTPS using the Tomcat 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>
   ```

   > **Note:** Your <*Store_Password*> and <*Key_Password*> should be the same. If they differ, you will get an error along the lines of `java.io.IOException: Cannot recover key`. For more information, see Tomcat documentation.

For information about importing a third-party certificate, see "Importing a SSL Certificate from a Certificate Authority to Tomcat " on the next page.

> **Note:** HPE does not recommend using self-signed certificates in production environments as they may negate the benefits of end-to-end security by decreasing the ability of a user to detect a man-in-the-middle (MITM) attack.

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

> **Note:** PPM sets this parameter to `false` by default, for enabling SSL requires other user information. However, HPE recommends that you set this parameter to `true` to enable secure web log on. The use of SSL protects sensitive information from the risk of eavesdropping, data tampering, or message forgery in the process of transmitting.

> **Note:** PPM sets this parameter to `false` by default, for enabling SSL requires other user information. However, HPE recommends that you set this parameter to `true` to enable secure web log on. The use of SSL protects sensitive information from the risk of eavesdropping, data tampering, or message forgery in the process of transmitting.

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 431)`

> **Note:** The `HTTPS_PORT` value must be the `HTTP_PORT` number plus 363.

`HTTPS_WEB_THREAD_MIN` (see "HTTPS_WEB_THREAD_MIN" on page 431)

`HTTPS_WEB_THREAD_MAX` (see "HTTPS_WEB_THREAD_MAX" on page 431)

`HTTPS_KEYSTORE_LOCATION` (see "HTTPS_KEYSTORE_LOCATION" on page 430)

`HTTPS_KEYPASSWORD` (see "HTTPS_KEYPASSWORD" on page 430)

> **Note:** 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 "PPM Configuration Parameters" on page 401.

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

> **Note:** For information about how to stop and start PPM Servers, see "Starting and Stopping the PPM Server on a Single-Server System" on page 77.

5. (AIX systems only) If you have PPM Servers running on AIX, stop PPM Server, open the `server.xsl` file (located in the `<PPM_Home>/conf/jboss` directory) and add `algorithm="IbmX509"` to it, as follows:

```
<Connector enableLookups="true" SSLEnabled="true" acceptCount="10" debug="0"
scheme="https" secure="true" clientAuth="false" algorithm="IbmX509" >
```

Importing a SSL Certificate from a Certificate Authority to Tomcat

To import a SSL certificate from a certificate authority to Tomcat for the PPM secure web logon feature, do the following:

1. Create a local Certificate Signing Request (CSR).
   a. On the PPM Server machine, generate a private key with an alias name and a specified keystore file name locally.

      The private key is used to decrypt contents that are encrypted by public key sent by CA authority.

      The alias and specify keystore file name will be used for importing certificate later.

      keytool -genkey -alias <*your_alias*> -keyalg RSA -keystore <*the path and your_keystore_filename, such as c:\myKeystore*>

      > **Note:** When prompted for first and last name, enter the domain of the PPM Server host machine which is used by the `BASE_URL` parameter in the `server.conf` file in order to create a working certificate.

   b. On the PPM Server machine, create a certificate signing request.

      keytool -certreq -keyalg RSA -alias <*your_alias that is the same one used in step a*> -file <*the path and your CSR file name such as c:\certreq.csr*> -keystore <*full path and your_keystore_filename used in step a*>

      The CSR file is used to send to CA authority to request certificate.

   c. Open your CSR file, copy and paste the contents (usually it starts with begin line and ends with end line), and then submit to CA.

When CA receives the CSR file containing your public key, they will sign it with their private key and return the public key and certificate to you in a certificate file.

2. Import the Certificate.
   a. Download a Chain Certificate from the Certificate Authority you obtained the Certificate from.
      - For Verisign.com commercial certificates, go to http://www.verisign.com/support/install/intermediate.html
      - For Verisign.com trial certificates, go to http://www.verisign.com/support/verisign-intermediate-ca/Trial_Secure_Server_Root/index.html
      - For Trustcenter.de, go to http://www.trustcenter.de/certservices/cacerts/en/en.htm#server
      - For Thawte.com, go to http://www.thawte.com/certs/trustmap.html

      Note: Usually the chain certificate has root and intermediate levels. You always import root level first, then intermediate level. If the chain certificate file is not in DER format, then you must convert to DER format before you can import.

   b. Convert the root CA to DER format.
      i. Double click your chain certificate file stored on the PPM Server machine.

         The Certificate dialog opens.

      ii. Click on the **Certification Path** tab.

      iii. Highlight the root certificate (the certificate issued by the signing authority. Example: Entrust, Verisign.)

      iv. Click **View Certificate.**

         A new Certificate dialog for the root certificate opens.

      v. Click the **Details** tab.

      vi. Click **Copy to File**.

         The Export Certificate Wizard opens.

      vii. Click **Next**.

      viii. Select **DER encoded binary for X.509 (.CER)**, and click **Next**.

      ix. Create a new filename to store the newly formatted root certificate and store it on the PPM Server machine.

      x. Click **Finish**.

c. Convert the intermediate CA to DER format.

   i. Double click your chain certificate file stored on the PPM Server machine.

      The Certificate dialog opens.

   ii. Click on the **Certification Path** tab.

   iii. Highlight the Intermediate certificate.

   iv. Click **View Certificate**.

      A new Certificate dialog for the root certificate opens.

   v. Click the **Details** tab.

   vi. Click **Copy to File**.

      The Export Certificate Wizard opens.

   vii. Click **Next**.

   viii. Select **DER encoded binary for X.509 (.CER)**, and click **Next**.

   ix. Create a new filename to store the newly formatted root certificate and store it on the PPM Server machine.

   x. Click **Finish**.

d. Import root CA by running the following command:

```
keytool -import -alias root -keystore <the path of your_keystore_filename
that used in step a) -trustcacerts -file <path and filename_of_the_root_
certificate file>
```

> **Note:** HPE recommends you use alias (such as `root`) here.

e. Import the Intermediate CA.

```
keytool -import -alias intermediate -keystore <the path of your_keystore_
filename that used in step a) -trustcacerts -file <path and filename_of_the_
intermediate_certificate file>
```

f. Import the signed certificate.

> **Note:** If the signed certificate file is not in DER format, follow the steps in "Importing a SSL Certificate from a Certificate Authority to Tomcat " on page 98 to convert the certificate file to DER format before import.

```
keytool -import -alias <your_alias that is the same one used in step a> -
keystore <full path and your_keystore_filename used in step a> -trustcacerts
-file <path and your_certificate_filename>
```

Now the pair of the private key and public key is ready for use in your keystore file permanently.

g. To verify the signed certificate is imported correctly, run the following command:

keytool -list -v -keystore <*full path of your_keystore_filename*>

You should see three entries, root certificate, intermediate certificate as well as the private key, public key and signed certificate listed in the output.

3. Configure PPM.

a. In the `server.conf` file, add the following parameters:

```
#turn on SSL login
com.kintana.core.server.ENABLE_SSL_LOGIN=true

#if you use http_port 8080 then https_port will be set to 8443. Please
verify if your IIS is using port 443, otherwise you have to choose either
stop IIS or use a different http_port and https_port here
com.kintana.core.server.HTTPS_PORT=8443
#
com.kintana.core.server.HTTPS_WEB_THREAD_MIN=5

#
com.kintana.core.server.HTTPS_WEB_THREAD_MAX=75
#keystore file physical location on PPM machine created from step a
com.kintana.core.server.HTTPS_KEYSTORE_LOCATION=C:/Java/keystore_ppm_dev

#Encrypted keypassword. By default it is changeit, and you need to run
kEncrypt.sh to encrypt this password. Each PPM instance has its own
encryption content.
com.kintana.core.server.HTTPS_KEYPASSWORD=#!#7w:x?vv=MdXJ}2&bJbrykTMY3FI>R1
{<+Kw^fjN=hjw8hz2HrTd_X8w+~|Tx19ZiO_oS }rpTHSX(B@)LM{A~c~M<N9GVw,2jLOf
(e=WZNbLo)xarUny.mKp|p{ +1LySpZS flrG{v3&:?k8|<y.y0 b`Kp|G/`s^q.GR|4?s}
&jD$rtamfkqZr?$UT-#!#
```

b. Modify the following parameters:

```
#if your IIS on the same PPM machine has to use port 443, then you must
change PPM http_port to something else other than 8080 and also update
https_port parameter
com.kintana.core.server.HTTP_PORT=8080

#the domain name must match the name from step a when you generate a private
key
com.kintana.core.server.BASE_URL=
http://itprojectsystem_dev.xyz.com:8080/
```

c. Save the file.

d. Stop PPM Server and run **`kUpdateHtml`**.

e. Start PPM Server and test the login using `BASE_URL` such as `http://itprojectsystem_dev.xyz.com:8080/`.

Additional Considerations for Configuring Secure Web Logon

This section describes additional steps required to set up secure Web logon if your users access PPM using Internet Explorer (IE) 9.0, or if you have PPM Servers running on AIX.

Enable Transport Layer Security on Internet Explorer 9.0

Make sure that users who access PPM using Internet Explorer (IE) 9.0 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.)

> **Note:** TLS is enabled by default in 9.0.

Edit the server.xsl File (AIX Only)

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

```
<Connector enableLookups="true" SSLEnabled="true" acceptCount="10" debug="0"
scheme="https" secure="true" clientAuth="false" algorithm="IbmX509" >
```

Generating Password Security (Optional)

For password security, PPM 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 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 database schema)
   - `DB_PASSWORD` (password for the PPM database schema)

     > **Caution:** 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 503.

Configuring Solaris and Linux Environments to Use Deployment Management

PPM can connect to a machine on which the environment variable TERM is set to dumb. To enable 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 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 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 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`

> **Note:** 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 476.

- 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 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.

> **Note:** 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

Operating System	Fonts Location
	• /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 "PPM Configuration Parameters" on page 401.

Enabling IPv6

To enable support for Internet Protocol version 6 (IPv6), add the `ENABLE_IPV6` parameter to the `server.conf` file and set it to `true`.

> **Note:** If the `ENABLE_IPV6` parameter is not present in the `server.conf` file, the system uses IPv4 by default.

In addition, if you want to specify a literal IPv6 address, make sure you enclose the literal address with "[" and "]" characters for the following parameter values:

- `BASE_URL`
- `JDBC_URL`
- `RMI_URL`
- `SERVER_NAME`

> **Caution:** If you want to specify a literal IPv6 address for the `MULTICAST_IP` parameter, do not enclose the literal address with "[" and "]" characters.
>
> To specify a valid IPv6 multicast IP address, see http://www.iana.org/assignments/ipv6-multicast-addresses/ipv6-multicast-addresses.xhtml.

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 as administrator:

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

   > **Note:** All PPM 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 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 provides this default account for logging on the first time. HPE recommends that you disable the admin account or change the password after you generate accounts for all of your users.
4. Click **Sign-In.** The PPM standard interface opens.

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

Accessing the JMX Console

You can use the JMX 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.

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

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

> **Caution:** Do not omit the forward slash after `jmx`, otherwise, the URL does not work.

No user name or password is required. As long as you have the privilege to access the Administration Console, you can access the JMX console.

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 is the only application that uses the database instance. If other applications share the database, adjust the recommended parameter values accordingly.

> **Note:** If Oracle Database is running over Oracle VM (OVM), HPE recommends you enable the Hard Partitioning feature, also known as CPU pinning.
>
> For more information, see Oracle VM documentation: http://docs.oracle.com/cd/E26996_01/E18549/html/vm_hardpart.html

> **Note:** For more recommendations on improving PPM performance, see the *Deployment Best Practices for PPM Operational Reporting.*

Database Parameters

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

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

ARCHIVE_LAG_TARGET

`ARCHIVE_LAG_TARGET` limits the amount of data that can be lost and effectively increases the availability of the standby database by forcing a log switch after the specified amount of time elapses.

A `0` value disables the time-based thread advance feature; otherwise, the value represents the number of seconds. Values larger than 7200 seconds are not of much use in maintaining a reasonable lag in the standby database. The typical, or recommended value is 1800 (30 minutes). Extremely low values can result in frequent log switches, which could degrade performance; such values can also make the archiver process too busy to archive the continuously generated logs.

Recommended Setting

Set the parameter to `0` to help lower "log file sync" waits.

_B_TREE_BITMAP_PLANS

The `_B_TREE_BITMAP_PLANS` parameter enables creation of interim bitmap representation for tables in a query with only binary index(es).

Recommended Setting

Set the `_B_TREE_BITMAP_PLANS` parameter value to `false`. HPE recommends that you set this parameter at the *instance* level instead of at the system level. You can use the `ON LOGON` trigger so that the setting does not interfere with other application schemas that use the database.

To set the parameter to `false`, run:

```
ALTER SYSTEM SET "_B_TREE_BITMAP_PLANS"=false scope=both
```

This prevents Oracle from unnecessary conversions between ROWID and BITMAPS when running queries.

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 supports only the values `SIMILAR` and `EXACT`. `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.

  > **Caution:** Setting `CURSOR_SHARING=FORCE` results in errors during PPM project creation.

- `SIMILAR`. This 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. HPE recommends that you set the `CURSOR_SHARING` parameter to `EXACT`. If any other value is used, PPM Server will try to force the value of this parameter to `EXACT` for all PPM-related databases transactions.

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).

> **Note:** HPE recommends that you set a value for this parameter, even if SGA_TARGET is set.

FILESYSTEMIO_OPTIONS

FILESYSTEMIO_OPTIONS specifies I/O operations for file system files.

Available values include none, setall, directIO, and asynch.

Recommended Setting

Set the FILESYSTEMIO_OPTIONS parameter to setall to enable both direct I/O and asynchronous I/O where possible.

_GC_POLICY_TIME

The _GC_POLICY_TIME parameter allows you to control whether or not to disable Dynamic Resource Manager.

Recommended Setting

Set the _GC_POLICY_TIME parameter to 0 to disable Dynamic Resource Manager, eliminating cluster related waits.

> **Note:** Consult your Oracle support when you want to use this hidden parameter.

_GC_UNDO_AFFINITY

The _GC_UNDO_AFFINITY parameter allows you to control whether or not to disable Dynamic Resource Manager.

Recommended Setting

Set the _GC_UNDO_AFFINITY parameter to False to disable Dynamic Resource Manager, eliminating cluster related waits.

> **Note:** Consult your Oracle support when you want to use this hidden parameter.

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.

> **Note:** If multiple PPM 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.

> **Note:** 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 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

The range of values of OPTIMIZER_INDEX_CACHING is: 0 to 100. The default value is 0.

For most OLTP system, OPTIMIZER_INDEX_CACHING can be set to 90.

For most PPM deployments, set the OPTIMIZER_INDEX_CACHING parameter value to 90.

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 range of values of `OPTIMIZER_INDEX_COST_ADJ` is 1 to 10000. The default value 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.

For most OLTP system, `OPTIMIZER_INDEX_COST_ADJ` can be set from 10 to 50.

For PPM deployment, `OPTIMIZER_INDEX_COST_ADJ` = Full Scan Cost / Index Scan Cost (=20/80 * 100 =25).

Recommended Setting

Although there is no perfect OPTIMIZER_INDEX_COST_ADJ value for every PPM deployment, HPE recommends that you set this parameter to 25.

OPTIMIZER_MODE

The `OPTIMIZER_MODE` parameter establishes the default behavior for choosing an optimization approach for the instance.

Values:

- `first_rows_`*n*

  The optimizer uses a cost-based approach and optimizes with a goal of best response time to return the first *n* rows (where *n* = 1, 10, 100, 1000).

- `first_rows`

  The optimizer uses a mix of costs and heuristics to find a best plan for fast delivery of the first few rows.

- `all_rows`

  The optimizer uses a cost-based approach for all SQL statements in the session and optimizes with a goal of best throughput (minimum resource use to complete the entire statement).

Recommended Setting

Set this parameter to `all_rows`.

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.

PARALLEL_FORCE_LOCAL

`PARALLEL_FORCE_LOCAL` controls parallel execution in an Oracle RAC environment. By default, the parallel server processes selected to execute a SQL statement can operate on any or all Oracle RAC nodes in the cluster. By setting PARALLEL_FORCE_LOCAL to true, the parallel server processes are restricted so that they can only operate on the same Oracle RAC node where the query coordinator resides (the node on which the SQL statement was executed on).

Recommended Setting

Set the parameter value to `true` to avoid parallel operations across the cluster.

PROCESSES

The `PROCESSES` parameter value determines the maximum number of operating system user processes that can simultaneously connect to the Oracle database. PPM 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 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.

RESOURCE_MANAGER_PLAN

RESOURCE_MANAGER_PLAN specifies the top-level resource plan to use for an instance. The resource manager will load this top-level plan along with all its descendants (subplans, directives, and consumer groups). If you do not specify this parameter, the resource manager is off by default.

You can change the setting of this parameter using the ALTER SYSTEM statement to turn on the resource manager (if it was previously off) or to turn off the resource manager or change the current plan (if it was previously on). If you specify a plan that does not exist in the data dictionary, Oracle returns an error message.

Recommended Setting

Set the RESOURCE_MANAGER_PLAN parameter to none (the default value) to disable the resource manager.

SESSION_CACHED_CURSORS

The SESSION_CACHED_CURSORS parameter specifies the number of session cursors to cache. Repeated parse calls of the same SQL (including recursive SQL) or PL/SQL statement causes the session cursor for that statement to be moved into the session cursor cache.

Subsequent parse calls will find the cursor in the cache and do not need to reopen the cursor.

Oracle uses a least recently used algorithm to remove entries in the session cursor cache to make room for new entries when needed.

The SESSION_CACHED_CURSORS parameter is used to reduce the amount of parsing with SQL statements that use host variables and with PL/SQL cursors.

If SESSION_CACHED_CURSORS is not set, it defaults to 0 and no cursors will be cached for your session. (Your cursors will still be cached in the shared pool, but your session will have to find them there.) If it is set, then when a parse request is issued, Oracle checks the library cache to see whether more than 3 parse requests have been issued for that statement.

If so, Oracle moves the session cursor associated with that statement into the session cursor cache.

Subsequent parse requests for that statement by the same session are then filled from the session cursor cache, thus avoiding even a soft parse. (Technically, a parse cannot be completely avoided; a "softer" soft parse is done that is faster and requires less CPU.)

Recommended Setting

Set the SESSION_CACHED_CURSORS parameter value to 200 or a greater value.

The value of SESSION_CACHED_CURSORS must be less than the value of OPEN_CURSORS.

After SESSION_CACHED_CURSORS is modified, Oracle needs to be restarted.

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. HPE 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.

_SORT_ELIMINATION_COST_RATIO

For certain restrictive (with good filters specified) and limited (returns few records) searches, PPM Center uses the FIRST_ROWS_N optimization mode.

If a search such as this also uses SORT on one or more fields returned by the search, Oracle uses the INDEX on the sorted columns under the FIRST_ROW_N optimization, even if other indexes on supplied filters may yield to a better execution plan for a SQL statement. This often leads to a less desirable INDEX FULL SCAN on the index on sorted column.

Recommended Setting

Set the parameter value to 5. This directs Oracle to consider an execution plan with ORDER BY sort elimination, as long as the plan is no more expensive than five times the cost of the best-known plan (that uses sort).

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 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 database schema, and then issue the following SQL statement.

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

> **Note:** You typically assign this grant during PPM installation or upgrade.

Generating Database Links (Oracle Object Migration)

PPM can use database links to communicate with other databases. Usually a database link created and associated with a particular environment in PPM 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 Deployment Management activities
- Some Deployment Management Extensions, such as the Extension for Oracle E-Business Suite, to facilitate 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 database), issue an SQL statement similar to the following in the PPM database schema.

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

For more information about database links, see:

- *Deployment Management Extension for Oracle E-Business Suite Guide*
- *Object Migrator Guide*

- *GL Migrator Guide*
- Oracle's reference document on the SQL language

Configuring the PPM Workbench to Run as a Java Applet

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

- "(Optional) Enabling SOCKS Proxy " below
- "Running PPM Workbench with HTTP(S)" on the next page
- "Running PPM Workbench with RMI(S) (Optional) " on the next page
- "Providing Users with the Java Plug-In" on page 122

(Optional) Enabling SOCKS Proxy

Using the SOCKS proxy feature in PPM 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:

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.

For example: `com.kintana.core.server.WORKBENCH_PLUGIN_VERSION=1.7.0_04`

Running PPM Workbench with HTTP(S)

Starting from 9.40, PPM Workbench communicates with PPM Server via HTTP(S) by default, the same port that is used in communication between Web browser and PPM Server.

If you find PPM Workbench communicates with PPM Server using RMI(S) port, you can do the following to enable the communication via HTTP(S):

1. Stop the PPM Server.
2. Set the parameter `ENABLE_WORKBENCH_HTTP` to `true` in `server.conf` file. By default, it is `true`.
3. Provide a value in the parameter `WORKBENCH_SERVICE_URL`. This is the address of PPM Server that PPM Workbench communicates with via HTTP(S).

   If the value of this parameter is null, the system uses the value specified in `BASE_URL` for the communication between PPM workbench and PPM Server via HTTP(S).
4. Run the `kUpdateHtml.sh` script.
5. Start the PPM Server.

> **Note:** When PPM Workbench communicates with PPM Server via HTTP(S) and PPM is run in the SSO mode, you can only open workbench in Internet Explorer using the PPM menu **Open > Administration > Open Workbench**.

Running PPM Workbench with RMI(S) (Optional)

If you set the parameter `ENABLE_WORKBENCH_HTTP` to `false`, PPM Workbench communicates with PPM Server using RMI(S) port.

To run 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.

> **Note:** HPE 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.

> **Note:** 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 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 `http://javadl.sun.com/webapps/download/GetFile/1.7.0_11-b21/windows-i586/xpiinstall.exe`.

> **Note:** Normally it is not recommended to set this parameter. The XPI is installed if you install JRE. To download and install JRE, go to http://www.oracle.com/technetwork/java/javase/downloads/index.html.
>
> However, if needed, you can go to the following address to find the specific web location for downloading the `.xpi` file for your Firefox browser:
>
> http://www.oracle.com/technetwork/java/javase/autodownload-140472.html

> **Note:** Normally it is not recommended to set this parameter. The XPI is installed if you install JRE. To download and install JRE, go to http://www.oracle.com/technetwork/java/javase/downloads/index.html.
>
> However, if needed, you can go to the following address to find the specific web location for downloading the `.xpi` file for your Firefox browser:
>
> http://www.oracle.com/technetwork/java/javase/autodownload-140472.html

For information about the Java plug-in supported for the current PPM version, see the *System Requirements and Compatibility Matrix*. For information about server configuration parameters, see "PPM Configuration Parameters" on page 401.

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

On the menu bar, select **Open > Administration > Open Workbench** to start the PPM Workbench from the PPM standard (HTML) interface,

Troubleshooting Tips

- 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.

- If the user is stuck on Verifying Installation screen or sees the message "Unable to launch the application", the user probably needs to configure the Java plug-in add-on in your Windows Internet Explorer browser to allow access to PPM Workbench.

  To configure Java plug-in add-on,

  a. In the Internet Explorer, click **Tools > Manage Add-ons**.

     The Manage Add-ons window displays.

  b. Locate the **Java Plug-in 1.7.0** add-on.

     The details for the add-on display. If the add-on is disabled, click **Enable**.

  c. Click **More Information** link in the details section.

  d. In the **You have approved this add-on to run on the following websites** box, type *.

  e. Click **Close** twice.

- When open PPM Workbench in 64-bit Internet Explorer by clicking **Open > Administration > Open Workbench**, you are prompted to install Java plug-in 1.7.0. After installing the Java Plug-in by following the screen instructions, you still receive prompt asking you to install Java plug-in 1.7.0.

  This is because Oracle provides automatic installation kit for 32-bit Java Plug-in 1.7.0 only. To install the 64-bit Java plug-in 1.7.0, go to Oracle website, manually download and install the 64-bit Java plug-in 1.7.0.

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 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.

> **Tip:** 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 73 (for example, if you plan to install an 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 "Maintaining the System" on page 264.

Chapter 5: Advanced System Configuration

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

> **Note:** For information on how to integrate the HPEPPM product Web service component with external single sign-on systems, see "Implementing User Authentication" on page 188.

This chapter contains the following topics:

Configuring an External Web Server

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

PPM Server can run with external Web servers such as Sun Java System Web Server, Microsoft IIS, Apache HTTP Server, HPE-UX Apache-based Web Server, and IBM HTTP Server (IHS). For detailed information about which Web server versions PPM 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

Because of Windows registry limitations, you cannot use just one Web server for multiple stand-alone PPM 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, HPE 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, HPE-UX Apache-based Web Server, and IBM HTTP Server (IHS).

> **Note:** For detailed information about which Web server PPMversions 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. Later, you identify this port in the PPM `server.conf` file and your `workers.properties` file.

> **Note:** 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 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, HPE-UX Apache-based Web Server, and IBM HTTP Server).

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

Proceed to the following topics:

- "Configuring the workers.properties File for a Single Server" on the next page
- "Configuring the uriworkermap.properties File on Microsoft IIS and Apache-Based Servers" on page 131

Configuring the workers.properties File for a Single Server

The "Sample File " below 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 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 165.

Configure 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 instance to which this worker connects. This is the name defined by the `KINTANA_SERVER_NAME` server configuration parameter in the `server.conf` file.

> **Note:** 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 is installed.

> **Note:** If the PPM 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
/reports/*=load_balancer
/logs/*=load_balancer
/pdf/*=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 PPM Center-Supported External Web Servers

This section provides information about how to set up the following external PPM-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 a Single-Server System" on page 77.)

3. Place the configured `workers.properties` file (see "Configuring the Workers Properties File" on page 128) 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.

> **Note:** 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.

> **Caution:** 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 159.

(Optional) Enable Cookie Logging on the Sun Java System Web Server

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.

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 7.0 Web Server on a Windows Server 2008 System

> **Note:** To enable Microsoft Internet Information Services with PPM Center version 9.40, make sure to select **Allowing double escaping** as described in step 12 of the configuration process.

The Tomcat redirector plug-in DLL, `isapi_redirect.dll`, does not work under 64-bit mode on Windows 2008 with IIS 7.0. To successfully configure IIS 7.0 on a Windows 2008 system, you must configure the plug-in to work in 32-bit mode.

> **Note:** The 64-bit `isapi_redirect.dll` file is for Windows 2003 64-bit.

To configure the Tomcat redirector plug-in DLL 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.

   > **Note:** 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) as follows:

   For a 32-bit systems:

   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\Apache Software
   Foundation\Jakarta Isapi Redirector\1.0
   ```

   For 64-bit systems:

   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Apache Software
   Foundation\Jakarta Isapi Redirector\1.0  (for 64bit systems)
   ```

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*>.

      > **Caution:** 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.

> **Note:** 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`

      Add a string named `extension_uri` and set its value to `/jakarta/isapi_redirect.dll`.

      ii. 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`

      iii. Add a string with the name `log_level` and set its value to `ERROR`. (For more verbose logging, options include `DEBUG` and `INFO`.)

      iv. 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`).

      v. 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`

      vi. Create a file named `rewrites.properties` in the <*ISAPI_REDIRECTOR_HOME*> directory

vii. 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 "Configuring the Microsoft Internet Information Services 7.5 Web Server on a Windows Server 2008 System" on page 140, 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 "Configuring the Microsoft Internet Information Services 7.5 Web Server on a Windows Server 2008 System" on the next page).

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. Check the **Allow double escaping** setting is correct by following these steps:

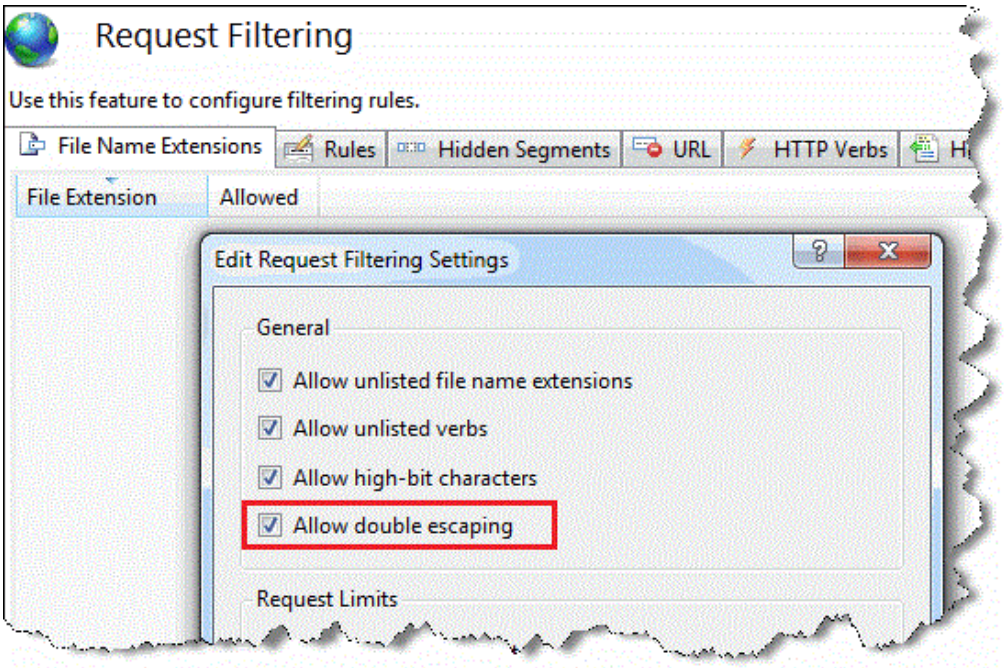
a. Open Internet Information Services (IIS) Manager.

b. Select the name of the Web site you created (or identified) in step b pf step 6 . And then, in the center panel, double-click **Request filtering**.

c. Right-click in the center panel and select **Edit Feature Settings** from the menu displayed.

The Edit Request Filtering Settings window is displayed.

d. Under **General**, select the **Allow double escaping** checkbox, as shown in the following figure.



e. Click **OK**.

13. Restart the IIS service. (Restarting the Web site is not enough. You must restart World Wide Web Publishing Service from the Services management console.)

> **Note:** 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. Before the connection is established, your browser may display the error message "HTTP Error 404 - File or directory not found. Internet Information Services (IIS)".

14. Start the PPM Server(s).

> **Caution:** If your PPM 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

> **Note:** To enable Microsoft Internet Information Services with PPM Center version 9.40, make sure to select **Allowing double escaping** as described in step 8 of the configuration process.

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.

> **Note:** 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>`.

> **Caution:** 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 of step 2.

e. Right-click the Web site you created (or identified) in step b of step 2, 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.

> **Note:** 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.**

d. Click **OK** to close the window.

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 of step 2, 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 of step 2).

   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. Check the **Allow double escaping** setting is correct by following these steps:

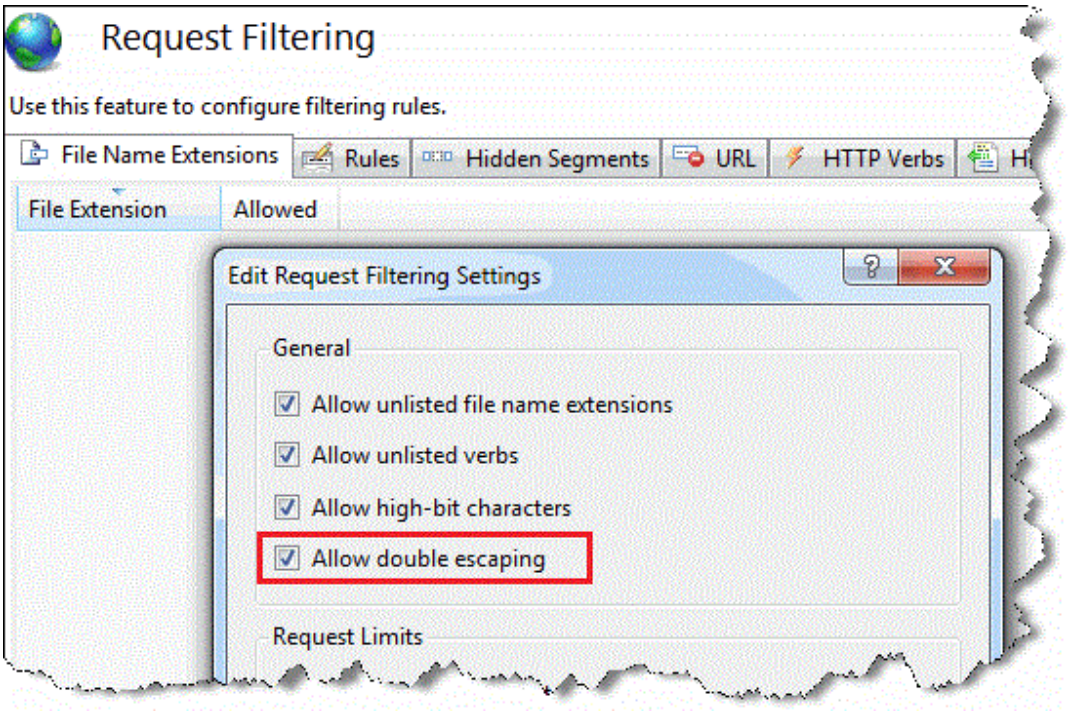
   a. Open Internet Information Services (IIS) Manager.

   b. Select the name of the Web site you created (or identified) in step b of step 2. And then, in the center panel, double-click **Request filtering**.

   c. Right-click in the center panel and select **Edit Feature Settings** from the menu displayed.

      The Edit Request Filtering Settings window is displayed.

   d. Under **General**, select the **Allow double escaping** checkbox, as shown in the following figure.



e. Click **OK**.

9. Restart the IIS service. (Restarting the Web site is not enough. You must restart World Wide Web Publishing Service from the Services management console.)

> **Note:** 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. Before the connection is established, your browser may display the error message "HTTP Error 404 - File or directory not found. Internet Information Services (IIS)".

10. Start the PPM Server(s).

> **Caution:** If your PPM 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 8.5 Web Server on a Windows Server 2012 R2 System

Configure the IIS 8.5 Web server on a Windows Server 2012 R2 system cluster

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. 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>`.

      > **Caution:** The `<ISAPI_REDIRECTOR_HOME>` directory must have run permission.

   d. Copy the following files to the `<ISAPI_REDIRECTOR_HOME>` directory you created (or selected) in step c of step 2:
      - The `workers.properties` and `uriworkermap.properties` files (located in the `<PPM_Home>\integration\webserverplugins\configuration` directory
      - The `isapi_redirect.dll` file under the `<PPM_Home>/integration/webserverplugins/iis/windows/x86-32/` directory
   e. Right-click the Web site you created (or identified) in step b of step 2, 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.

      > **Note:** 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.

2. 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.**

d. Click **OK** to close the window.

3. Configure both a `workers.properties` file and a `uriworkermap.properties` file.

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:

      In 32-bit systems:

      ```
      HKEY_LOCAL_MACHINE\SOFTWARE\Apache Software Foundation\Jakarta Isapi Redirector\1.0
      ```

      In 64-bit systems:

      ```
      HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\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 of step 2, 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.

5. 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.**

6. 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 of step 2).

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.

7. Check the **Allow double escaping** setting is correct by following these steps:

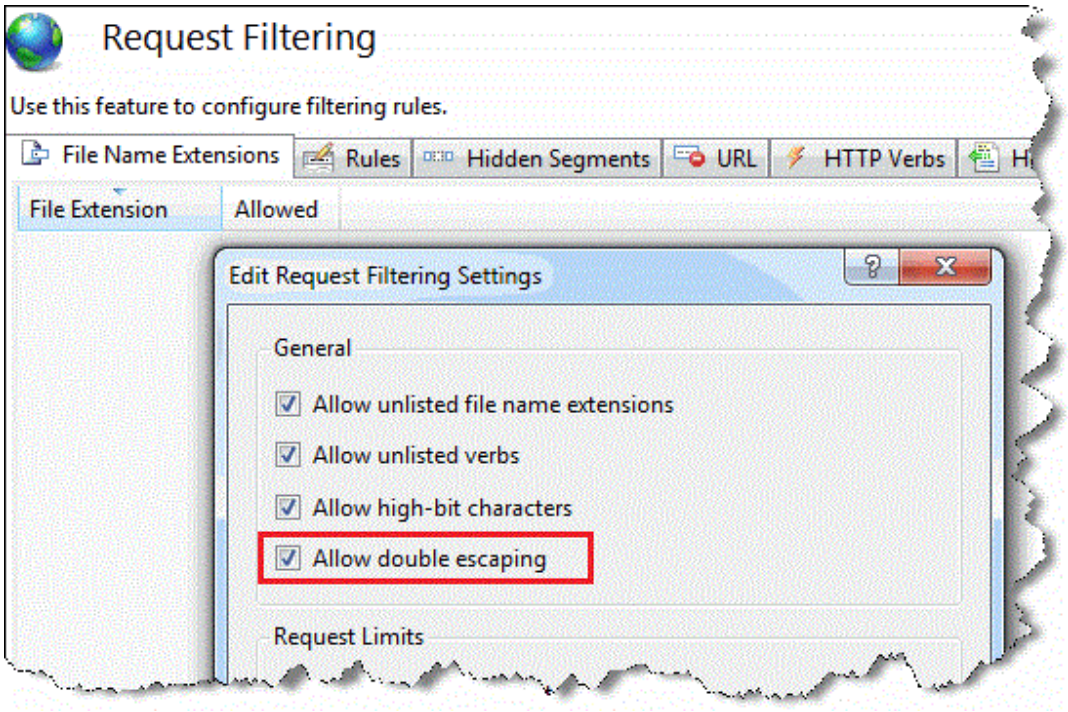
   a. Open Internet Information Services (IIS) Manager.

   b. Select the name of the Web site you created (or identified) in Step b of Step 2. And then, in the center panel, double-click **Request filtering**.

   c. Right-click in the center panel and select **Edit Feature Settings** from the menu displayed.

      The Edit Request Filtering Settings window is displayed.

   d. Under **General**, select the **Allow double escaping** checkbox, as shown in the following figure.



e. Click **OK**.

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.)

> **Note:** 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. 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).

> **Caution:** If your PPM 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.

The Tomcat redirector plug-in DLL, `isapi_redirect.dll`, does not work under 64-bit mode on Windows 2012 R2 with IIS 8.5. To successfully configure IIS 8.5 on a Windows 2012 R2 system, you must configure the plug-in to work in 32-bit mode.

Configure the Tomcat redirector plug-in DLL to run in 32-bit mode on a Windows Server 2012 R2 system (64-bit machine)

1. Stop the IIS 8.5 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.

   > **Note:** 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. Enable 32-bit applications by doing either of the following:
   - Change the default value of the application pool :
     i. Select **Application Pools** under your server.

     The **Appliaction Pools** pane is displayed in the center panel.

     ii. Right-click in the center panel and select **Set Application Pool Defaults** from the menu displayed.

     iii. Under **General**, set **Enable 32-Bit Applications** to **True**.
   - Change the application pool that your Web site runs under:
     i. Select **Application Pools** under your server.

     ii. Right-click the application pool that your Web site runs under and select **Advanced Aettings** from the menu displayed.

     iii. Under **General**, set **Enable 32-Bit Applications** to **True**.
5. Make sure that you configure the Windows registry for the Tomcat connector (isapi_redirect) as follows:

   For 32-bit systems:

   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\Apache Software
   Foundation\Jakarta Isapi Redirector\1.0
   ```

   For 64-bit systems:

   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Apache Software
   Foundation\Jakarta Isapi Redirector\1.0
   ```

Configuring an Apache-Based Web Server

This section provides the steps you use to:

- "Compile a Binary JK Module" below (Do this if, and only if, a precompiled binary does not work on your system.)
- "Configure Apache HTTP Server Version 2.2 or 2.4 Using mod_jk (IPv4 Only)" on the next page
- "Configure Apache HTTP Server Version 2.2 or 2.4 Using mod_proxy" on page 153
- "Configure IBM HTTP Server Versions 6.1 and 7.0" on page 156
- "(Optional) Generate Redirect URL Based On Server Configuration Parameter BASE_URL" on page 158

This information applies to Apache HTTP Server, HP-UX Apache-based Web Server, and IBM HTTP Server.

Compile 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.

> **Note:** 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 159.

Configure Apache HTTP Server Version 2.2 or 2.4 Using mod_jk (IPv4 Only)

> **Note:** This section applies to IPv4 only.

> **Note:** The configuration steps provided here for Apache HTTP Server version 2.4 were tested on Linux, but it should work on Windows as well.

This section provides the procedure for configuring Apache HTTP Server Version 2.2 or 2.4 using mod_jk to work with PPM.

To configure Apache 2.2 or 2.4 using mod_jk for PPM:

1. Navigate to `<PPM_Home>/integration/webserverplugins/apache-2.2/Windows/x86-32` or `<PPM_Home>/integration/webserverplugins/apache-2.4/Windows/x86-32`, and then copy the `mod_jk.so` file to the Apache module directory (usually `<Apache_Home>/modules`).

   > **Note:** The Jakarta Tomcat Connector (`mod_jk`) is used to connect Apache Web Server and PPM. The Ajp13 protocol keeps an open socket and controls the communications between PPM (its built-in Tomcat component) and Apache.

2. Instruct Apache to load the Jakarta Tomcat Connector (`mod_jk`). You can use the Apache LoadModule configuration directives in the `httpd.conf` file, which is located in `<Apache_Home>/conf` (where `<Apache_Home>` is the Apache 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
JKLogLevel ERROR
```

> **Caution:** If you plan to enable SSL on Apache, then you must also do the following:
>
> - *For Apache 2.2*
>
> Add the **JkMountCopy On** to the virtual host directive in the `httpd-ssl.conf` file.
>
> - *For Apache 2.4*
>
> Add the **JkMountCopy All** to the virtual host directive in the `httpd-ssl.conf` file.

4. Check to make sure that `include conf/extra/httpd-ssl.conf` is *not* commented out in the `httpd.conf` file.
5. Navigate to the `<PPM_Home>/integration/webserverplugins/configuration` directory, and then copy the `workers.properties` and `uriworkermap.properties` to the Apache configuration directory (usually `<Apache_Home>/conf`).
6. Configure the `workers.properties` file. (For detailed information and instructions, see "Configuring the Workers Properties File" on page 128.)
7. Configure 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 131.

> **Note:** 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.

8. Restart your Apache HTTP Server 2.2 or 2.4 and check to see whether your configuration works.

Configure Apache HTTP Server Version 2.2 or 2.4 Using mod_proxy

This section provides instructions on how to configure an external Web server on both Windows and Linux systems.

Configure Apache HTTP Server Version 2.2 or 2.4 Using mod_proxy on a Windows System

1. Download and install the Apache HTTP Server version 2.2 or 2.4 from the Apache HTTP Server Project web site (http://httpd.apache.org/download.cgi).

   > **Note:** Make sure that you use the MSI installer that includes OpenSSL.

2. Navigate to `C:\Program Files\Apache Group\Apache\conf` and open the Apache server configuration file `httpd.conf` in a text editor.

3. In the `httpd.conf` file, uncomment the following lines:

   ```
   LoadModule proxy_module modules/mod_proxy.so
   LoadModule proxy_balancer_module modules/mod_proxy_ balancer.so
   LoadModule proxy_http_module modules/mod_proxy_http.so
   LoadModule proxy_http_module modules/mod_ssl.so
   ```

4. Add the following line to the `httpd.conf` file:

   ```
   Include conf/extra/reverse_proxy.conf
   ```

5. Save and close the `httpd.conf` file.

6. Configure SSL for your Apache web server. (For information, see "Configure Apache HTTP Server Version 2.2 or 2.4 Using mod_jk (IPv4 Only)" on page 152.)

7. Use the following template to create a file named `reverse_proxy.conf`.

   > **Note:** Make sure that your `reverse_proxy.conf` correctly conforms to this format. In each directive section, list each node with IP, PORT, and the respective KINTANA_SERVER_ NAME.

   For example, for a two-node cluster:

   ```
   ProxyPass /itg balancer://itg stickysession=JSESSIONID|jsessionid
   nofailover=Off
   ProxyPassReverse /itg balancer://itg
   ProxyPass /dashboard balancer://dashboard stickysession=JSESSIONID|jsessionid
   nofailover=Off
   ProxyPassReverse /dashboard balancer://dashboard

   ProxyPassReverse/itg http://<PPM_Webserver_IP_Address>/itg
   ProxyPassReverse/dashboard http://<Proxy_Server_IP_Address>/dashboard

   <Proxy balancer://itg>
   BalancerMember http://<PPM_Server_IP_Address:HTTP_PORT>/itg route=KINTANA_
   ```

```
SERVER_NAME
BalancerMember http://<PPM_SERVER_IP_ADDRESS:HTTP_PORT>/itg route=KINTANA_
SERVER_NAME
</Proxy>

<Proxy balancer://dashboard>
BalancerMember http://<PPM_SERVER_IP_ADDRESS:HTTP_PORT>/itg route=KINTANA_
SERVER_NAME
BalancerMember http://<PPM_SERVER_IP_ADDRESS:HTTP_PORT>/itg route=KINTANA_
SERVER_NAME
</Proxy>

ProxyPreserveHost on
```

8. Restart the Apache web server.
9. Check your PPM Center `server.conf` file to make sure that all of your base URLs are of the following type:

   `http(s)://<Reverse_Proxy_Server_IP_Address>/`

Configure Apache HTTP Server Version 2.2 or 2.4 Using mod_proxy on a Linux System

1. Download and install the Apache HTTP Server Version 2.2 or 2.4 from the Apache HTTP Server Project web site (http://httpd.apache.org/download.cgi).
2. Download the UNIX source file `httpd-2.2.xx.tar.gz` or `httpd-2.4.xx.tar.gz`.
3. Run the following commands:

   ```
   $ gzip -d httpd-NN.tar.gz

   $ tar xvf httpd-NN.tar

   ./configure --prefix=/etc/httpd-NN --enable-proxy --enable-proxy-http --enable-
   proxy-balancer --enable-ssl -enable-so

   make

   make install
   ```

4. Configure SSL for your Apache web server. (For information, see "Configure Apache HTTP Server Version 2.2 or 2.4 Using mod_jk (IPv4 Only)" on page 152.)
5. Open the `httpd.conf` file in a text editor.

   To find the Apache installation directory and print the `httpd.conf` file location, run `# httpd -v`.

6. Add the following line to the `httpd.conf` file:

   ```
   Include conf/extra/reverse_proxy.conf
   ```

7. Use the following template to create a file named `reverse_proxy.conf`.

   > **Note:** Make sure that your `reverse_proxy.conf` correctly conforms to this format. In each directive section, list each node with IP, PORT, and the respective KINTANA_SERVER_NAME.
   >
   > For example, for a two-node cluster:

   ```
   ProxyPass /itg balancer://itg stickysession=JSESSIONID|jsessionid
   nofailover=Off
   ProxyPassReverse /itg balancer://itg
   ProxyPass /dashboard balancer://dashboard
   stickysession=JSESSIONID|jsessionid nofailover=Off
   ProxyPassReverse /dashboard balancer://dashboard

   ProxyPassReverse/itg http://<PPM_Webserver_IP_Address>/itg
   ProxyPassReverse/dashboard http://<Proxy_Server_IP_Address>/dashboard

   <Proxy balancer://itg>
   BalancerMember http://<PPM_Server_IP_Address:HTTP_PORT>/itg route=KINTANA_
   SERVER_NAME
   BalancerMember http://<PPM_SERVER_IP_ADDRESS:HTTP_PORT>/itg route=KINTANA_
   SERVER_NAME
   </Proxy>

   <Proxy balancer://dashboard>
   BalancerMember http://<PPM_SERVER_IP_ADDRESS:HTTP_PORT>/itg route=KINTANA_
   SERVER_NAME
   BalancerMember http://<PPM_SERVER_IP_ADDRESS:HTTP_PORT>/itg route=KINTANA_
   SERVER_NAME
   </Proxy>

   ProxyPreserveHost on
   ```

8. Restart the Apache web server.
9. Check your PPM Center `server.conf` file to make sure that all of your base URLs are of the following type:

   `http(s)://<Reverse_Proxy_Server_IP_Address>/`

Configure IBM HTTP Server Versions 6.1 and 7.0

This section provides the procedure for configuring IBM HTTP Server (IHS) to work with PPM.

To configure IHS for PPM:

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).

   > **Note:** The Jakarta Tomcat Connector (mod_jk) is used to connect IHS and PPM. The Ajp13 protocol keeps an open socket and controls the communications between PPM (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
   ```

   > **Caution:** 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 128.)

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 131

   > **Note:** 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.

(Optional) Generate Redirect URL Based On Server Configuration Parameter BASE_URL

If you enable reverse proxy, you would also need to enable generating correct redirect URL to avoid potential http redirection issues.

To enable generating redirect URL, set the server configuration parameter `SCHEME_BASED_REDIRECT_FILTER_ENABLED` in `server.conf` to `true` (default).

Setting the `SCHEME_BASED_REDIRECT_FILTER_ENABLED` parameter to `true` enables the SchemeBasedRedirectFilter. The SchemeBasedRedirectFilter generates a correct redirect URL by adding the BASE_URL value as prefix to the redirect URL, such that every redirect URL starts with scheme
(`http/https`) and the same base URL, and then sends to the correct target.

If the parameter is not present in `server.conf`, the system would treat it as a `true` condition by default.

If you do not use https or reverse proxy, you can disable the SchemeBasedRedirectFilter by setting the `SCHEME_BASED_REDIRECT_FILTER_ENABLED` parameter to `false`.

Enabling Secure Sockets Layer on an External Web Server

> **Note:** PPM does not enable SSL by default, for enabling it requires other user information. However, HPE recommends that you enable it, especially in production environment, to make sure data being transmitted is encrypted. The use of SSL protects sensitive information from the risk of eavesdropping, data tampering, or message forgery in the process of transmitting.

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).

   > **Note:** It may take several days for the certificate authority to validate the company.

> HPE does not recommend using self-signed certificates in production environments as they may negate the benefits of end-to-end security by decreasing the ability of a user to detect a man-in-the-middle (MITM) attack.

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><Web_Server_Port_or_Binding_Port>.`

   > **Note:** 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.

> **Note:** 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 "Configure IBM HTTP Server Versions 6.1 and 7.0" on page 156.

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).

PPM 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 uses the compression capabilities of both the Tomcat Web container and the compression capabilities in third-party Web servers (Microsoft Internet Information Services, Apache-based Web server, or Sun Java System Web Server).

If you deploy PPM 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 7.0 and 7.5

1. Open the IIS Manager window and navigate to the level you want to manage.
2. In **Features View**, double-click **Compression**.
3. On the **Compression** page, select the box next to **Enable dynamic content compression**.
4. Click **Apply** in the Actions pane.

For more information, see Enable HTTP Compression of Dynamic Content (IIS 7), 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.

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

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>
   ```

Creating an SSH Tunnel for RMI Server (Optional)

You can make a RMI server available from outside the firewall through RMI port forwarding. This applies to both normal RMI (`rmi://`) and secure RMI (`rmis://`).

To port forward RMI,

1. Create an SSH tunnel from the public server to the PPM Server.

   a. Run the following command on the public server:

      ```
      ssh -gvNL <Public_Server_Port>:<PPM_Server_IP_Address>:<PPM_Server_Port>
      <PPM_Username>@<PPM_Server_IP_Address>
      ```

      where,
      `<Public_Server_Port>` is the available port on the public server;
      `<PPM_Server_IP_Address>` is the IP address of your PPM Server;
      `<PPM_Server_Port>` is the available port on your PPM Server;
      `<PPM_Username>` is the user account that you want to use to access your PPM Server.

      For example,

      ```
      ssh -gvNL 8082:ppm.hp.com:50001  user1@ppm.hp.com
      ```

b. Enter the password for <*PPM_Username*> when prompted.

c. Wait until the tunnel is successfully established.

2. With the tunnel open, add the new `CLIENT_RMI_URL` parameter to the `server.conf` file if it is not present:

   `com.kintana.core.server.CLIENT_RMI_URL=rmi://<Public_Server_IP_Address>:<Public_Server_Port>/KintanaServer`

3. Save the server.conf file and run the kUpdateHtml.sh script (located in the <*PPM_Home*>`/bin` directory).

4. Restart the PPM Server.

5. You can now access PPM Workbench from PPM Center.

6. On the public server, press Ctrl+C to close the tunnel when you no longer need it.

> **Note:** This does not change the RMI URL used when opening the PPM Workbench via Java webstart.

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 a Single-Server System" on page 77.

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

1. Back up the <*PPM_Home*>`/server.conf` file.

> **Caution:**
> Because the backups you create contain sensitive information such as cryptographic keys and payload data, HPE strongly advises that you protect the backups themselves. Oracle Advanced Security provides transparent data encryption of data stored in the database, the encryption of disk-based backups of the database, and network encryption for data traveling across the network between the database and client or mid-tier applications. It also provides a suite of strong authentication services to Oracle Database.
>
> To use Enterprise User Security in Oracle Database Enterprise Edition, you must license Oracle Internet Directory (OID). If you need to use stronger authentication alternatives for enterprise user security, you must license Oracle Advanced Security and the Oracle Internet Directory (OID). For more information, see the release notes for your Oracle software

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.

   > **Note:** 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 "PPM Configuration Parameters" on page 401. For more information about the `kUpdateHtml.sh` script, see "kUpdateHtml.sh" on page 514.

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.)

> **Note:** For information about how to start the PPM Server, see "Starting and Stopping the PPM Server on a Single-Server System" on page 77. 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, 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 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.

   > **Note:** By default, Sun Java System Web Server is configured to process JSP files. Because of this, HTTP requests are not redirected to PPM.
7. If PPM does not receive requests from the Web server, try to access PPM 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 instance, you find that you cannot export PPM Dashboard pages in PDF format, do the following:

1. From the PPM standard interface, open the Administration Console.
2. 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.)

> **Note:** For information about how to open the Administration Console, see "Opening the Administration Console" on page 273.
>
> For information about how to set parameter values in the Administration Console, see "Modifying Parameters from the Administration Console" on page 284.
>
> For information about dashboard configuration parameters, see "Server Configuration Parameters Related to the PPM Dashboard" on page 476.
>
> For information about how to add a server configuration parameter to your PPM instance, see "Modify Server Configuration Parameters Not Listed in the server.conf File " on the next page.

Configuring a Server Cluster

This section provides the following information about server clustering in the PPM environment:

- Server clustering overview
- Creating a shared 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 "System Overview" on page 18, particularly "Server Cluster Configurations (Recommended)" on page 23. 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.

> **Note:** 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.

> **Caution:** 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, RMIS, and so on) that do not conflict.

Modify 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. "Using the Server Configuration Utility to Modify Server Configuration Parameters" on page 403 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 "Using the Server Configuration Utility to Modify Server Configuration Parameters" on page 403. 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.

> **Note:** 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

The table below 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 "PPM Configuration Parameters" on page 401.

Table 1. Server configuration parameters affected by clustering

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
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 uses Tomcat 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 1, successful server cluster setup requires that you define additional node-specific parameters to specify ports (Table 3) and cluster-specific (Table 2) server configuration parameters.

For the cluster-specific server configuration parameters listed in Table 2, you must set the same values for all nodes in the cluster.

Table 2. Required cluster-specific parameters

Parameter	Description
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'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. **Caution:** The value you specify for MULTICAST_CLUSTER_NAME must *not* include the text string `"http://"`.
mcast_port	Port used by PPM 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.

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 3 lists the server configuration parameters you use to specify these ports.

Table 3. Required node-specific parameter for multiple nodes on a single host

Port Parameter	Description
APP_SERVER_	Protocol listening port for the JBoss Application Server UIL2 service. For a

Table 3. Required node-specific parameter for multiple nodes on a single host, continued

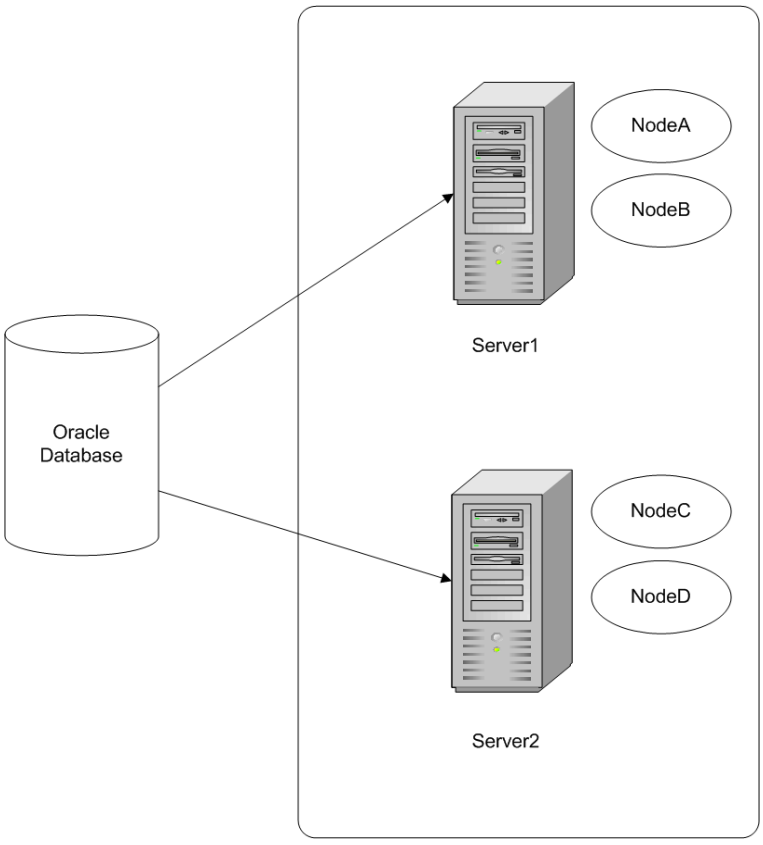
Port Parameter	Description
UIL2_BINDING_PORT	PPM Server in a server cluster, specify a port that is unique for the node in the cluster.

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

The figure below 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.

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.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_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_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_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_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 instance is deployed. For information about the Administration Console, see "Tools in the Administration Console" on page 272.

> **Caution:** 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" below
- "Preparing a Shared Folder for server.conf on a UNIX System" on the next page
- "High-Level Steps for Server Cluster Configuration" on the next page
- "External Web Server, Single Machine" on page 174
- "External Web Server, Multiple Machines" on page 176
- "Hardware Load Balancer, Multiple Machines" on page 178

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

1. Create a shared folder on a file server.
2. Attach the shared folder to each machine that is to host PPM.
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 nodes on Windows. The value of the environment variable is the location of the shared folder.

   > **Caution:** 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

1. Create a shared folder on a file server.
2. Mount the shared folder to each machine that is to host PPM.
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, 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 a Single-Server System" on page 77.
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 171
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 128.
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 1. Server configuration parameters affected by clustering" on page 168):

- 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 a Single-Server System" on page 77).
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 128.)
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
```

> **Note:** 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 2. Required cluster-specific parameters" on page 169.
7. After each @node directive, do the following:
   a. Set values for the parameters listed in "Table 1. Server configuration parameters affected by clustering" on page 168 (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 3. Required node-specific parameter for multiple nodes on a single host" on page 169.

      For your convenience, the table in "Sample Port Sets" on page 178 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.
    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 180.

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.

   > **Note:** For information about how to configure a machine for integration with an external Web server, see "Configuring an External Web Server" on page 126. 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 162.

2. Stop the PPM Server. (See "Starting and Stopping the PPM Server on a Single-Server System" on page 77.)
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.

   > **Note:** 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 128.)
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.

   > **Note:** 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 user account that has read and write permission on the shared host.
>
> `//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 2. Required cluster-specific parameters" on page 169.
9. After each `@node` directive, do the following:
   a. Set values for the parameters listed in "Table 1. Server configuration parameters affected by clustering" on page 168 (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 3. Required node-specific parameter for multiple nodes on a single host" on page 169. (For your convenience, HPE 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. HPE-supplied port sets" on page 179.)
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.

> **Note:** 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.

> **Note:** Sticky sessions are required for hardware load balancing in the PPM environment.

Sample Port Sets

The table below 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. HPE-supplied port sets

Port Name[a]	Node A	Node B	Node C	Node D	Node E
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.					

> **Note:** 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 a Single-Server System" on page 77.)

> **Caution:** If your PPM 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>
```

To stop a secondary node, run the `kStop.sh` script, as follows:

```
sh ./kStop.sh -name <PPM Server> -now -user <User_Name>
```

On Windows, there is one service (called "HPE `PPM <PPM Server>`") 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 HPE PPM Web server module detects that the node is unavailable and stops sending it

HTTP requests. When the node becomes available again, the HPE PPM Web server module detects the node and sends the requests again.

> **Note:** 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

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.

   > **Caution:** If your PPM 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>/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 27.)

> **Note:** 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`, and `MULTICAST_PORT` parameters in the `server.conf` file. These values are shared by all of the nodes in a cluster.

  > **Caution:** 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.

- Each cluster must have unique `MULTICAST_CLUSTER_NAME` setting. All nodes within this cluster share the same value for this parameter.
- 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.

  > **Note:** 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
- mcast_port

- If the MULTICAST_IP value is shared by multiple clusters within a network, then values for the 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 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, and MULTICAST_PORT settings.
- If clusters other than those related to PPM are set up, and these use the same multicast IP/port, the environment may also conflict.

Disabling Nodes from Running Background Services

HPE recommends that, if you have a server cluster configured, and there are nodes in the cluster that do not handle incoming user requests, you disable the nodes from running the PPM background service.

To disable nodes that receive no user traffic from running background services:

1. Stop the nodes.

   > **Note:** For instructions on how to start and stop PPM Servers, see "Starting and Stopping the PPM Server on a Single-Server System" on page 77.

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

      > **Note:** For information about adding parameters to nodes in a server cluster, see "@node Directive in the server.conf File " on page 166.

4. Navigate to the <*PPM_Home*>/bin directory, and then run the kUpdateHtml.sh script.

5. Start the nodes.

   Tip: Wait for a few seconds between each node startup.

Tip: To restrict services to nodes that receive no user traffic, remove the service node from the `workers.properties` file during external web server integration configuration or remove it from hard load balancer redirection list.

Detecting Multicast Routing and Configuration Issues for a Server Cluster

In PPM version 9.11 or earlier, if multicast traffic could not pass between two nodes in a server cluster, the nodes ran in isolation and no warning was issued. The only way to determine whether the nodes were communicating was to use the JGroups send and receive test utilities.

Starting from version 9.12, PPM logs clear warnings in the server logs if multicast traffic has not been detected from a node after a specified time interval has elapsed, even though the node can be reached on its JMS connection factory and the PPM_SERVER_INSTANCE table indicates that the node is running. You can configure the amount of time that must pass before PPM logs warnings by setting the `MULTICAST_WARNING_MINUTES` server configuration parameter.

Node behavior is monitored somewhat differently on the `MULTICAST_PORT` and `APP_SERVER_MULTICAST_PORT` ports. To provide system administrators with better visibility into multicast behavior, server logging was improved as follows:

- The `MULTICAST_PORT` (used to monitor the cluster) tracks incoming messages from each node individually. After a given node (X) is first heard from on the port, PPM logs the following message to the server log just once:

  ```
  Node X is reachable on MULTICAST_PORT.
  ```

  If a node is not heard from on the `MULTICAST_PORT` port, PPM logs the following message:

  ```
  No multicast traffic has been heard from node X on the MULTICAST_PORT port for
  over 3 minutes, even though the node appears to be up. Please check your
  multicast routing and/or configuration, or your firewall settings.
  ```

You can disable the multicast monitoring feature by setting the `MULTICAST_WARNING_MINUTES` server configuration parameter to zero or less. HPE strongly recommends that you not disable the feature unless HPE Software Support specifically requests that you do so.

Multicast in PPM Cluster Environment

There are two multicast channels:

Multicast Channel	Description
Cache port 46545	Cache invalidation channel. Hardcoded to port 46545 in `cache.conf`. Uses JGroups protocol.
MULTICAST_ PORT	ClusterMonitor channel. Controls background services scheduler. Uses JGroups protocol.

All channels are on the same `MULTICAST_IP`.

View Group Memberships

- Run the following command on the machine you want to show group membership:

  On Windows platform

  `netsh interface ip show joins`

  On UNIX platform

  `netstat -gn`

  To check total group membership in a cluster, run the command on all physical servers in the cluster.

  You should find that total group membership = node_count x 3.

- Set `ENABLE_JGROUPS_JMX_MONITORING` to `true` and use JMX console.

  You should find a new section near the bottom called `ppm.jgroups`. Group memberships are in the `View` attribute in `service=cacheChannel`.

View Multicast Routes

To view multicast routes, run the following command:

On Windows platform

`route print | grep 224`

On RedHat platform

`/sbin/route | grep 224`

If you see no multicast route on Windows platform, run the following:

```
route add 224.0.0.0 mask 240.0.0.0  0.0.0.0  METRIC mmm IF nnnn
```

Which NIC do Channels Bind to on a Multi Network Card (NIC) Machine

- JGroups: Controlled by `jgroups.bind_addr` Java property.

  For example, in `kStart.sh`:

  ```
  SYSTEM_PROPS="$SYSTEM_PROPS -Djgroups.bind_addr=123.123.123.123"
  ```

  Since PPM version 9.12, the NIC that JGroups has bound to is logged at level `STATUS` on startup.

  Since PPM version 9.14, JGroups automatically binds to the NIC which has IP address `SERVER_NAME`.

- MULTICAST_PORT channel: Binds to NIC that has multicast route with most preferred (that is, the lowest) metric.

Set JGroups Channels in TCP Mode

You can set the two JGroups channels in TCP mode. To switch to TCP mode,

1. Add the following to the server.conf file:

   ```
   com.kintana.core.server.MULTICAST_ENABLE_TCP=true
   ```

2. Set two ports in `server.conf` for each PPM node:
   - First channel: `PPM_NODE:MULTICAST_TCP_CACHE_PORT`
   - Second channel: `PPM_NODE:MULTICAST_TCP_INTEGRITY_PORT`

To test the multicast setup, you can use the McastSenderTest and McastReceiverTest utilities. For more details, visit this URL: http://www.jgroups.org/manual/html/ch02.html. You may also use ssmping utility from http://www.venaas.no/multicast/ssmping/.

Switching Between Stand-Alone and Server Cluster Configurations

If you upgrade a stand-alone instance of PPM, 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.

> **Note:** For information about server clustering, see "Server Cluster Configurations (Recommended)" on page 23.

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 a Single-Server System" on page 77.)
2. Remove the additional node configuration from the `server.conf` file.
3. Remove the additional node file system from `<PPM_Home>/server`.
4. Run the `kUpdateHtml.sh` script.
5. 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

> **Note:** For information about server clustering, see "Server Cluster Configurations (Recommended)" on page 23.

To switch from a stand-alone to a server cluster deployment:

1. Stop the PPM Server. (See "Starting and Stopping the PPM Server on a Single-Server System" on page 77.)
2. Make sure that the following server cluster-related parameter is in the `server.conf` file (located in the `<PPM_Home>` directory), and that it is uncommented:
   - `APP_SERVER_UIL2_BINDING_PORT`
3. Save and close the `server.conf` file.
4. Complete the PPM Server cluster environment setup as described in "Configuring a Server Cluster" on page 165.

5. Run the `kUpdateHtml.sh` script.
6. After you complete the server cluster setup, follow the steps described in "Verifying Successful Cluster Configuration" on page 180.

Chapter 6: Implementing User Authentication

This chapter contains the following topics:

Overview of Implementing User Authentication

PPM uses a framework similar to Java Authentication and Authorization Service (JAAS) to integrate with pluggable authentication schemes. Integration of PPM with CA SiteMinder and LDAP is supported. This section provides information on how to integrate PPM 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.

The sections cover the different types of user authentication methods supported for use with PPM. They provide instructions on how to:

- Integrate PPM with an LDAP directory server
- Implement Web remote single sign-on with PPM
- Implement generic single sign-on with PPM
- Implement lightweight single sign-on authentication (LW-SSO)
- Integrate PPM with SiteMinder

> **Note:** The user experience logging off of a PPM 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 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 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 database.

> **Note:** This section addresses LDAP directory server integration with a PPM. 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 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 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`. The `KINTANA_LDAP_PASSWORD` parameter in the `server.conf` file is an encrypted string enclosed with `#!#` character delimiters.

     Example

     `com.kintana.core.server.KINTANA_LDAP_PASSWORD=#!#encryptedstring#!#`

     You may set the `KINTANA_LDAP_PASSWORD` parameter in two ways:

     - Run the `kConfig.sh` script and provide the plaintext LDAP password when prompted. The script will write out the `server.conf` file with the `KINTANA_LDAP_PASSWORD` entry encrypted as above. Or,
     - If the LDAP password change is the only change you want to make to the `server.conf` file, then do the following:
       A. Run `kEncrypt.sh`, and provide the plaintext LDAP password when prompted.
       B. Paste the encrypted string output into the `server.conf` file `KINTANA_LDAP_PASSWORD` entry enclosed with the `#!!#` character delimiters as in the example above.

   - `KINTANA_LDAP_ID`. Specify the PPM 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

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.

      > **Note:** 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.

> **Note:** 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
```

> **Note:** 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 users 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.

      Note that you can import users from Org Units that do not have unique names but are of different hierarchical levels. A **Hierarchy** column is added to pages or pop-up windows that are related to Org Units to help differentiate the hierarchical levels of the Org Units you import.

      After running the report, check for duplicated user information and accuracy of each of the users' information in the PPM Workbench.
   c. For the **LDAP Import?** option, click **Yes.**

Support for Multi-Domain LDAP Import

PPM provides support for multi-domain LDAP import through the following two attributes of the `KNTA_USERS_INT` parameter:

- DISTINGUISHED_NAME

  Maps to a unique and fixed field of the LDAP server. For example, `DISTINGUISHED_NAME =`

distinguishedName.

> **Note:** By default, the DISTINGUISHED_NAME attribute maps to distinguishedName of the LDAP server. If distinguishedName of the LDAP server is changeable, make sure you map DISTINGUISHED_NAME to another field that is unique and fixed on the LDAP server.

Required if using SSO, LDAP, or NTLM as the user authentication mode.

- LDAP_USERNAME

  Maps to a user's Logon ID, which is used by the user to log on to the NTLM or SSO server. For example, LDAP_USERNAME = sAMAccountName.

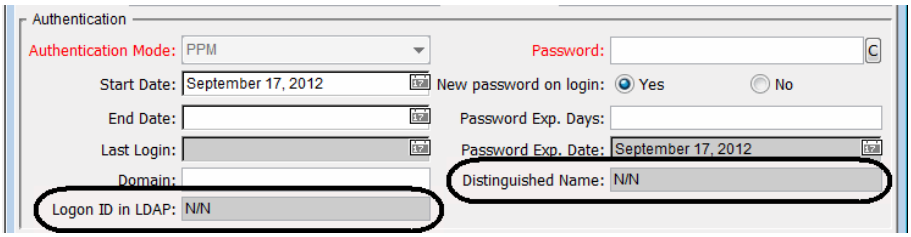
  Required if using SSO, LDAP, or NTLM as the user authentication mode.

The KNTA_USERS_INT parameter exists in the following four configuration files under the <*PPM Server*>/integration/ldap directory:

- LdapAttribute.conf
- LdapAttribute_AD.conf
- LdapAttribute_NDS.conf
- LdapAttribute_Netscape.conf

If you do not find the DISTINGUISHED_NAME and LDAP_USERNAME columns, make sure to add them and their mapping values into each of the four configuration files manually.

After you run the Import Users report, the **Distinguished Name** and **Logon ID in LDAP** fields are added to the **User Information** tab of the User window in the User Workbench.



Running the Import Users report populates these two fields with appropriate values.

These two fields are not editable. If the fields are empty or display incorrect values, contact HPE Software Support.

Authenticating Against Multiple LDAP Domains

PPM 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 uses the value set for LDAP_URL.

If the URLs provided for LDAP_URL_FULL do not have a DN value, PPM uses the value set for LDAP_BASE_DN.

> **Note:** 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 487.

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

This section provides information on how to implement Web remote single sign-on with PPM. 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 as follows:

1. A user logs in to a Windows desktop.
2. The user accesses PPM 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 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 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 162.
- To ensure that you have the required access rights, make sure that the system username you use to log on to PPM is same as the account username for the active directory.
- Clients must use Microsoft Internet Explorer to log on to PPM. 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

To configure Web remote single sign-on with PPM:

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 162.

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
      ```

      > **Note:** For information on how to edit the `server.conf` file, see "PPM Configuration Parameters" on page 401.

   c. Save and then close the `server.conf` file.
   d. Run the `kUpdateHtml.sh` script.

      > **Note:** For information about the `kUpdateHtml.sh` script, see "kUpdateHtml.sh" on page 514.

   e. Disable Tomcat from authenticating the user. Otherwise, you will get the "`No Access`" error message when trying to access PPM.
      i. Open the following file in an editor (for example, notepad, or VI editor):
      `<PPM_Home>/conf/jboss/server.xsl`
      ii. Append **`tomcatAuthentication="false"`** to the end of the Connector protocol line.

      For example,

      ```
      <Connector enableLookups="false" redirectPort="8443" debug="0"
      protocol="AJP/1.3" tomcatAuthentication="false">
      ```

      iii. In PPM Workbench, create the same Window user account in PPM Workbench and select **NTLM** as its authentication mode. Give proper access grants.
      iv. In the `server.conf` file, for `authentication_mode`, add `NTLM`.
      v. Restart the PPM Server.

3. On the IIS external Web server, do the following:
   a. From IIS Microsoft Management Console, select the default Web site.
   b. In the **Home** pane for the default Web site, scroll to the **Security** section, and then double-click **Authentication**.

c. In the Authentication pane, right-click **Anonymous Authentication** and select **Disable** from the context menu.

d. Stop, and then restart the IIS Windows service.

4. Stop and 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 199.

Implementing Generic Single Sign-On with PPM

This section provides information on how to configure PPM 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 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 user, he is granted access to the PPM standard interface and PPM Dashboard.

Requirements for Implementing Generic Single Sign-On

To implement generic single sign-on with PPM, your PPM 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

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 user is also a valid single sign-on user and can be authenticated.
2. External Web server:
   a. Integrate PPM 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 162.
   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.

      > **Note:** 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" on the next page.

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`.

      > **Note:** For information on how to edit the `server.conf` file, see "PPM Configuration Parameters" on page 401.

   b. Run the `kUpdateHtml.sh` script.
   c. Restart the PPM Server.

      > **Note:** For details on how to stop and start the PPM Server, see "Starting and Stopping the PPM Server on a Single-Server System" on page 77.

   d. Log on to PPM.
   e. Check the timestamp on the PPM Server.
   f. Navigate to the `<PPM_Home>/server/<PPM Server>/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.

> **Tip:** After you check for problems and error messages, you can remove the debugging code you added to the `logging.conf` file in step 2.

The session timeout property does not work in your generic single sign-on implementation. If you set the server configuration parameter `ENABLE_GENERIC_ SSO_TIMEOUT` to `true`, you enable the session timeout property. PPM redirects you to the sign-in page when it times out and forwards you to the Dashboard page after you click **Sign-in**.

> **Note:** In MLU environment, a window pops up for you to select the session language before you click **Sign-in**.

If you set the parameter to `false`, the session timeout property does not work in your generic SSO implementation.

Implementing Lightweight Single Sign-On Authentication (LW-SSO)

You can configure PPM to use lightweight single sign-on. Lightweight single sign-on, or LW-SSO, is an access control method that enables users to log on just once to gain access to the resources of multiple HPE software systems. The applications inside the configured group of HPE software systems trust the authentication, and there is no need for further authentication when switching from one application to another.

> **Note:** For Information about the system requirements for LW-SSO, see the *System Requirements and Compatibility Matrix* document. .

Configuration Requirements for LW-SSO Support

Applications that are part of the software group configured for LW-SSO must meet the following requirements:

- Each application must have a token expiration configured (see "LW_SSO_EXPIRATION_ PERIOD" on page 207). The expiration value must be at least as high as that of the application session expiration value.

  > **Note:** HPE recommends that you set the value to 60 (minutes). For an application that does

> not require a high level of security, you can configure a value of 300 minutes.

- All applications that participate in the LW-SSO integration *must* use the same GMT time.
- If applications that participate in the LW-SSO integration are required to integrate with applications in different DNS domains, then multi-domain functionality requires that trusted hosts settings (or the protectedDomains settings) are configured for each. In addition, you must add the correct domain in the `lwsso` element of the configuration for each participating application.
- To receive information sent as **SecurityToken for URL** from other applications, the host application must have the correct domain specified in the `lwsso` element of the configuration.

LW-SSO Security Warnings

The following security warnings apply to the LW-SSO configuration:

- Confidential initString parameter in LW-SSO

  LW-SSO uses Symmetric Encryption to validate and create a LW-SSO token. The `initString` parameter within the configuration is used to initialize the secret key. An application creates a token, and each application that uses the same `initString` parameter validates the token.

  - You cannot use LW-SSO without setting the `initString` parameter.
  - The `initString` parameter is confidential information and must be treated as such in terms of publishing, transporting, and persistency.
  - The `initString` is to be shared only between applications that are integrated with each other using LW-SSO.
  - The `initString` parameter value must be at least12 characters long.
- Enable LW-SSO only if it is specifically required

  Otherwise, leave it disabled.
- Level of authentication security

  The application that uses the weakest authentication framework and issues a LW-SSO token that is trusted by other integrated applications determines the level of authentication security for all the applications. HPE recommends that only applications using strong and secure authentication frameworks issue an LW-SSO token.
- Symmetric encryption implications

LW-SSO uses symmetric cryptography to issue and validate LW-SSO tokens. Therefore, any application that uses LW-SSO can issue a token to be trusted by all other applications that share the same `initString` parameter value. This potential risk is relevant if an application that shares the `initString` value either resides in, or is accessible from, an untrusted location.

- User mapping (Synchronization)

  The LW-SSO framework does not ensure user mapping between the integrated applications. Therefore, the integrated application must monitor user mapping. HPE recommends that the same user registry (as LDAP/AD) be shared among all integrated applications.

  Failure to map users may cause security breaches and negative application behavior. For example, the same user name may be assigned to different real users in different applications.

  In addition, in cases where a user logs onto an application (AppA), and then accesses a second application (AppB) that uses container or application authentication, the failure to map the user forces the user to manually log on to AppB and enter a user name. If the user enters a different user name than was used to log on to AppA, the following can occur: If the user subsequently accesses a third application (AppC) from AppA or AppB, then they will access it using the user names that were used to log on to AppA or AppB, respectively.

- Identity Manager

  Used for authentication purposes, all unprotected resources in the Identity Manager must be configured as nonsecureURLs settings in the LW-SSO configuration file.

- LW-SSO Demo mode

  LW-SSO Demo mode restrictions are as follows:

  - Use only for demonstration purposes
  - Use only in unsecured networks
  - Must *not* be used in production. Any combination on the Demo mode with the production mode must not be used.

LW-SSO Known Issues

The following issues apply to LW-SSO:

- Security context

  The LW-SSO security context supports only one attribute value per attribute name. Therefore, if the SAML2 token sends more than one value for the same attribute name, the LW-SSO framework accepts only one value. Similarly, if the IdM token is configured to send more than one value for the same attribute name, the LW-SSO framework accepts only one value.

LW-SSO Limitations

The following limitations apply to LW-SSO authentication:

- Client access to the application

  If a domain is defined in the LW-SSO configuration:

  - The application's clients must access the application with a fully qualified domain name (FQDN) in the login URL. For example, `http://myserver.companydomain.com/WebApp`
  - LW-SSO cannot support URLs with an IP address. For example, `http://192.168.12.13/WebApp`
  - LW-SSO cannot support URLs without a domain. For example, `http://myserver/WebApp`

    If a domain is not defined in the LW-SSO configuration. If a domain is not defined in the LW-SSO configuration, the client can access the application without a FQDN in the logon URL. Note that, in this case, a LW-SSO session cookie is created specifically for a single machine without any domain information, and, therefore, is not delegated by the browser to another, and does not pass to other computers located in the same DNS domain. This means that SSO will not work in the same domain.

- LW-SSO framework integration

  Applications can leverage and use LW-SSO capabilities only if you integrate them within the LW-SSO framework in advance.

- Multi domain support
  - Multi domain functionality is based on the HTTP referer. Therefore, LW-SSO supports links from one application to another and does not support typing a URL into a browser window, except when both applications are in the same domain.
  - The first cross-domain link that uses HTTP POST is not supported. Multi domain functionality does not support the first HTTP POST request to a second application (only the HTTP GET request is supported). For example, if your application has an HTTP link to a second application, an HTTP GET request is supported, but an HTTP FORM request is not supported. All requests after the first can be either HTTP POST or HTTP GET.
  - **LW-SSO Token size.** The amount of information that LW-SSO can transfer from one application in one domain to another application in another domain is limited to 15

Groups/Roles/Attributes. (Each item may be an average of 15 characters long.)

- **Linking from protected (HTTPS) to unprotected (HTTP) pages in a multi domain scenario.** Multi domain functionality does not work when linking from a protected (HTTPS) to an unprotected (HTTP) page. This is a browser limitation where the referer header is not sent when linking from a protected to a non-protected resource.
- Third-party cookie behavior in Internet Explorer

  Microsoft Internet Explorer 6 contains a module that supports the "Platform for Privacy Preferences (P3P) Project", which means that cookies coming from a third party domain are blocked by default in the Internet security zone. Internet Explorer also treats session cookies as third-party cookies. These are therefore blocked, which causes LW-SSO to stop working.

  To solve this issue and make sure that cookies are accepted, add the launched application (or a DNS domain subset as `*.<mydomain>.com`) to the Intranet/Trusted zone on your computer. (In Microsoft Internet Explorer, select **Menu > Tools > Internet Options > Security > Local Intranet > Sites > Advanced.**)

  > **Note:** The LW-SSO session cookie is only one of the cookies used by third-party applications that are blocked.

- Multi domain functionality is based on the HTTP referer. Therefore, LW-SSO supports links from one application to another and does not support typing a URL into a browser window, except when both applications are in the same domain.
- The first cross-domain link that uses HTTP POST is not supported. Multi domain functionality does not support the first HTTP POST request to a second application (only the HTTP GET request is supported). For example, if your application has an HTTP link to a second application, an HTTP GET request is supported, but an HTTP FORM request is not supported. All requests after the first can be either HTTP POST or HTTP GET.
- **LW-SSO Token size.** The amount of information that LW-SSO can transfer from one application in one domain to another application in another domain is limited to 15 Groups/Roles/Attributes. (Each item may be an average of 15 characters long.)
- **Linking from protected (HTTPS) to unprotected (HTTP) pages in a multi domain scenario.** Multi domain functionality does not work when linking from a protected (HTTPS) to an unprotected (HTTP) page. This is a browser limitation where the referer header is not sent when linking from a protected to a non-protected resource.
- Third-party cookie behavior in Internet Explorer

Microsoft Internet Explorer 6 contains a module that supports the "Platform for Privacy Preferences (P3P) Project", which means that cookies coming from a third party domain are blocked by default in the Internet security zone. Internet Explorer also treats session cookies as third-party cookies. These are therefore blocked, which causes LW-SSO to stop working.

To solve this issue and make sure that cookies are accepted, add the launched application (or a DNS domain subset as `*.<mydomain>.com`) to the Intranet/Trusted zone on your computer. (In Microsoft Internet Explorer, select **Menu > Tools > Internet Options > Security > Local Intranet > Sites > Advanced.**)

> **Note:** The LW-SSO session cookie is only one of the cookies used by third-party applications that are blocked.

- SAML2 token
  - Logout functionality is not supported if the SAML2 token is used. Therefore, if the SAML2 token is used to access a second application, then a user who logs out of the first application is not logged out of the second application.
  - The SAML2 token's expiration is not reflected in the application's session management. Therefore, if the SAML2 token is used to access a second application, then each application's session management is handled independently.
- JAAS Realm

  The JAAS Realm in Tomcat is not supported.
- Using spaces in Tomcat directories

  Using spaces in Tomcat directories is not supported. You cannot use LW-SSO if the Tomcat installation path includes spaces (for example, `\Program Files\`) and the LW-SSO configuration file resides in the `common\classes` Tomcat folder.
- Load balancer configuration

  A load balancer deployed with LW-SSO must be configured to use sticky sessions.
- Demo mode

  In Demo mode, LW-SSO supports links from one application to another but, because there is no HTTP referer header, does not support typing a URL into a browser window.

Configuring PPM for LW-SSO

> **Caution:** Before you start to configure PPM for LW-SSO, make sure that you first read "LW-SSO Security Warnings" on page 201.

1. Add the parameters described in the following table to the `server.conf` file and assign values to each.

Parameter	Description
ENABLE_LW_SSO_UI	Use to enable the LW-SSO user interface. Set this parameter to `true`.
LW_SSO_DOMAIN	Use to specify the LW-SSO domain. **Example:** `xyz.com`
LW_SSO_INIT_STRING	Use to specify the value of the `initString` parameter For information about the `initString` parameter, see "LW-SSO Security Warnings" on page 201.
LW_SSO_EXPIRATION_ PERIOD	The token for validating user logon has an expiration value that determines an application's session validity. Use this parameter to specify the LW-SSO token expiration period in minutes. Configure a token expiration for each application that uses LW-SSO. HPE recommends that you set the value to 60 (minutes). **Note:** The expiration value must be at least as high as that of the application session expiration value.
LW_SSO_TRUSTED_ DOMAIN	Use to specify one or more LW-SSO trusted domains. To separate multiple domains, use semicolons (;). **Example:** `xyz.come;abc.net`
LW_SSO_CLEAR_ COOKIE	Use to specify that PPM must clear the LW-SSO token when a user logs out of PPM. **Note:** For security purposes, HPE recommends that you always keep this parameter set to `true`.
ENABLE_LW_SSO_ WEB_ SERVICE	For integration of PPM Tasks with Service Manager RFCs only. To specify that PPM always uses the current user to call Service

Parameter	Description
	Manager Web service, set to `true`. Default: `false`

2. Run `kUpdateHtml.sh`.
3. Stop, and then restart the PPM Servers.

Integrating PPM with CA SiteMinder

You can configure PPM 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" below
- "Single Sign-on Mode" on page 210
- "Requirements for Integrating with SiteMinder" on page 212
- "Overview of Integrating PPM with SiteMinder" on page 212
- "Configuring PPM for Integration with SiteMinder" on page 212
- "Configuring PPM Users" on page 216
- "Configuring SiteMinder for Integration with PPM" on page 217

Mixed Mode

In the mixed mode configuration, PPM users can continue to log on using the PPM 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 to use both SiteMinder and its own authentication simultaneously. In this case, the authentication mode to be used must be specified in each PPM user account.

Integration Architecture for Mixed Mode

In a mixed mode configuration, users log on to PPM, 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 1 shows a system diagram of the SiteMinder integration in mixed mode. Figure 2 shows the integration architecture for mixed mode with the optional external Web server.

Figure 1. SiteMinder integration architecture for mixed mode

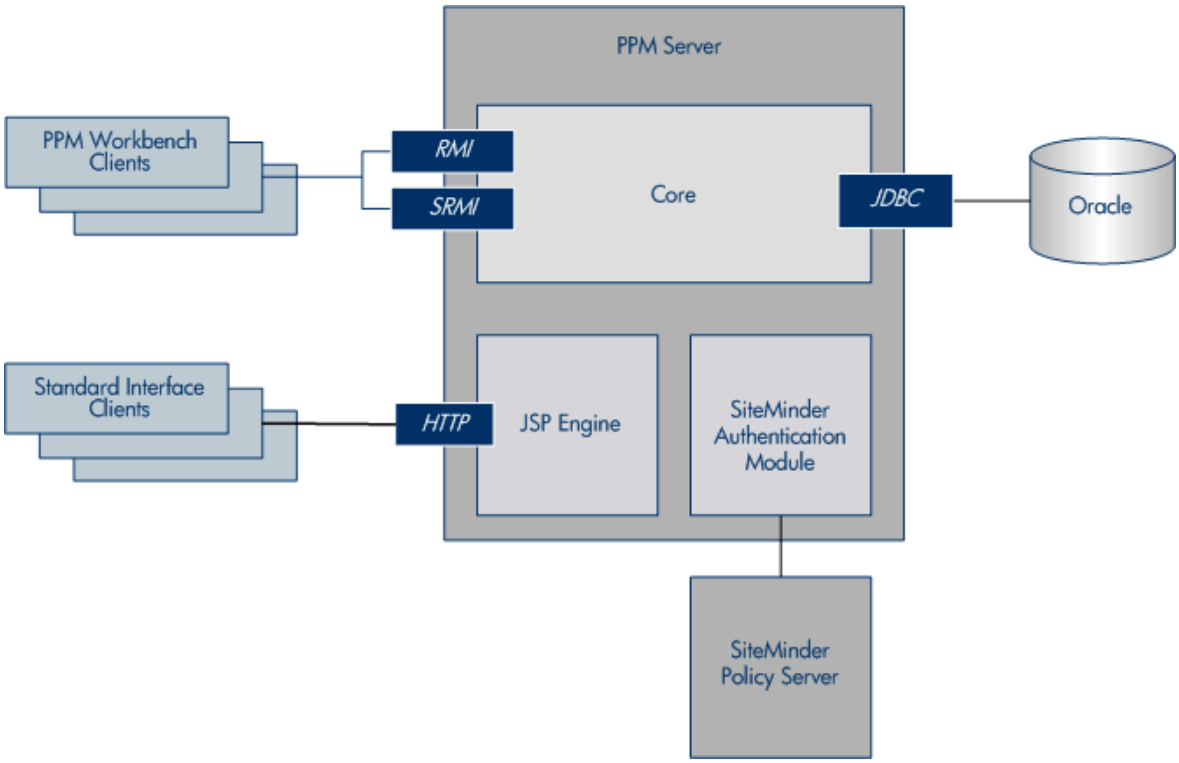
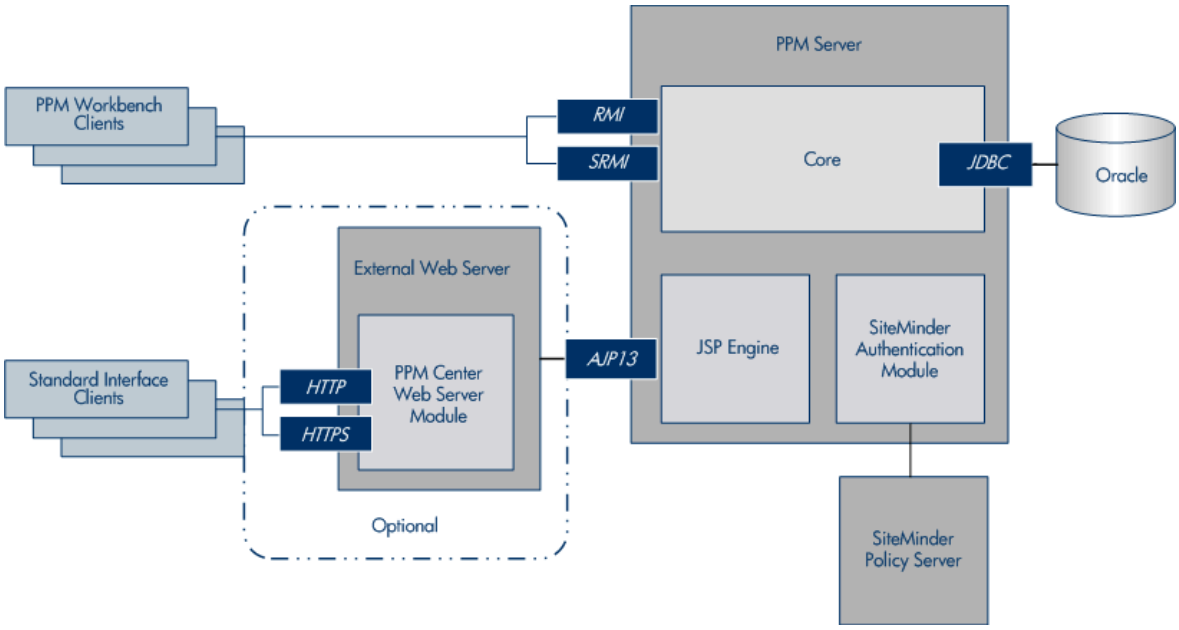


Figure 2. SiteMinder integration architecture for mixed mode with optional external Web server



Single Sign-on Mode

In the SSO mode configuration, Web requests are authenticated before being passed to PPM, bypassing the PPM logon page. To enable SSO mode, the SiteMinder Web Agent must be plugged into any third-party Web server software that PPM 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.

Note that you cannot use SiteMinder to manage PPM application-level authorization for controlling access to various screens and functions. Application-level authorization is controlled by the PPM security model using security groups, access grants, product licensing, and so on. Therefore, user accounts must exist in both PPM and the SiteMinder Policy Server, but PPM does not have to maintain the associated passwords.

Integration Architecture for SSO Mode

Single sign-on configuration requires that PPM be integrated with an external Web server that has both the SiteMinder Web Agent and PPM Web Server Module installed. (The PPM internal Web server does not support SiteMinder SSO because there is no compatible Web agent or a suitable API to create one.)

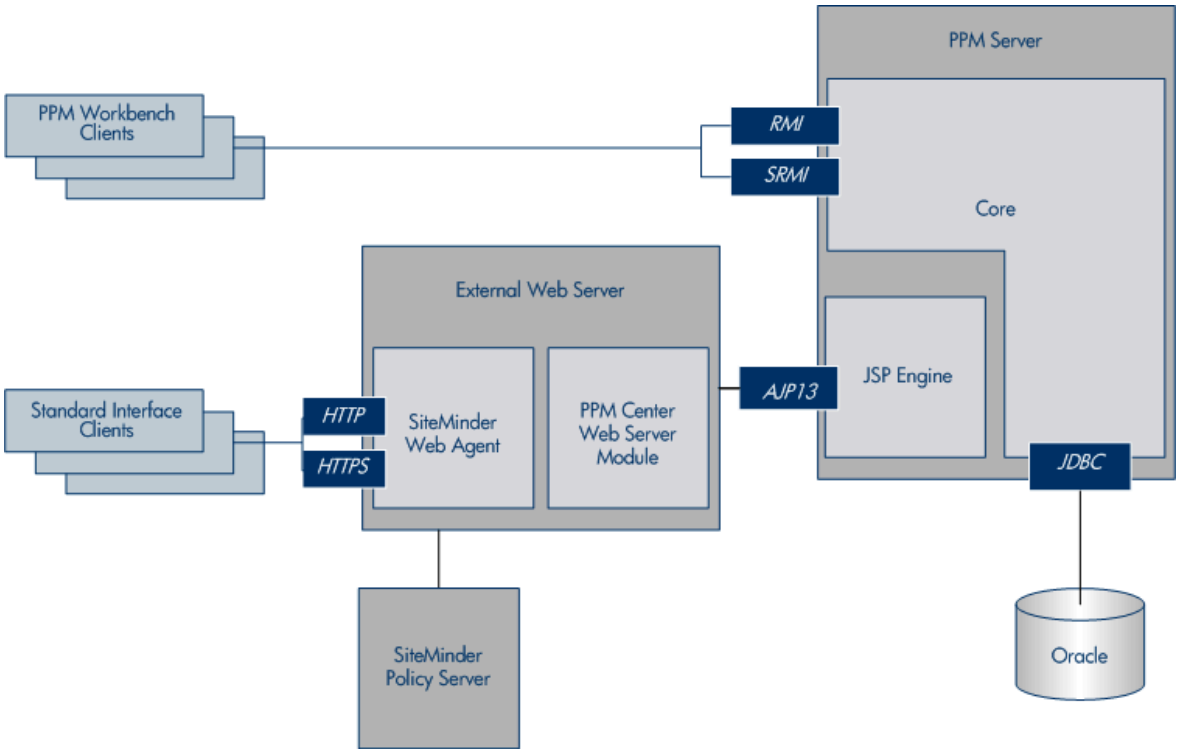
The SiteMinder Web Agent is the single access point for all Web clients. The SiteMinder Web Agent intercepts all incoming requests and ensures that they are authenticated before passing them to the PPM Web Server module. The requests then proceed to the PPM Server.

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.

> **Caution:** PPM Workbench does not support SSO mode if you start it from the menu bar (select **Administration > Open Workbench on Desktop**). However, if PPM is launched as an application, it uses SiteMinder to authenticate. See "Configuring the PPM Workbench to Run as a Java Applet" on page 120.

The figure below shows a system diagram of SiteMinder integration in SSO mode.

SiteMinder integration architecture for SSO mode



Requirements for Integrating with SiteMinder

The requirements for integrating PPM with SiteMinder are as follows:

- An External Web server (required for SSO mode, optional for mixed mode)
- PPM Web Server Module
- CA SiteMinder version 6.0 (for both SSO and mixed modes) is installed and functioning correctly

  > **Note:**
  > - For information on how to install SiteMinder, see the product documentation.
  > - Support for CA SiteMinder version 12.0 is available since PPM version 9.22.

- SiteMinder Java Agent API is installed (for mixed mode only)

Overview of Integrating PPM with SiteMinder

PPM integration with SiteMinder involves the following tasks:

1. "Configuring PPM for Integration with SiteMinder" below
2. "Configuring SiteMinder for Integration with PPM" on page 217.

   > **Note:** The configuration of SiteMinder for integration with PPM must be performed by a SiteMinder administrator.

Configuring PPM for Integration with SiteMinder

To configure PPM to integrate with SiteMinder:

1. Verify that your PPM 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>\deploy\itg.war\WEB-INF\lib` directory.
- On a UNIX system, copy the `smjavaagentapi.jar` file to the `<PPM_Home>/server/<PPM Server>/deploy/itg.war/WEB-INF/lib` directory.

> **Note:** 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. (Mixed mode only) 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.`

> **Note:** Regardless of which directory you place the DLL files in, check to 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.

> **Note:**
>
> - 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.
> - SiteMinder native dll files are available in both 32-bit and 64-bit versions. PPM Center requires that the version of SiteMinder native dll files be consistent with the version of JDK software installed on PPM Center, otherwise PPM Center may fail loading these local native code.
>
>   For example, if you use 32-bit JDK software, make sure you use 32-bit version of SiteMinder native dll files as well.

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

> **Caution:** 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.

- 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

> **Caution:** 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. Actions for mixed mode authentication only:
   a. To enable selection of either SiteMinder or PPM authentication for PPM 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.

      > **Tip:** You may want to set a few user accounts to use the PPM authentication mode to enable access to PPM in the event that the SiteMinder Policy Server is unavailable.

   a. To enable selection of either SiteMinder or PPM authentication for PPM 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.

      > **Tip:** You may want to set a few user accounts to use the PPM authentication mode to enable access to PPM in the event that the SiteMinder Policy Server is unavailable.

6. Actions for SSO mode only:
   a. To enable only SiteMinder authentication for PPM 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
```

> **Note:** When both the SiteMinder Web Agent and PPM Web server module are installed on the external Web server, the SiteMinder Web Agent always takes precedence for requests in the form of `/itg/*`.

a. To enable only SiteMinder authentication for PPM 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
```

> **Note:** When both the SiteMinder Web Agent and PPM 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 Users

To configure PPM users to authenticate using SiteMinder, complete the following steps:

1. Make sure that the usernames for PPM users match those used by SiteMinder.
2. Make sure that PPM users are set up to use SiteMinder authentication.

> **Caution:** 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. If this occurs, revert to the `server.conf` file you created in step 4, 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

Before you configure SiteMinder for use with PPM, make sure that the Policy Server is working correctly and that the User Directory to be used for PPM authentication is correctly configured. The SiteMinder Test Tool is useful for verifying that the installation is functioning correctly.

Configuring SiteMinder for PPM 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, perform the following steps.

> **Caution:** 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.`

   > **Note:** PPM uses the `LogoffUri` property to log off users correctly when they log off of the PPM standard interface.

6. Create a realm for PPM 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, PUT, and DELETE 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, DELETE.**
- 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 Web requests.

   > **Note:** 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 instance is accessed.

   - For Apache, use the following format.

     `http://<Server_Domain>:<Port>/SmMakeCookie.ccc`

It is important to understand that PPM reads the information that SiteMinder automatically injects into the HTTP Request header.

PPM 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.

> **Note:** For configuration details for these and other SiteMinder parameters, see the SiteMinder documentation.

Applying FIPS 140-2 Compliant Encryption Algorithm for PPM

PPM applied the enhanced encryption algorithm to comply with FIPS 140-2 (Federal Information Progressing Standards 140-2) in the following cases:

- Logging on to PPM with Oracle database authentication
- Creating user
- Editing user profile
- Configuring PPM database
- Logging on to PPM with LDAP authentication
- Importing LDAP users

To apply the FIPS 140-2 compliant encryption algorithm,

1. Stop the PPM Server.
2. Unzip the `fs_home.jar` file located in the `<PPM_Home>/deploy/922/SP2` directory.
3. Copy the following three `.jar` files from the `<fs_home>/utilities/fips` directory to the `<JAVA_Home>/jre/lib/ext` directory:
   - `cryptojce.jar`
   - `cryptojcommon.jar`
   - `jcmFIPS.jar`
4. Edit the `java.security` file located in the `<JAVA_Home>/jre/lib/security` directory:
   a. Add the following before the existing security providers:

      ```
      security.provider.1=com.rsa.jsafe.provider.JsafeJCE
      ```

   b. Change the sequence numbers of the providers to make sure that the numbers start with 1, followed by 2, 3, 4, and so on.
   c. Add the following two lines after the security provider list:
      - `com.rsa.cryptoj.fips140initialmode=FIPS140_MODE`
      - `com.rsa.crypto.default.random=ECDRBG`

d. Comment out the line `securerandom.source=file:/dev:/urandom` by adding a number sign # before it.

a. Add the following before the existing security providers:

`security.provider.1=com.rsa.jsafe.provider.JsafeJCE`

b. Change the sequence numbers of the providers to make sure that the numbers start with 1, followed by 2, 3, 4, and so on.

c. Add the following two lines after the security provider list:

- `com.rsa.cryptoj.fips140initialmode=FIPS140_MODE`
- `com.rsa.crypto.default.random=ECDRBG`

d. Comment out the line `securerandom.source=file:/dev:/urandom` by adding a number sign # before it.

5. Run the `ppm_fips_security_extension.sql` script located in the *PPM_Home*`/bin/db` directory.

> **Note:** Back up the following DB tables before running this script:
>
> - KNTA_USERS
> - KNTA_PASSWORD_CHANGES
> - KNTA_USERS_INT

6. Run the `sh ./kFIPSMigrate.sh` script located in the *PPM_Home*`/bin` directory.

> **Note:**
>
> - As an administrator, you should have the execution privilege to run this script.
> - When running this script, you are required to enter the start user id and the end user id to decide how much data would be processed in a batch. You can get the user ids from the KNTA_USERS table.

7. Run the `sh ./kFIPSEncrypt.sh` script located in the *PPM_Home*`/bin` directory to get the encrypted values for the passwords you set for DB and LDAP.

> **Note:**
>
> - As an administrator, you should have the execution privilege to run this script.
> - If your system is not integrated with LDAP, you do not need to run the script for the encrypted value of the LDAP password.

8. Configure the following three parameters in the `server.conf` file from the *PPM_Home* directory.

- Set the `com.kintana.core.server.FIPS_ENABLE` parameter to `true` to enable the new encryption algorithm
- Set the `com.kintana.core.server.DB_PASSWORD` parameter to the encrypted value you get in Step 7 to reset the DB password
- Set the `com.kintana.core.server.LDAP_PASSWORD` parameter to the encrypted value you get in Step 7 to reset the LDAP password

> **Note:**
>
> - You may have to modify these parameter values directly in the `server.conf` file. HPE recommends that you do not run `ppm_config.exe` (on Windows) or `kConfig.sh` (on Unix) to modify these parameters.
> - If your system is not integrated with LDAP, you do not need to reset the LDAP password.

9. Run `sh ./kUpdateHtml.sh` script located in the *PPM_Home*`/bin` directory to apply your changes on the three parameters.
10. Start the PPM Server.
11. (Optional) If your system is integrated with LDAP, and you want to import data from LDAP and set default password for the Import Users report or the Run PPM Organization Unit Interface report, you need to add an additional command for either of the reports.

    To do so,

    a. Log on to PPM.
    b. From the menu bar, select **Open > Administration > Open Workbench.**

       The PPM Workbench opens.
    c. From the shortcut bar, select **Configuration > Report Types**.

       The Report Type Workbench opens.
    d. Click **List**, and then select the desired report type.
    e. Open the report type either by double-clicking it or clicking **Open**.

       The Report Type: <*Report*> window opens.
    f. Click **New Cmd** under the **Commands** tab.

       The New Command window opens.
    g. Name the new command as you want.
    h. Type the following in the Steps field:

```
ksc_run_java com.kintana.core.server.tools.FIPSPasswordInterfaceTable "
[TEMP_GROUP_ID]"
```

> **Note:** TEMP_GROUP_ID is the name for the temp token by default. If you have changed the token name, replace TEMP_GROUP_ID with the name you used for the token.

i. Click **Add**.

   You are back to the Report Type: <*Report*> window.

j. Adjust the sequence of the added command by using the up or down button, making sure that the added command is under the Encrypt Password command.

k. Click **OK**.

Chapter 7: Improving System Performance

This chapter provides information about how to identify and correct performance problems on your PPM system, as well as what you can do to improve system performance. For more information on improving performance, see the *Deployment Best Practices for PPM Operational Reporting*.

This chapter contains the following topics:

Identifying Performance Problems

This section 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 instance to maximize performance, see the *Deployment Best Practices for PPM Operational Reporting*.

Isolating Performance Problems

The section titled "Configuring or Reconfiguring the Database" on page 108 and "PPM Configuration Parameters" on page 401 contain information on the initial settings that HPE 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 7-1, Figure 7-2, and Figure 7-3 to isolate performance problems and determine how to fix them.

Figure 7-1. Identifying and addressing system performance problems

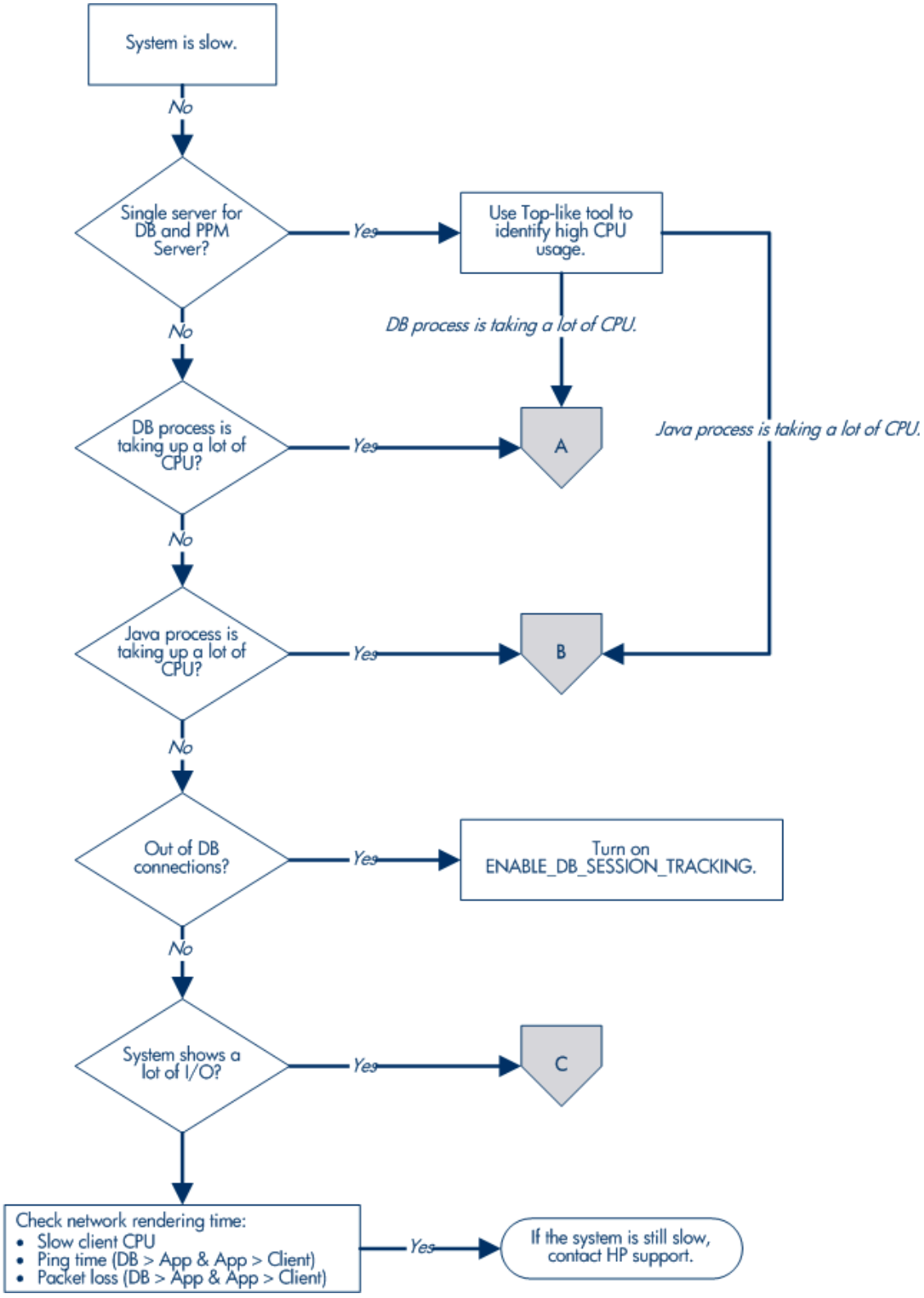


Figure 7-2. Identifying and addressing database performance problems (A)

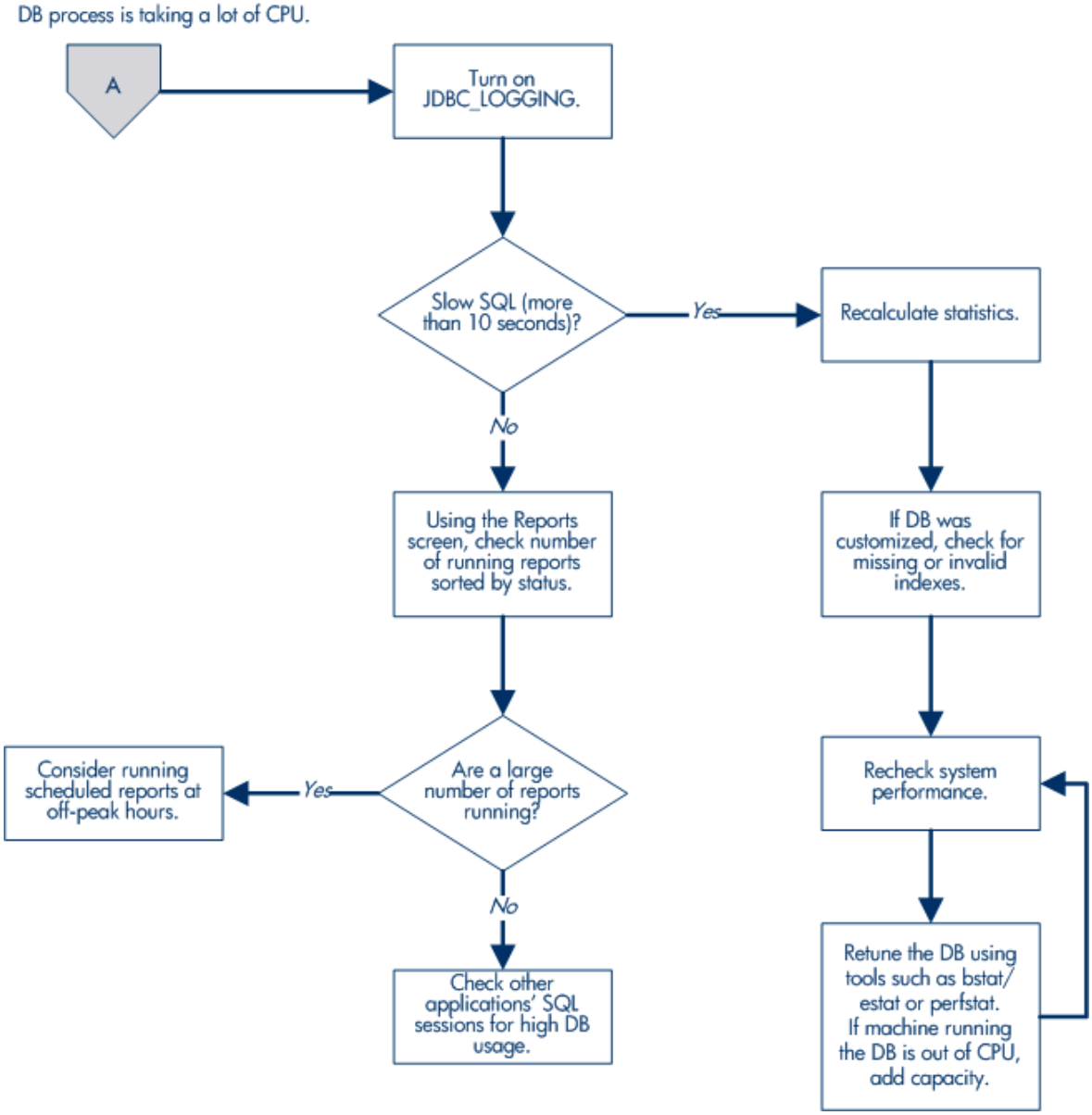


Figure 7-3. Identifying and addressing Java process performance problems (B)

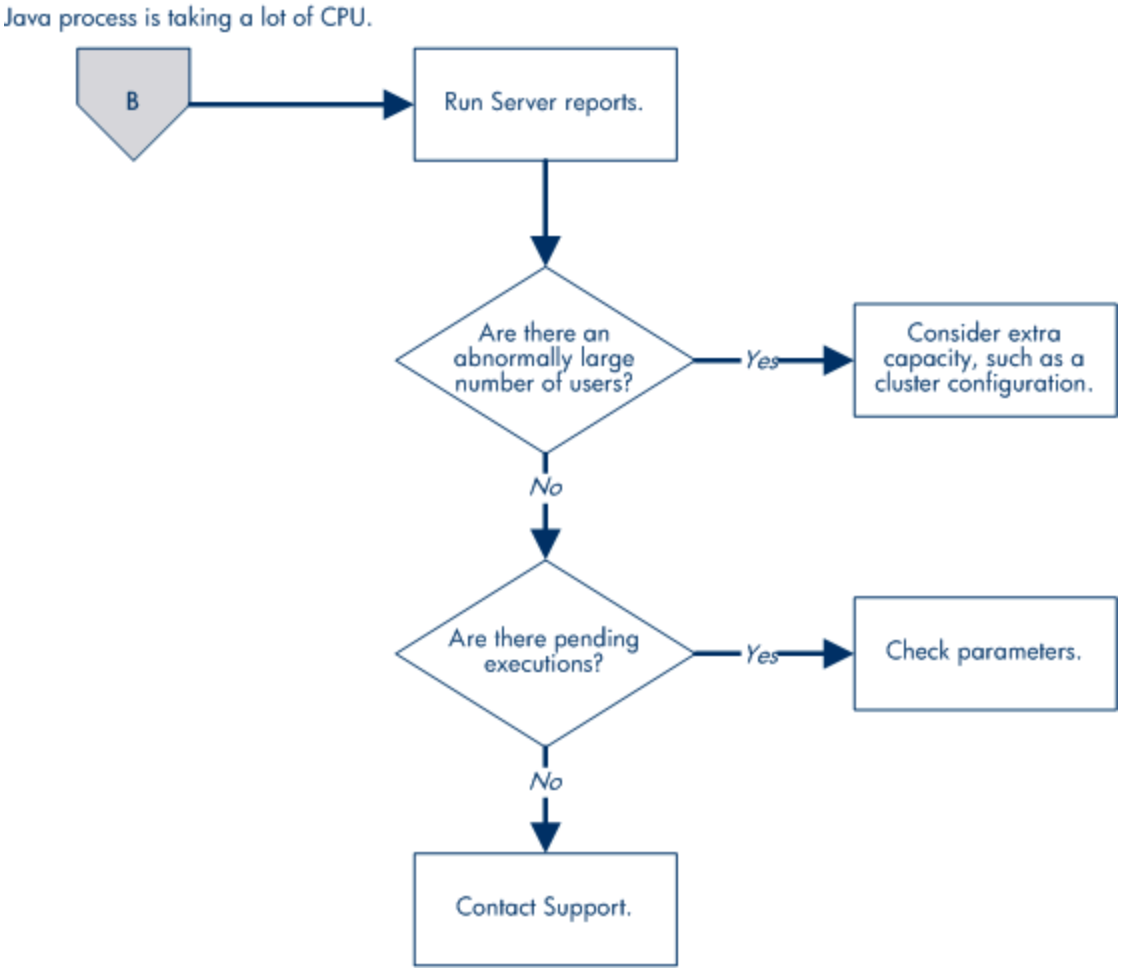
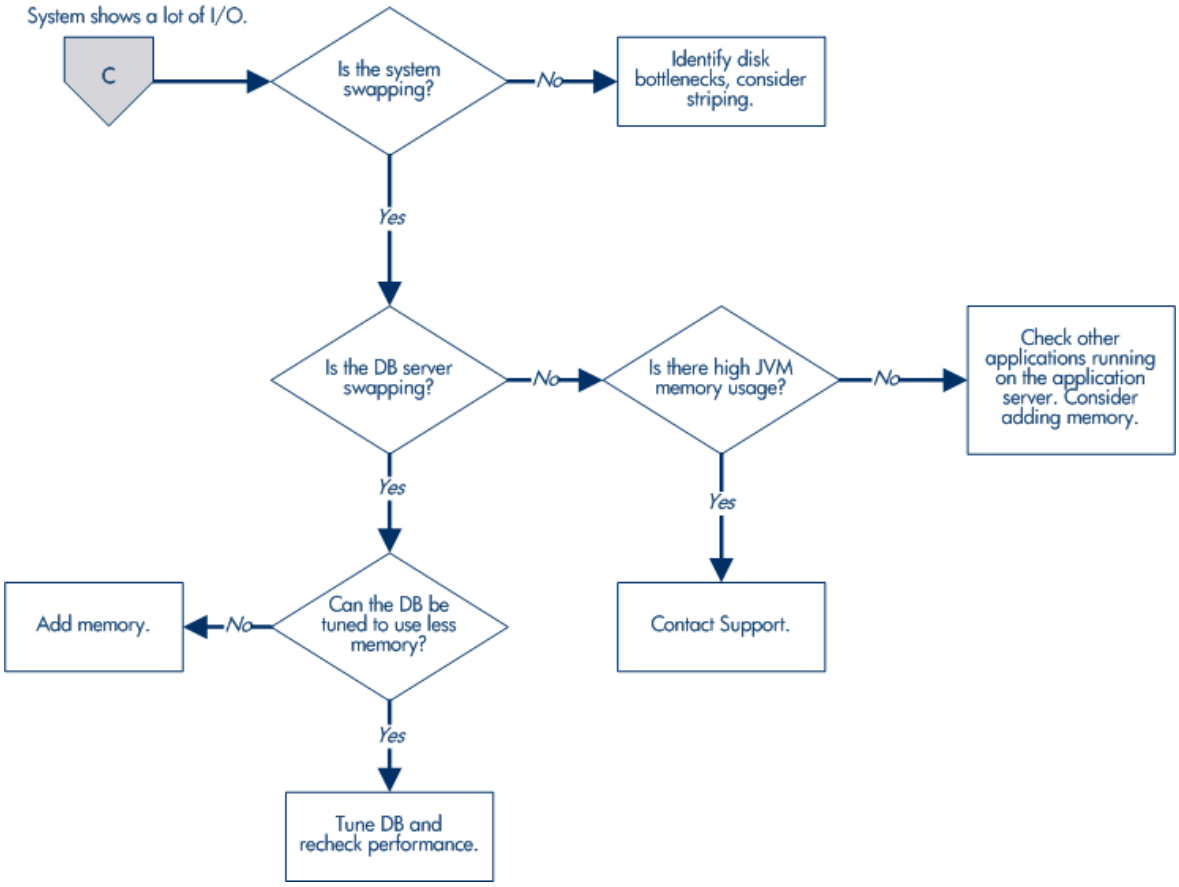


Figure 7-4. Identifying and addressing I/O performance problems (C)



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.

> **Tip:** Consider upgrading to the latest PPM service pack. HPE 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 "PPM Configuration Parameters" on page 401.

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 307 and "Running server reports from the command line " on page 310.

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, refer to the Oracle documentation.

Improving System Performance

This section provides information about how you can improve system performance. For additional information about improving performance on your PPM instance, see the *Deployment Best Practices for PPM Operational Reporting.*

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 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.

> **Note:** For information about how you can monitor background service activity, see"Background Services Monitor" on page 245.

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 deployments, HPE recommends that you dedicate one PPM Server on a single JVM to processing PPM 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 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.

> **Note:** For Information about PPM background services and instructions on how to schedule them, see "PPM Background Services" on page 266.

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.

> **Note:** For Information about PPM background services and instructions on how to enable or disable them, see "PPM Background Services" on page 266.

Start by running services on a single node, and then add services nodes as required

Although you can easily run services across multiple nodes, HPE 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.

> **Note:** 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 165.

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 database into the following segments:

- PPM tables
- PPM indexes
- Redo logs
- Rollback tablespaces
- Temporary tablespaces
- System tablespace
- Tablespace for management and related utilities

HPE recommends that PPM 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 the following table.

Table 7-1. Database disk recommendations

Disk	Recommendations for Data Placement
1	PPM tables
2	PPM indexes
3	Redo logs
4	• Rollback tablespaces • Temporary tablespaces • System tablespace • Tablespace for management and related utilities

For PPM database instances that have higher transaction volumes (more than 10,000 requests per month), HPE 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 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 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, 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 returned by a search. The default is 1000. You can increase or lower the value based on your environment to achieve a better performance. The value you set for the `REQUEST_SEARCH_RESULTS_MAX_ROWS` server configuration parameter is displayed as the default in the **Limit Rows Returned To** field.

  If you want to restrict the maximum number of records returned for the current search only, you can change the value in the **Limit Rows Returned To** field directly.
- 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 "PPM Configuration Parameters" on page 401. 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.
- `DAYS_TO_KEEP_LOGON_SESSIONS_ROWS` determines how many days to keep records of all user sessions.
- `HOURS_TO_KEEP_DEBUG_MESSAGE_ROWS` determines how long (in hours) to keep rows in the `KNTA_DEBUG_MESSAGES` table.

> **Note:** For descriptions of and valid settings for these parameters, see "PPM Configuration Parameters" on page 401

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 "PPM Configuration Parameters" on page 401 and "Logging Parameters" on page 480.

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.

> **Note:** HPE strongly recommends that you consult HPE Software Support before enabling 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 480.

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.

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" on page 316. For descriptions of all logging parameters, see "Logging Parameters" on page 480.

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 standard interface. For instructions on how to enable and schedule services, see "Enabling and Scheduling PPM Services " on page 271.

Monitoring Activity in PPM

This section contains information about configurable monitors that capture information on user interface activity, portlets, and background services in PPM. These monitors capture UI activities (mainly URL requests), background service runs, and portlet activity on the PPM Server.

Starting from PPM Center version 9.30, the `ENABLE_ALL_PERFORMANCE_MONITOR` parameter is available to generally control PPM Center monitors. You can configure this parameter in the Administration Console.

If you set this parameter to `true`, you enable the UI monitor, the portlet monitor, and the Background Services monitor, regardless of whether these three monitors are enabled or not. If you set this parameter to `false`, the UI monitor, the portlet monitor, and the Background Services monitor are enabled or disabled according to their own parameters. By default, the parameter is set to `false`.

Action Monitor

The PPM 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.

Change the Action Monitor Parameters Using the Administration 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 Administration Console:

1. Log on to PPM. From the menu bar, select **Open > Administration > Open Administration Console**.
2. Under the Administration Console node, select **Administration Task > Application configuration**.
3. On the Application Configuration page, select the desired PPM Server node from the **Scope** drop-down list, and search for parameter ENABLE_UI_MONITOR.
4. Set ENABLE_UI_MONITOR parameter to true or false to enable or disable the UI monitor.

To change the UI_MONITOR_THRESHOLD parameter value:

1. Log on to PPM. From the menu bar, select **Open > Administration > Open Administration Console**.
2. Navigate to the Application Configuration page, select the desired PPM Server node from the **Scope** drop-down list, and search for parameter UI_MONITOR_THRESHOLD.
3. Specify a value for the UI_MONITOR_THRESHOLD parameter.

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>/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)"
;
```

> **Note:** You can also use the 'UI_MONITOR_PERSIST_STATE' parameter in the Administration Console page to enable or disable persisting the captured information to the database table PPM_PERFORMANCE_LOG.

> **Note:** For information on how to create a report on action monitor activity, see Deployment Best Practices for PPM Operational Reporting.

SQLs in PPM_PERFORMANCE_LOG for Action Monitor

When the parameter ENABLE_ALL_PERFORMANCE_MONITOR is set to true, and if the latency of serving a URL request exceeds the threshold value specified in the UI_MONITOR_THRESHOLD parameter, then the SQLs that have been executed when serving the URL request are recorded in the table PPM_PERFORMANCE_LOG.

> **Note:** Only the top five SQLs that take the longest time when serving the URL request are recorded.

Portlet Monitor

The PPM 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 enable portlet monitor in Administration Console:

1. Log on to PPM. From the menu bar, select **Open > Administration > Open Administration Console**.
2. Navigate to the Application Configuration page, select the desired PPM Server node from the **Scope** drop-down list.
3. Search for parameter ENABLE_PORTLET_MONITOR, and set its value to true.
4. (Optional) Search for parameter **PORTLET_MONITOR_THRESHOLD**, and specify a value for it. The default value is 10,000 milliseconds.

This parameter determines the portlet load time threshold. If the amount of time required to load portlets exceeds the threshold value, portlet monitor logs portlet load information in the `thresholdLog.txt` file, which resides in the `<PPM_Home>/server/<PPM Server>/logs` directory.

> **Note:** If the parameter `PORTLET_MONITOR_PERSIST_STATE` is set to `true`, portlet monitor logs portlet load information in the table PPM_PERFORMANCE_LOG.

To disable portlet monitor, set both the parameter `ENABLE_ALL_PERFORMANCE_MONITOR` and the parameter `ENABLE_PORTLET_MONITOR` to `false`.

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 standard interface. The following sections provide instructions for performing these tasks.

Creating a Portlet Performance Report Type

1. Log on to PPM.
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`.

      > **Note:** 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

1. After you have created the report type from the PPM Workbench, return to the PPM 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 the previous page).

   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**

Field (* Required)	Description
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. If you are creating a page performance report, this field in disabled.
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 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:

> **Note:** For information on all of the background services in PPM, see "PPM Background Services" on page 266.

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.

> **Note:** 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>/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" ;
```

SQLs in PPM_PERFORMANCE_LOG for Background Services Monitor

When the parameter `ENABLE_ALL_PERFORMANCE_MONITOR` is set to `true`, and if a background services activity exceeds the threshold value specified in the `BACKGROUND_SERVICE_MONITOR_THRESHOLD` parameter, the SQLs that have been executed when running the activity are recorded in the table PPM_PERFORMANCE_LOG.

> **Note:** Only the top five SQLs that take the longest time when running an activity are recorded.

Viewing the Services Audit Results Page

You can open the Services Audit Results page to quickly view information about PPM background services.

1. Log on to PPM.
2. On the **Open** menu, click **Administration > View Services Audit Page.**

> **Note:** 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 Services " on page 271.

The following table lists the columns displayed on the page.

Column Heading	Description
Service Name	Name of the background service
Status	Status (enabled or disabled) of the background service
Is Running?	Shows whether the service is running, not running, or disabled
Run Interval	Run interval set for the service
Last Run Node	Node on which the service was last run **Note:** A dash (-) character in this column indicates that the service was triggered, but did not run because there are no data to process.
Last Completed Run	Time and date the service last ran
Next Scheduled Run	Time and date the service is scheduled to run next

Viewing Success Logs of Background Services

If you have specified the following parameters in Administration Console, you can view success logs of background services.

Parameter Name	Description	Values
ENABLE_LOG_ SUCCESS_SERVICE_ LIST	Specify the reference codes of background services. If these services are run successfully, PPM will record the success logs for them. Reference codes are separated by semicolon.	For example: `_COST_RATE_RULE_ UPDATE_SERVICE; _ COST_ROLLUP_SERVICE`
SERVICE_RECORDS_ EXPIRATION_DAYS	Specify the duration (in days) of the success logs. The log that expire the duration will be removed automatically from the Service Records page.	Default: `14`

To view success logs of a background service:

1. On the Service Audit Results page, click the service foe which you have enabled recording success logs.

   The Service Records page for the background service lists all the success logs.

2. Click the detail link to view details of a log.

Limitation: In 9.40, you can only enable recording success logs for Cost Rate Rule Update Service.

Accessing Services Exceptions Details

On the Services Audit Results page, you can see whether any services have thrown exceptions.

If a service has encountered one or more exceptions or errors during its last 50 runs, a red icon (**!**) is displayed to the left of the service name on the Services Audit Results page. You can change the threshold for displaying the icon by changing the value for the `SERVICE_RECORDS_RETAIN_COUNT` server configuration parameter in the `server.conf` file.

To view details about exceptions a service has encountered, click the service name. The Search Exceptions page opens and displays a list of the exceptions.

The Search Exceptions page provides the following information:

- Presence or absence of exceptions during a particular service run.
- Name of the node on which the service was run
- Time the service run started
- Time the service run finished
- Duration of the run, in seconds
- Any additional information provided by the service. All services by default provide the memory footprint during the start and end of the service. This information is useful to troubleshoot any issues related to memory consumption by a service run.

To enable this functionality, add the `LOG_EXCEPTIONS_TO_DB` server configuration parameter to the `server.conf` file and set its value to `true`.

Purging Exceptions Thrown by Services

Whenever a service run is deleted from the system, the exceptions related to that service run are also removed. By default, the system retains the 50 most recent service run records. You can change this

by setting a new value for the SERVICE_RECORDS_RETAIN_COUNT server configuration parameter in the server.conf file.

Accessing Application Exception Details

If an exception occurs, the user sees an error message that displays a GUID number and advises the user to contact the PPM administrator. The PPM administrator can then get detailed information about the exception from the new Search Exceptions page.

> **Note:** To access the Search Exceptions page, you must have an Administrator license and the following access grants:
>
> - Server Tools: Execute admin tools
> - Sys Admin: Edit Services Schedules

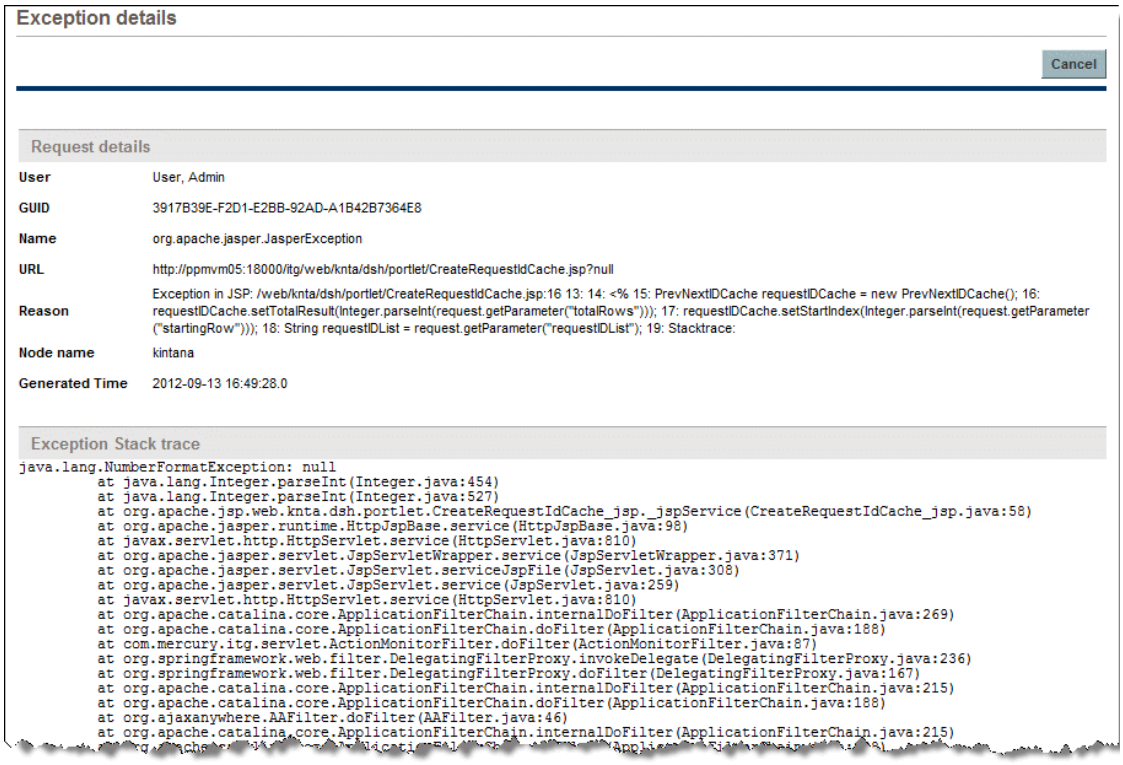
To enable this functionality, set the LOG_EXCEPTIONS_TO_DB server configuration parameter value to true.

To access and use the Search Exceptions page:

1. Log on to PPM.
2. From the menu bar, select **Search > Administrative > Exceptions.**

   The Search Exceptions page opens.
3. In the **View Details for Exception GUID** field, enter the GUID provided in the error message.
4. Click **Go.**

   The Exception details page opens.



The Exception details page displays the following information:

Field	Description
User	Full name of the logged-in user who encountered this exception. This field is empty if the exception was thrown by a background service.
GUID	GUID identifier for this exception. Most web exceptions are tagged with a GUID. Background service exceptions usually do not have a GUID.
Name	Name of the exception.
URL	URL of the page that encountered this exception.
Reason	Reason the exception occurred. For background service exceptions, this can be an error message.
Node name	Name of the node on which the exception occurred.
Generated Time	Time at which the exception occurred.
Exception Stack trace	Stack trace for the exception.

Purging Exceptions (Other than Services Exceptions)

Exceptions (except for services-related exceptions) older than 14 days are automatically purged by default. You can control the frequency with which exceptions are purged by setting a new value for the `EXCEPTIONS_RETAIN_PERIOD` server configuration parameter in the `server.conf` file.

Identifying Database Connection Issues

The Connection details page enables you to identify suspect connections that are leaking. To access the Connection details page, you must have an administrator license and the "Server Tools: Execute admin tools" access grant.

To enable this feature, set the `ENABLE_CONNECTION_CORRELATION` server configuration parameter to `true.`

To access the Connection details page:

1. Log on to PPM.
2. From the menu bar, select **Open > Administration > View connection correlation.**

The table on the left portion of the page displays the stack trace for each connection recorded when the connection was acquired from the pool.

To display the stack trace associated with the connection at the time it was acquired, click the corresponding connection entry.

> **Note:** Because this feature involves some database overhead, HPE recommends that you use it only to troubleshoot connection leak issues.

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 functionality. It is platform-dependent because it uses different mechanisms to generate thread dumps on Windows than on other, UNIX-like platforms.

> **Note:** 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 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 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 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 enters the critical condition.

> **Tip:** To collect thread dumps when a threshold value is not desired, you can,
>
> - Use the Watchdog Tool and set `memory_threshold` to `0`. Or,
> - (Recommended) Use Stack Trace tool jstack to create all thread dumps on all operating systems.
>
> For example, **`jstack pid >a.log`**
>
> The jstack tool is present in the <*JDK_HOME*>`/bin` directory.

Generating the GC Log

The Watchdog utility requires that a Java GC log file be present.

To enable verbose GC logging when starting PPM:

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.

> **Note:** 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
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	

Name	Description	Required	Default
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:

PPM Cache Architecture

This section describes the PPM Cache architecture up to PPM 9.40, as well as the changes introduced to the Cache in PPM 9.31. It should help PPM Administrators to correctly tune PPM Caches in order to achieve optimal system performance.

PPM Cache Overview

There are two type of caches in PPM that can be tuned by PPM Administrators:

- **Legacy cache** (table components, request type search fields, list validation values, and so on)

  It is configured in the `<PPM_HOME>/conf/tune.conf` directory. Setting the parameters in `server.conf` works too, and some of the parameters can be edited from the Administration Console. You can view its cache statistics in the **Server Cache Status** report in Server Tools of PPM Workbench.

- **New cache** (requests, request types, modules, portlets, workflows, and so on)

  It is configured in the `<PPM_HOME>/conf/cache.conf` directory. You can view its cache statistics in the **CacheManager Statistics** report in Server Tools of PPM Workbench.

> **Note:** There is a third type of caches in PPM (the Hibernate second level cache). However, this is an internal cache that cannot be configured or tuned. Its configuration is defined in the `<SERVER_HOME>\deploy\itg.war\WEB-INF\conf\ehcache.xml` directory and it should not be tampered with.

Advantages of New Cache over Legacy Cache

The new cache has the following advantages over the legacy cache:

- All cache objects are stored using Java SoftReference. As a result, if the JVM runs out of heap memory, objects in the cache will be automatically garbage collected to free up memory. This makes it possible to store large amounts of data in the cache without risking out-of-memory issues under heavy system load.
- There are more cache configuration parameters (at least until PPM 9.31, in which new cache configuration has been simplified).
- It is possible to flush a single object in the new cache by using the `ksc_flush_cache` special command (which only works with new cache), whereas only full cache flush is possible when using `kRunCacheManager.sh` (which works with both legacy and new cache).
- The new cache provides configurable staleness checks, whereas such checks are hard coded in the legacy cache.
- More data is included in the cache report (number of staleness checks, average load time, and number of flushes of different types).

Understanding PPM Cache Statistics Reports

Legacy Cache

The following information is available for each legacy cache in the **Server Cache Status** report:

- The maximum number of objects that can be cached (cache size)
- The number of additional objects that can be cached (free units)
- The number of hits, misses, and swaps

  > **Note:** Swaps mean replacing an object with another one when the max size is reached.

- Miss rate (the lower the better)
- Estimation of the amount of memory taken by the cache

New Cache

The following information is available for each new cache in the **CacheManager Statistics** report:

- Hits, misses, and hit rate (the higher the hit rate, the better)
- The number of cache flushes

  Cache flushes are broken down by the categories "old", "idle", "soft reference reclaimed", and "max cache size reached"
- The average load time to load an object from database when it is not in the cache
- The number of staleness checks performed
- The maximum cacheable objects (cache size), cached object count, and maximum idle time
- Whether the cache is distributed or not

  If it is distributed, removing an object from the cache in any node of the PPM cluster will send a message to all the other cluster nodes to remove that object from their caches.

PPM Cache Tuning

Tuning PPM Cache performance should be done at the same time as tuning PPM JVM heap size. This means finding the right balance between the following two things:

- JVM heap size

  Before PPM 9.20, only 32-bit JVM was supported, which limited the JVM heap size to 1.3 GB. Since PPM 9.20 and the adoption of 64-bit JVM, the size of the JVM heap memory is only limited by the installed physical memory. However, too large heap size can result in long full garbage collection times, during which the application is unresponsive. It is common to see PPM JVM heap sizes of up to 4 GB, and sometimes larger on Service nodes.
- Cache size

  A larger cache size means more cacheable objects, a better hit rate, and fewer objects to reload from the database, which result in better application performance. However, if cached objects end up taking too much memory, it will impact the performance of the application and might even cause out-of-memory problems. Note that only legacy cache is prone to cause such memory problems, as the JVM will automatically discard objects from the new cache whenever the available free memory runs too low.

To Summarize

- If JVM heap size is too large, full garbage collection periods will be too long and the application might become unresponsive for seconds, degrading users experience.
- If caches are too large and end up taking too much memory, the performance of the application will degrade and in the case of legacy cache it might even cause out-of-memory issues.

Tuning PPM cache

Following these steps may end up conflicting with each other. If that happens, use your best judgment.

- If you see a large number of swaps (in legacy cache) or "max cache size reached" flushes (in new cache), increase the cache size.
- If you see some "soft reference reclaimed" flushes (in new cache), increase the JVM heap or reduce the cache size for caches take large amounts of space.
- If you see a high miss rate (above 20% in legacy cache) or a low hit rate (below 80% in new cache) even after prolonged PPM usage (at least one day of heavy usage), increase the cache size.
- If you notice long full garbage collection time (many seconds) during which the system is unresponsive, reduce the heap size or better tune the JVM garbage collection. Note that you will need to use a JVM monitoring tool in order to ensure that JVM pauses are caused by full garbage collection.

Additional tips for tuning the PPM cache

- There is no "standard" cache configuration. Measure, tune, flush, and repeat until you reach satisfactory numbers. All PPM usages are different, and as a result cache configurations should be tuned accordingly.
- When tuning the PPM cache, try to do so after capturing statistics during the highest peak load time (usually happening on Friday afternoon or Monday morning). Tuning cache only makes sense if it is done to properly handle peak load usage.
- Try not to flush the caches (using `kRunCacheManager` or `ksc_flush_cache`) in an automated way unless you really have to. If you use `kRunCacheManager`, NEVER use the "**A**" option to flush all caches in an automated script. You should not flush more caches than necessary.
- If memory limit does not allow you to set the proper max cache size for all the entities, you can rank the caches by the value "Average Load time" x "Misses", and first increase the cache size of the

cache(s) with the highest value. They are the caches most likely to have a measurable performance impact.

- You might want to tune differently the PPM nodes in your cluster depending on whether they are Service nodes or Web User only nodes. The entities loaded (and thus the optimal cache settings) are different. For example, a pure Service node will never load portlets or menus, but might need a larger cache size for fiscal periods.
- Don't forget that when you use `kRunCacheManager.sh`, it will always flush selected caches on all nodes of your PPM cluster, but it will reset caches statistics and force garbage collection only on the node it is connected to.

Important notes about MLU environments

In the environments where more than one language is installed on PPM, if a same object is loaded by different users using different languages, the object will be stored in the cache once for each of the different languages used. Your cache should be set accordingly.

For example, on a system with 3 languages installed for PPM, if you have 20 request types, you should set the maxSize value for the request types cache to 60, i.e. 20 (request types) times 3 (languages). However, if you have 1000 users, you might still set the users cache maxSize value to 1000, since a given user will usually always log in PPM using the same language; so each user object will be loaded in the cache most often once, using the language of the user.

Flushing PPM Cache with kRunCacheManager.sh and ksc_flush_cache

There are two ways to flush a PPM Cache in a manual or automated way: `kRunCacheManager.sh` (command line tool), or `ksc_flush_cache` (special command). They should be called after modifying the data directly in the database without going through one of the supported PPM interfaces (Web UI, SOAP Web Service, REST Web Service, and so on).

kRunCacheManager.sh

`kRunCacheManager.sh` is a shell script that is run from the command line.

It works with both legacy cache and new cache.

Syntax

```
sh ./kRunCacheManager.sh [<URL>] <cache number>
```

- "URL" parameter is optional. If omitted, it will connect to the first running RMI_URL defined in `server.conf`. You can pass multiple RMI urls, separated with semicolons (;), and it will connect to the first running one. It is safe to omit this parameter unless you want to connect to a specific PPM node, for example to reset cache statistics or request a garbage collection.
- "Cache Number" parameter is the number next to the cache that you want to flush. In order to view the list of caches with their numbers, run the command without any parameter to list all caches and be prompted for possible options. You can also input a letter to trigger the following actions:
  - **A**: Flush All Caches. It is strongly advised not to use this option when running `kRunCacheManager` in an automated way.
  - **B**: Flush Validation Caches. This will only flush the validation related caches. It can be used when validation definition or values are directly edited in PPM database.
  - **C**: Reset Cache Statistics Counters. This only affects the node you are connected to.
  - **D**: Force Garbage Collection.

Issues and Limitations

- The order of the caches in the list is not consistently enforced and can vary between environments or PPM versions. As a result, if you flush a specific cache designated by its number in a script, it is possible that the corresponding cache may change at some point in the future. You should verify that the cache numbers haven't changed after every PPM upgrade or environment change.
- It is not possible to flush only one entity in a cache; the whole cache has to be flushed. This can result in performance impact as all objects from the flushed cache will need to be reloaded. This could have performance impact under heavy load if some caches are flushed too often.
- It is strongly advised NOT to use the "flush all caches" option ("`kRunCacheManager.sh A`") in an automated way, as it may have performance impact under heavy system load.
- If the cache is flushed using `kRunCacheManager.sh` while the cache maintenance thread is running (it runs every 10 seconds by default), an exception may be fired and the `kRunCacheManager.sh` cache flush action may be ignored.

ksc_flush_cache

`ksc_flush_cache` is a PPM special command that can be invoked from any command step (workflow step, PPM report command step, and so on.).

It works with new cache only.

Syntax

```
ksc_flush_cache <cache-name> [<id>]
```

- "cache-name" parameter is the name of the new cache, as defined in `cache.conf`. For example, in the following `cache.conf` line, the cache name is datasouce:

  ```
  cache.datasource.title = Dashboard Datasources
  ```

- "id" parameter is optional. If it is omitted, the whole cache is flushed. If it is specified, only that entity is removed from the cache.

Issues and Limitations

- `ksc_flush_cache` only works with new cache.
- As of PPM 9.30, passing an entity ID to `ksc_flush_cache` will only work when the cache key is an integer value. If it is a string value, it will not be flushed, and the only option to remove the designated entity will be to flush the whole cache.

PPM Cache Changes in PPM 9.31

The following changes have been done to PPM Cache in 9.31 in order to correct the existing issues and limitations.

Conversion From Legacy Cache to New Cache

All the legacy caches listed in `kRunCacheManager.sh` but one (Scoring Criteria) have been converted to new caches.

All these caches are now configured from `cache.conf`, and their parameters in `tune.conf` (or `server.conf`/Administration Console) have been deprecated and are not used anymore.

Simplification of New Cache Configuration

Some parameters in `cache.conf` have been removed in an attempt to simplify cache configuration.

- The only parameter that can be tuned for all caches is the cache size (the parameter "maxSize").
- Parameters "maxAge", "maxIdleTime", "resolver" and "stalenessCheckGraceInterval" have been removed.

- Parameter “distributed” is inferred automatically based on whether a staleness check is defined (distributed = false) or not (distributed = true).
- Cache is automatically set to disabled if maxSize = 0.

A staleness check has also been added for request types and table components even though it is not enabled by default. If you rely on direct DB updates to modify request types or table components, it is preferable to enable the staleness check rather than disable the cache. Staleness check will only work if the column LAST_UPDATE_DATE is modified during data update.

One side effect of cache simplification is that the cache maintenance thread (previously used to enforce maxAge and maxIdleTime of cached objects) is now only used to reload the `cache.conf` configuration when modified at runtime. It does not cause issues anymore when running at the same time as `kRunCacheManager.sh`.

Changes to CacheManager Statistics Report

- Flush counts of types “old” and “idle” have been removed.
- “Cache Flush All” count has been added. It displays the number of times of the whole cache being flushed. It helps identify abuses of “`kRunCacheManager.sh A`” or of unnecessary full caches flushes.

  Running “`kRunCacheManager.sh A`” will increment this value by 1 for all caches.

Changes to kRunCacheManager.sh

- The order of listed caches has changed: it still lists legacy cache followed by new caches, but now the new caches are ordered alphabetically by cache name.
- When all caches are listed, the cache names of new caches are displayed in parenthesis after the cache title.
- When flushing new caches, you can now use the cache name in place of the cache number. It is advised to always use the cache name, as it does not rely on any ordering of caches in that list.
- An extra optional parameter has been added to the command, to pass the entity ID to flush. So you can now flush one specific entity from a cache using the following syntax:

  ```
  sh ./kRunCacheManager.sh [<URL>] <cache name (or cache number)> [<entity ID>]
  ```

- A new action (**E**) has been added. It lists all keys in each of the new caches along with their type (String or Integer). This can help diagnose whether a specific entity is currently stored in the cache

or not. Note that in MLU environments, the same key can be displayed multiple times if it is stored in the cache using different languages.

Changes to ksc_flush_cache

- Passing an entity ID to `ksc_flush_cache` will now work regardless of the key type (String or Integer).
- You can now flush the legacy caches by passing their cache number (as in `kRunCacheManager.sh`) instead of the cache name. Since there is only one legacy cache left in 9.31, it means passing “1” as the cache name flushes the “Scoring Criteria” cache.

More Questions?

If you have more questions, join the conversation and ask your questions on:

- HPE PPM Customer Support forum (if you are an existing PPM Center customer):

  http://h30499.www3.hp.com/t5/Project-and-Portfolio-Management/bd-p/project-portfolio-mgnt-cust-forum
- Public HPE PPM Support and News forum:

  http://h30499.www3.hp.com/t5/Project-and-Portfolio-Management/bd-p/itrc-935

Chapter 8: Maintaining the System

This chapter provides information on how to maintain your PPM instance. The initial sections include descriptions of tools available in the PPM standard interface, the Administration Console, and the PPM Workbench. Later sections address the maintenance tasks required to keep your PPM instance running smoothly.

This section includes the following:

Administration Tools in the Standard Interface

The following sections provide information about the administration tools you can access through the PPM standard interface:

Viewing Server Running Server Reports, Requests, and Packages

1. Log on to PPM.
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

1. Log on to PPM.
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

1. Log on to PPM.
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.**

PPM Background Services

This section provides information about the background services available in PPM, and instructions on how to enable and schedule them. It also provides guidelines for scheduling background services to optimize resource use and system performance.

The following table lists the PPM background services.

> **Note:** For information on background services monitoring in PPM, see "Monitoring Activity in PPM" on page 239.

Table 9-1. Background services in PPM Center

Service Name	Description
ALM Startup	Ensures that the quartz scheduler that synchronizes PPM 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	When 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

Table 9-1. Background services in PPM Center, continued

Service Name	Description
Dog	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 synchronized to staffing profiles • Adds projects to the queue for Cost Rollup Service, which updates project cost in the workplan and in the financial summary.
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 entity, and removes those it finds from the PPM file system.
Evaluate TM Approvers	An 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 *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 Financial Management.
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

Table 9-1. Background services in PPM Center, continued

Service Name	Description
	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.
Integration SDK Sync Service	Periodically synchronizes user stories in all sprints from agile management systems to PPM Center tasks.
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 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. For information about PPM Mobility Access, see the *Demand Management Configuration Guide.*
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.
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	Removes duplicate rows in the Pending EV Updates table.

Table 9-1. Background services in PPM Center, continued

Service Name	Description
Cleanup	
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.
Program Health	Automatically updates program health indicators.
Project Health	Automatically updates project health indicators.
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 *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	Synchronizes the budgets linked with the staffing profiles that are updated

Table 9-1. Background services in PPM Center, continued

Service Name	Description
Linked Budget Sync Service	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.
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 HPE Document Management module uses PPM 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, 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 Time Management or the My Tasks portlet. For more information about this service, see the *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 *Project Management Configuration Guide.*
Time Sheet Notifications	Enables notifications to be sent on time sheets.
TM-PM Sync	Synchronizes time sheet updates from Time Management to project work plan tasks in 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.
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.

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 229.

Enabling and Scheduling PPM Services

You can enable and schedule the PPM background services through the standard interface.

To enable and schedule PPM services:

1. Log on to PPM.
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.

   > **Note:** 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.**

   > **Note:** 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 ( ).

> **Note:** 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.

> **Note:** 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.

HPE recommends that, if you have a server cluster configured, and there are nodes in the cluster that do not handle incoming user requests, you disable the nodes from running the PPM background service. For information about disabling nodes or restricting services from running on non-services nodes, see "Disabling Nodes from Running Background Services" in "Verifying Successful Cluster Configuration" on page 180.

For information about how to view the current status of background services, see "Viewing the Services Audit Results Page" on page 246.

Tools in the Administration Console

The following sections provide information about how to access the Administration Console and about the tasks you can use it to perform:

- "Running SQL Queries from the Administration Console" on page 291
- "Gathering Information for HPE Software Support from the Administration Console" on page 296
- "Changing Data Display in Administration Console Tables" on page 302
- Using the Unchecking Showing Total Number Tool from Administration Console
- "Installing Autopass License Key File and Viewing License Summary in Administration Console" on page 275

In order to access and use the Administration Console, you must:

- Have the User Administration license
- Have one or more of the following access grants:

Access Grant	Permissions
Sys Admin: Server Tools: Execute Admin Tools	Lets the user access the Administration Console and the server tools.
Sys Admin: Server Tools: Execute SQL Runner	Enables the **SQL Runner** menu in the Administration Console and lets the user run SQL queries from the Administration Console. Without this access grant, the **SQL Runner** menu is invisible.
Sys Admin: Server Tools: Execute File Browser	Enables the File Browser menu **Browse PPM Server files** in the Administration Console and lets the user browse and download PPM Server files. Without this access grant, the File Browser menu is invisible.

For more information about security groups and access grants, see the *Security Model Guide and Reference.*

Opening the Administration Console

You can open the Administration Console from either the PPM standard interface or from a Web browser window.

Opening the Administration Console from the PPM Standard Interface

1. Log on to PPM.
2. From the menu bar, select **Open > Administration > Open Administration Console.**

The left panel of the Administration Console window displays the **System Health,Administration Task, and Support Task** sections. The right panel displays details about the item selected in the left panel.

The **Nodes** item under **System Health** is selected by default. If your PPM instance includes multiple nodes, the **Node Name** column lists all of the nodes.

> **Note:** If you configure secure Web logon for PPM, you can launch Administration Console on HTTPS. For details, see "Configuring Secure Web Logon" on page 96.

Opening the Administration Console from a Web Browser Window

1. In the address field of a Web browser window, type the following:

   <*PPM_URL*>/itg/admin/AdminConsole.do

   where <*PPM_URL*> is the Web location (top directory name) of the PPM Server.

   Example: `http://12.34.56.789:12345/itg/admin/AdminConsole.do`

   The PPM logon page opens.

2. Log on to PPM.

The Administration Console window opens. The left panel of the window displays the **System Health,Administration Task, and Support Task** sections. The right panel displays details about the item selected in the left panel.

The **Nodes** item under **System Health** is selected by default. If your PPM instance includes multiple nodes, the **Node Name** column lists all of the nodes.

> **Note:** If you configure secure Web logon for PPM, you can launch Administration Console on HTTPS. For details, see "Configuring Secure Web Logon" on page 96.

Viewing PPM Server Status from the Administration Console

You can use the Administration Console in PPM to quickly assess the status of the nodes in your instance.

> **Note:** For information about how to run the Server Status Report from the Admin Tools window in

the PPM Workbench, see "Running Server Reports from the Admin Tools Window" on page 307.

1. Open the Administration Console. (See "Opening the Administration Console" on page 273.)

   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 whether a node 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.

Installing Autopass License Key File and Viewing License Summary in Administration Console

The Install License page in the Administration Console allows you to:

Install an Autopass License Key File

1. Obtain and save the license file somewhere on your computer.

   For information about obtaining an Autopass license, see "Activating and Generating Autopass License" on page 86.

2. Log in to PPM Center.
3. From the menu bar, select **Open > Administration > Open Administration Console**.
4. In the navigation pane, select **Administration Task > License**.
5. On the License Install page, click **Browse** to locate the license file you saved, and then click **Install**.

   The license file is installed and becomes effective right away, with a message popping up showing how many licenses are installed.

   There is no need to stop and restart the PPM Server for the license to become effective.

View A Summary of Autopass Licenses Purchased and Installed on the PPM Server

You can view a summary of Autopass licenses that you purchased and installed on the PPM Server from the Administration Console window.

1. Log in to PPM Center.
2. From the menu bar, select **Open > Administration > Open Administration Console**.
3. In the navigation pane, select **Administration Task > License**.

   The View History section displays a summary of all Autopass licenses installed on the PPM Server that are valid for use. Only activated licenses are shown in the View History section.

4. Expand the plus sign in front a license entry to view details.

   The following table describes columns in the View History table:

Column	Description
License	Full title of the Autopass license key you generated from the HPE Licensing for Software portal
Product Feature	A list of product features that are available with the Autopass license

Column	Description
Capacity	Number of users available with the Autopass license
Start Date	Start date of an Autopass license key
End Date	End date of an Autopass license key
Delete	Click the **Delete** button to remove a corresponding Autopass license key

> **Note:** To view consumption status of implicit features available with each Autopass license, you can still go to the License Administration window in the PPM Workbench.

Remove a License Key

1. In the Administration Console navigation pane, select **Administration Task > License**.
2. In the View History section, click **Delete** for the license key that you want to remove.

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 add translations of fiscal period names, shift existing fiscal periods, and import and export fiscal periods.

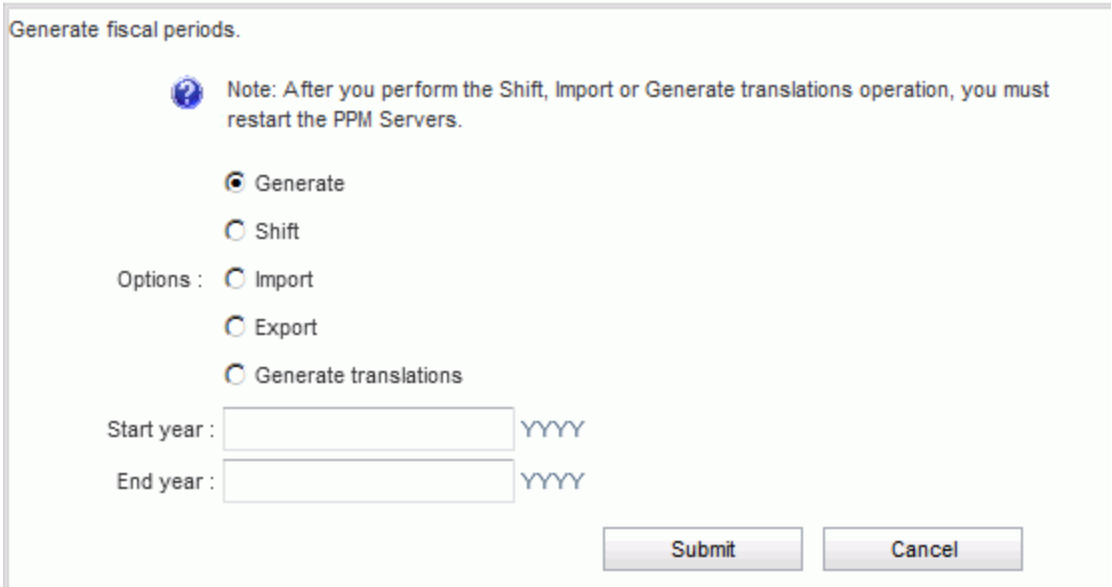
> **Note:** For information about how to use the `kGenFiscalPeriods.sh` script to generate fiscal periods, see the *Generating Fiscal Periods* guide.

- "Generating Fiscal Periods from the Administration Console" on the next page
- "Using the Administration Console to Shift Existing Fiscal Periods" on page 279
- "Using the Administration Console to Import Fiscal Periods" on page 280
- "Using the Administration Console to Export Fiscal Periods" on page 281
- "Using the Administration Console to Generate Translations for Fiscal Periods" on page 282

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 273.)
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.

   > **Note:** 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.

> **Note:** For information about how to stop and start PPM Servers, see "Starting and Stopping the PPM Server on a Single-Server System" on page 77.

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.

> **Note:** Shifting fiscal periods changes period data in the database. HPE 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.

   > **Note:** 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.

   > **Note:** 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 273.)
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 a Single-Server System" on page 77.

> **Note:** 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:

1. Open the Administration Console. (See "Opening the Administration Console" on page 273.)
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 from which you want to copy fiscal period definitions.
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, your PPM Servers.

> **Note:** For information about how to stop and start PPM Servers, see "Starting and Stopping the PPM Server on a Single-Server System" on page 77.

Using the Administration Console to Export Fiscal Periods

To export current fiscal period definitions from PPM to a specific file:

1. Open the Administration Console. (See "Opening the Administration Console" on page 273.)
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.
6. Click **Export to File.**
7. In the File Download dialog box, click **Save.**
8. Navigate to the PPM Server directory in which you want to store the file, and the save the file.

   > **Note:** 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 it now contains the file you exported.

   > **Note:** For information about how to use the `kGenFiscalPeriods.sh` script to export fiscal period definitions, see the *Generating Fiscal Periods* guide.

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, a user can view the fiscal periods in the session language he 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:

1. Open the Administration Console. (See "Opening the Administration Console" on page 273.)
2. In the left panel, under the **Administration Task** section, 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. (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 added languages.
6. To persist the added translations, click **Commit.**
7. To implement your changes, stop, and then restart, the PPM Servers.

   > **Note:** For information about how to stop and start PPM Servers, see "Starting and Stopping the PPM Server on a Single-Server System" on page 77.

> **Note:** For information about how to use the `kGenFiscalPeriods.sh` script to create periods for languages installed on a PPM instance, see the *Generating Fiscal Periods* guide. For information about using multiple languages on a single instance of PPM, see the *Multilingual User Interface Guide.*

Viewing and Modifying Server Configuration Parameters from the Administration Console

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 your PPM Server (s).

The Administration Console displays read-only parameters in gray text. Read-only parameters are either sensitive parameters such as passwords or parameters that you cannot change without compromising the PPM 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.

> **Note:** The Administration Console displays only the parameters that are defined in the 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), HPE recommends that you modify the values from the Administration Console. (See "Modifying Parameters from the Administration Console" on the next page.)

> **Note:** 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 89. For information about how to change parameter values in the `server.conf` file, see "PPM Configuration Parameters" on page 401.

Viewing Parameters from the Administration Console

1. Open the Administration Console. (See "Opening the Administration Console" on page 273.)
2. In the left panel of the Administration Console window, expand the **Administration Task** section, and then click **Application Configuration.**

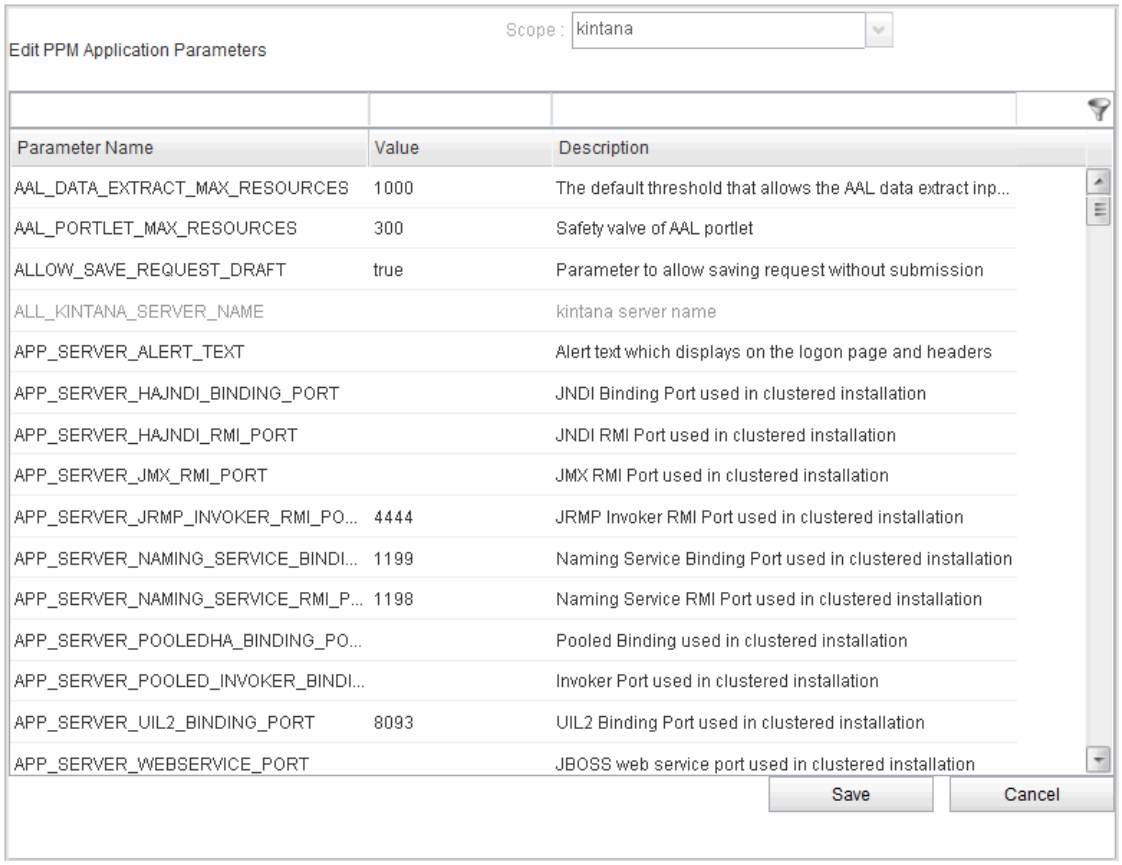
   The Edit PPM Application Parameters table in the right panel lists all the server configuration parameters, along with their descriptions and current values for the selected scope. (The **Scope** value defaults to **Cluster** or, if the instance consist of just one node, the name of the single PPM Server.)
3. To specify the scope of the parameters listed, 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.

     > **Note:** 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.
4. To view the setting for a specific parameter, in the box above the **Parameter Name** heading, type the first few letters of the parameter name to jump to the parameter names that match in the list.

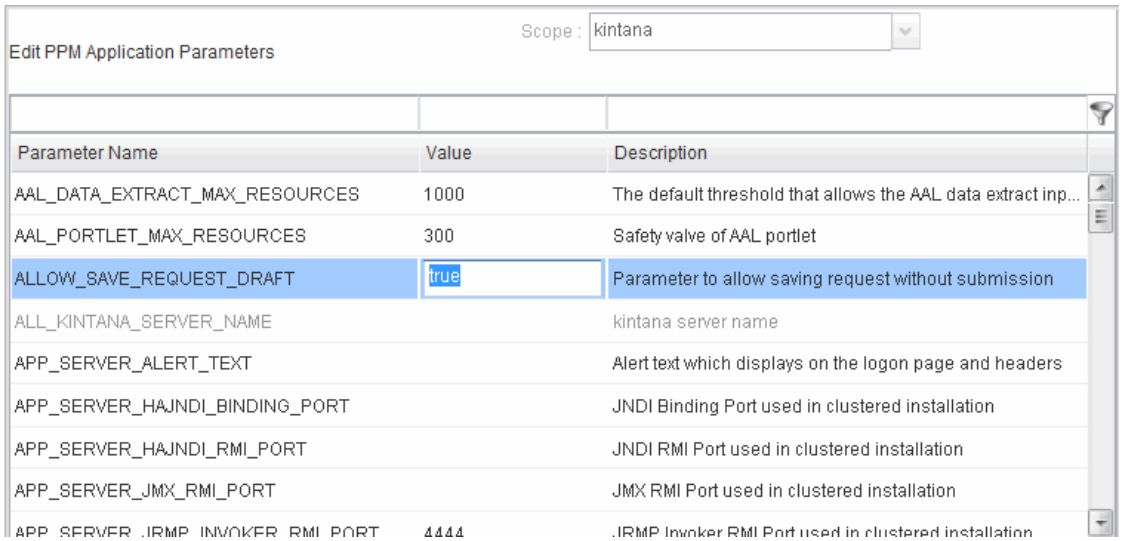
Modifying Parameters from the Administration Console

1. Open the Administration Console. (See "Opening the Administration Console" on page 273.)
2. In the left panel of the Administration Console window, expand **Administration Task,** and then click **Application Configuration.**

   The **Scope** list displays the names of all nodes in the cluster.



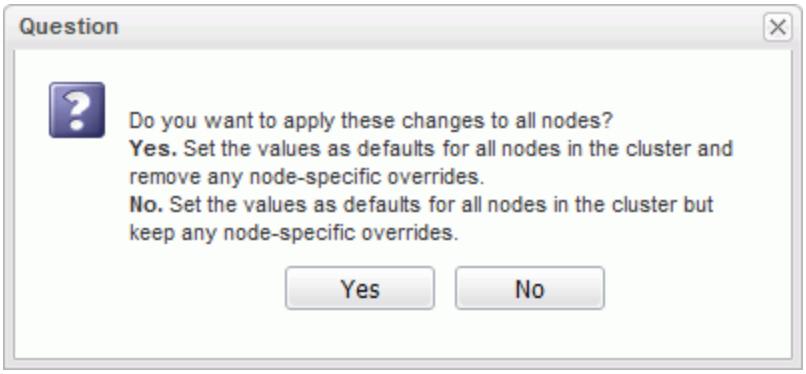
3. Click the **Value** field for the parameter to modify, and then type a new value to replace the existing value.



4. Repeat step 3 for each parameter you want to change, and then click **Save.**

> **Note:** Make sure that you save your changes before you close the Administration Console. Otherwise, any changes you made are lost.

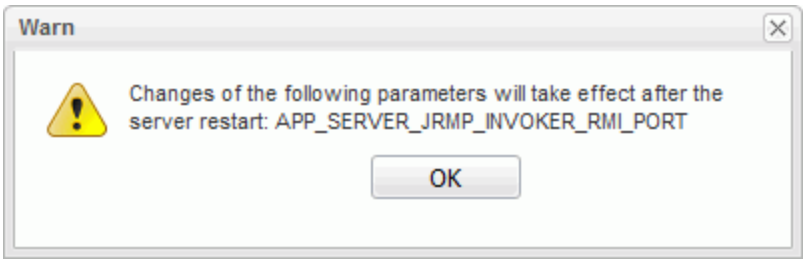
5. (Cluster only) If **Cluster** is the selected scope, and you change the value of a parameter, and then click **Save,** a dialog box opens and gives you the option of applying the changed parameter value across all nodes.



   To apply the new value to all nodes in the cluster, click **Yes.** To retain node-specific overrides for the parameter, click **No.**

6. > **Note:** The **Yes** option works only when you have created a shared folder for the `server.conf` file to give all nodes in the cluster access to the same `server.conf` file.

7. 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.



8. To implement your changes, stop the nodes, run `<PPM_Home>/bin/kUpdateHtml.sh`, and then restart the nodes, one at a time.

> **Note:** For information about how to stop and start PPM Servers, see "Starting and Stopping the PPM Server on a Single-Server System" on page 77. For information about the `kUpdateHtml.sh` script, see "kUpdateHtml.sh" on page 514).

The parameter values that you modify from the Administration Console take effect the next time the parameter values are used.

Configuring and Migrating the PPM Document Management System from the Administration Console

As the PPM administrator, you can use the DMS Configuration tool in the Administration Console to perform the following tasks directly:

The configuration is centralized. That is to say, changing the configuration from the Administrator Console impacts all PPM Server nodes when using clustered configuration. Moreover, no PPM Server restart is required for changes to take effect—the DMS Driver will be reloaded on all PPM Server nodes with the new configuration when you save the updated parameters.

Specify a Different Directory for Your DMS

To specify a different directory as the DMS repository for attachments:

1. In the **Administration Console Actions** panel, select **Administration Task > DMS Configuration.**
2. In the right panel, click **Edit.**

   The Administration Console displays the current DMS solution and the current directory specified as the document repository.

   > **Note:** The **Edit** button is available only when your current DMS is PPM File System.

3. In the **dms.filesys.attachmentDir** box, type the path to the directory you prefer to use as the new repository.

   Note that specifying a different directory path does not migrate the documents stored in the current directory to the new directory.

   If you specify the current DMS parameter to point to an empty DMS, the existing documents in your PPM Center will be lost.

4. Click **Save**.

Migrate Current DMS to a Different Supported DMS

To migrate your existing document management system to a different supported DMS:

1. In the **Administration Console Actions** panel, select **Administration Task > DMS Configuration.**
2. In the right panel, click **Migrate.**
3. From the **Target DMS** list, select the document management system to which you want to migrate your existing DMS data.
4. To test the connection to the selected target DMS, click **Next.**
5. To start the DMS migration, click **Start Migration.**

   The Administration Console displays the progress of the document migration from the PPM Center File System to the target DMS you selected. The progress data displayed includes the number of attachment files that successfully migrated as well as the number that failed to migrate.
6. To see the logs for files that could not be moved to the new DMS, click **View failed files logs.**
7. To try again to migrate unsuccessfully migrated files, click **Retry failed files.**
8. To switch to the PPM DMS, click **Start Transition.**
9. To retire the old DMS and launch the new DMS, click **Commit.**

For more information about migrating PPM DMS, see the Document Management Guide and Reference.

Browsing and Downloading <PPM_Home> Directory Files from the Administration Console

You can browse and download <*PPM_Home*> directory files from the Administration Console with the File Browser tool.

The Administration Console File Browser tool is available to any PPM Administrator with the following access grants:

- Sys Admin: Server Tools: Execute Admin Tools
- Sys Admin: Server Tools: Execute File Browser

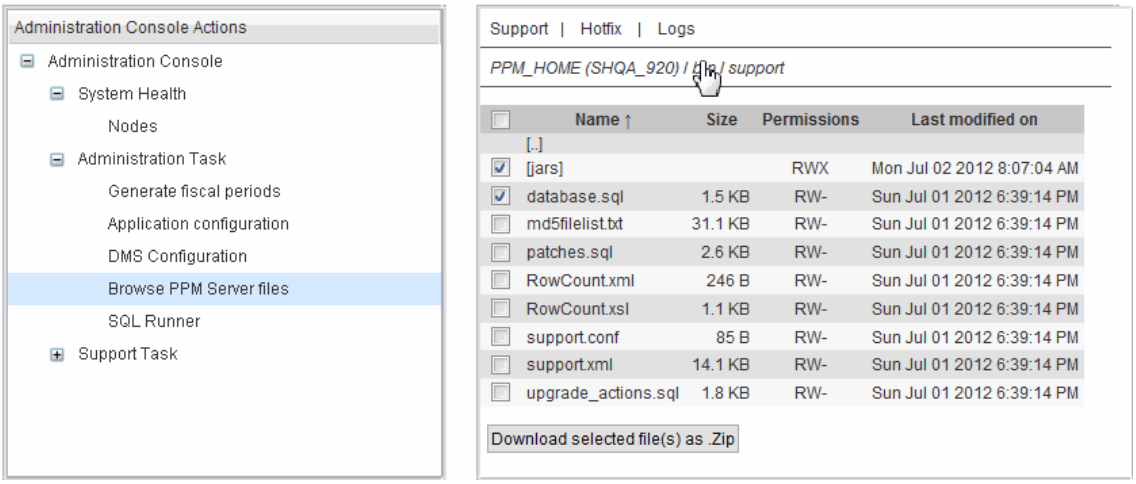
> **Caution:** Users can view contents of all the files in <*PPM_Home*> in read-only mode once they have access to the File Browser, HPE recommends that you provide the Server Tools: Execute File Browser access grant only to selected PPM administrators.

> **Note:** The <*PPM_Home*>/security directory is not accessible through the File Browser and is not listed in the contents of <*PPM_Home*> as it contains the sensitive private key used for SSL encryption. You must connect directly to the PPM Server machine to access this folder.

When initially accessed, the File Browser displays all files and folders located in the <*PPM_Home*> directory:

- Click any file to download it, or click any folder to view its contents.
- When a folder's contents is displayed, you can select one or more files or folders and download them as a Zip file by clicking **Download selected file(s) as .Zip** available at the bottom of the pane.

  The folder hierarchy in the zip file will be rebuilt relatively to the <*PPM_Home*> directory. Empty folders in the Zip file will contain an empty file so that empty folders are not removed automatically from the Zip file. These empty files are created dynamically when the Zip file is generated and are not present in the PPM Application Server file system.

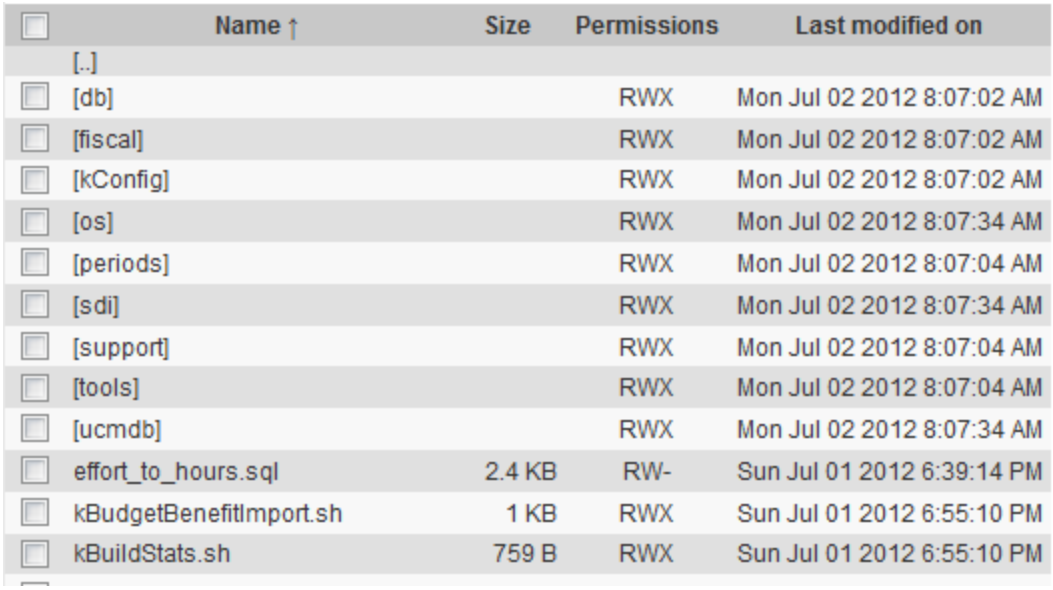


  File information displayed includes name, size, permissions (R for read, W for write, X for execute), and last modification date.

- You can sort the files and folders displayed by clicking a column header. Click twice on the same

column header to reverse the sorting order. A small vertical arrow will be displayed in the column header currently used as the sorting criterion.

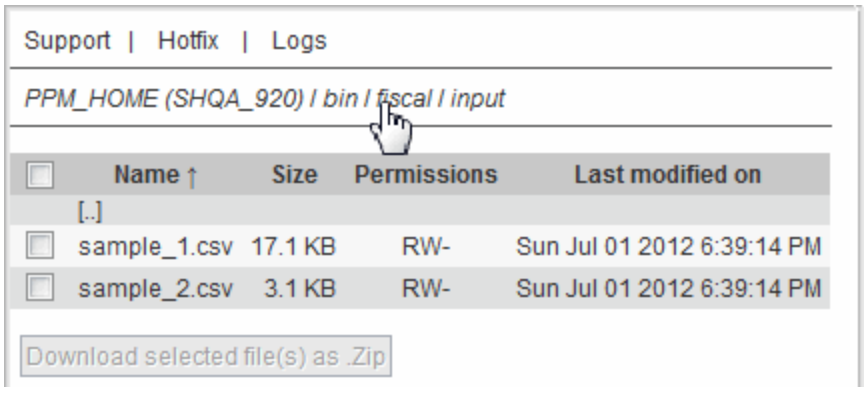
Note that when sorting files and folders, the folders are always displayed before the files, independent of the sorting criterion.



Name ↑	Size	Permissions	Last modified on
[..]			
[db]		RWX	Mon Jul 02 2012 8:07:02 AM
[fiscal]		RWX	Mon Jul 02 2012 8:07:02 AM
[kConfig]		RWX	Mon Jul 02 2012 8:07:02 AM
[os]		RWX	Mon Jul 02 2012 8:07:34 AM
[periods]		RWX	Mon Jul 02 2012 8:07:04 AM
[sdi]		RWX	Mon Jul 02 2012 8:07:34 AM
[support]		RWX	Mon Jul 02 2012 8:07:04 AM
[tools]		RWX	Mon Jul 02 2012 8:07:04 AM
[ucmdb]		RWX	Mon Jul 02 2012 8:07:34 AM
effort_to_hours.sql	2.4 KB	RW-	Sun Jul 01 2012 6:39:14 PM
kBudgetBenefitImport.sh	1 KB	RWX	Sun Jul 01 2012 6:55:10 PM
kBuildStats.sh	759 B	RWX	Sun Jul 01 2012 6:55:10 PM

Also, unless you are viewing the <*PPM_Home*> folder, a [..] folder is displayed at the beginning of the list to let the user access the parent folder.

- To navigate to other folders, click a desired bookmark on top of the File Browser, or click any element of the breadcrumbs representing the current folder (path is relative to <*PPM_Home*>).



Note:

- When PPM is configured in cluster mode, the File Browser only displays the files of the <*PPM_Home*> directory that contain the PPM instance to which you are currently connected to.

- If you want to access another <*PPM_Home*> of the cluster, you must manually connect to a PPM instance hosted in this <*PPM_Home*>> folder.
- When zipping files and folders, you can only select files and folders located in the same folder. If you switch to another folder while some items are selected, the selected items are not included in the generated zip file.

Cluster Configuration Considerations

- When PPM is configured as a server cluster and there are multiple PPM instances in the same <*PPM_Home*>, you can access all the files of all the PPM instances located under the same <*PPM_Home*> folder as the instance you are connected to.
- When selecting **Logs** or **HotFix** links in the bookmarks, you must manually choose the PPM instance you want to access. The folder is automatically displayed when the instance is selected.

Performance Concerns

The files listing can take many seconds if there are many thousands of files to list in the folder which contents are displayed, for example, the <*PPM_Home*>/cache folder, where images of PPM Charts are saved and only deleted after 7 days by default.

Note that there is no significant performance or memory impact on the PPM Server if you decide to download the entire contents of <*PPM_Home*> as a zipped file. This is a safe (though time-consuming) operation, but some temporary files might not be included in the resulting zip file if, for example, they are locked when you try to include them in the Zip file.

Running SQL Queries from the Administration Console

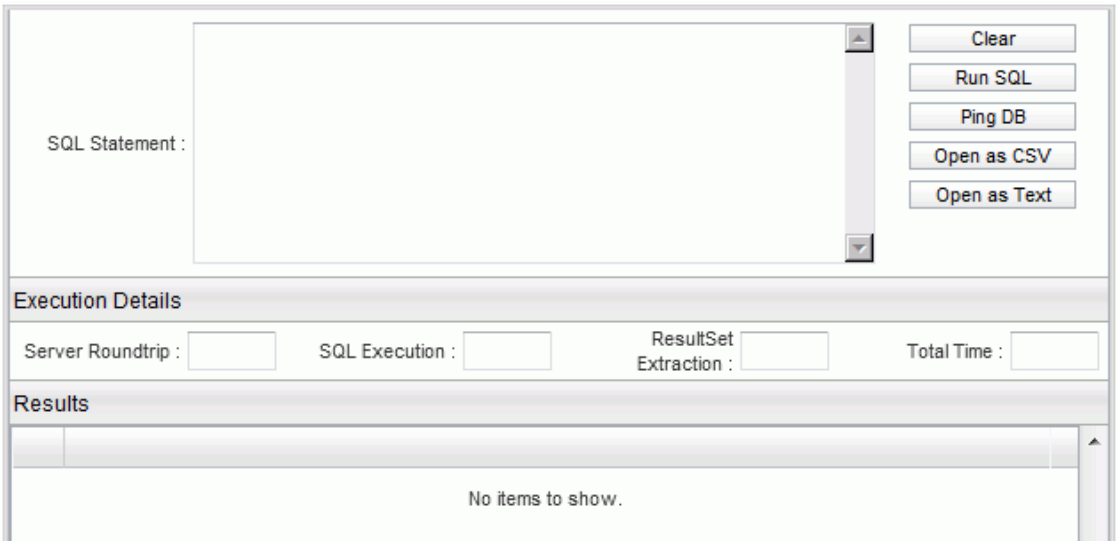
You can run database queries directly against the PPM database schema from the Administration Console.

Note: For information about how to run SQL queries from the PPM Workbench, see "Running SQL Statements from the PPM Workbench" on page 310.

Running SQL statements from the Administration Console is essentially the same as running them from the PPM Workbench. However, from the Administration Console you can run a statement that ends with a semicolon (;).

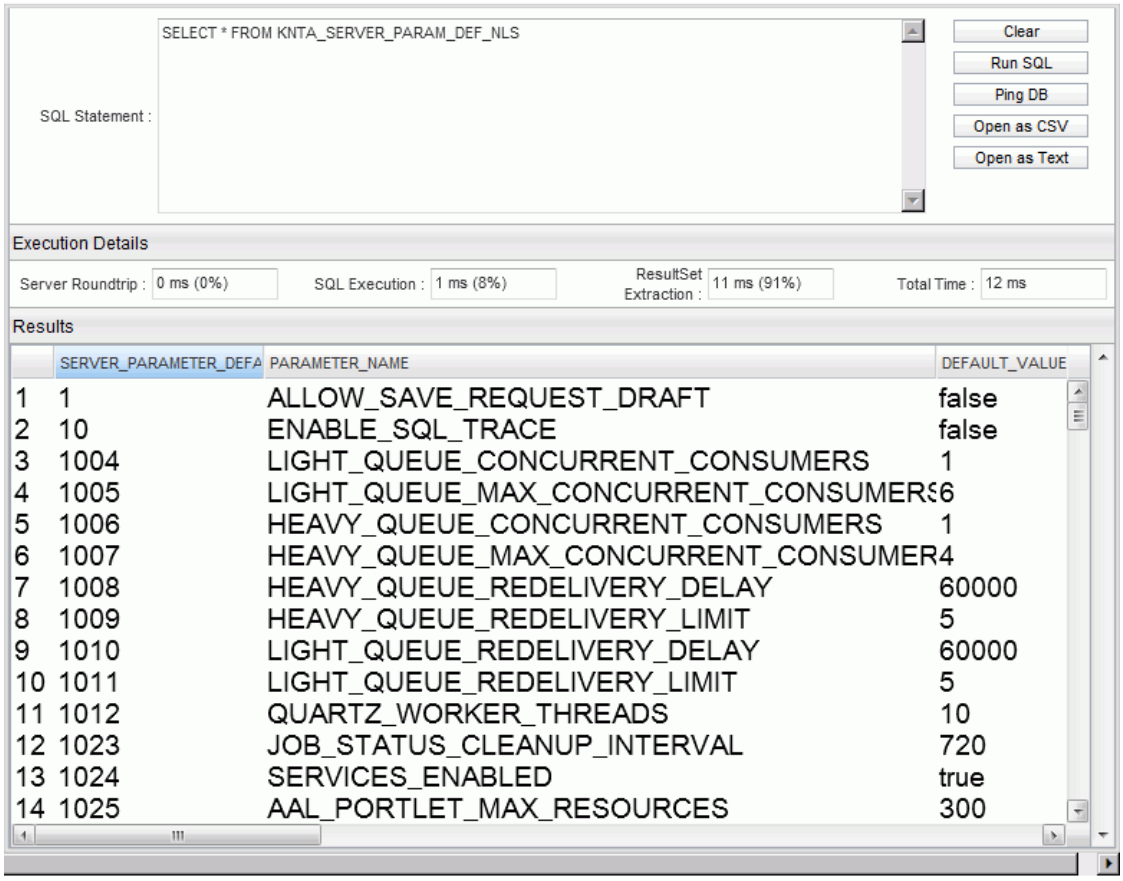
To run a SQL statement (select statement only) from the Administration Console:

1. In the **Administration Console Actions** panel, select **Administration Task > SQL Runner.**
2. In the **SQL Statement** box, type the select statement to run against the PPM database.



3. Click **Run SQL.**

   The **Results** table displays the query results in numbered rows.



4. To test the connection speed between the PPM Server and the PPM database, click the **Ping DB** button.

> **Note:** Because the Administration Console uses a Web Interface, it ignores the connection time between the web client and the PPM Server.

SQL Statement Execution Details

The **Execution Details** section of the SQL runner page displays the following:

- The **Server Roundtrip** value represents the time spent (in milliseconds) between the PPM Server and the PPM database.

> **Note:** In Workbench, the **Server Roundtrip** value represents the time spent between the PPM Workbench client and the PPM Server). In either context, the measure is a good indication of measure network latency.

- The **SQL Execution** value represents the duration of the SQL run in milliseconds.
- The **ResultSet Extraction** value represents the amount of time (in milliseconds) required to extract results from the results set.
- The **Total Time** value represents the amount of time (in milliseconds).

Exporting SQL Run Results

You can export the SQL run results in either comma-separated or text format. If you do export the results, any sorting and grouping you perform from the Administration Console is discarded. Only the raw results are exported.

To export the data to a file in comma-separated format:

1. Click **Open as CSV.**

   The Administration Console displays the message "Do you want to open or save `ppm_sql_<date>_<time>.csv` (<*size*> KB) from `<Server_IP_Address>`?"

2. Click **Open, Save, Save as,** or **Save and open.**

To export the data to a text file:

1. Click **Open as CSV.**

   The Administration Console displays the message "Do you want to open or save `ppm_sql_<date>_<time>.txt` from <*Server_IP_Address*>?"

2. Click **Open, Save, Save as,** or **Save and open.**

Creating a Dashboard Datasource (and List Portlet) from a SQL SELECT Statement in the Administration Console SQL Runner

You can create a dashboard datasource and a list portlet from any SQL statement that runs in the Administration Console SQL Runner.

You can choose whether the dashboard and/or the list portlet is enabled or disabled upon creation. Default choice is enabled, however you might want to disable it if it is created on a production

environment and that some specific access grants should be added to it to restrict its access before it can be used by PPM Users.

Once created from the Administration Console, the datasource can be edited from PPM Workbench (to add filters and access grants) and the list portlet can be edited from the PPM Center portlet definition page, like any datasource or custom list portlet.

To create a dashboard datasource and a list portlet from the Administration Console,

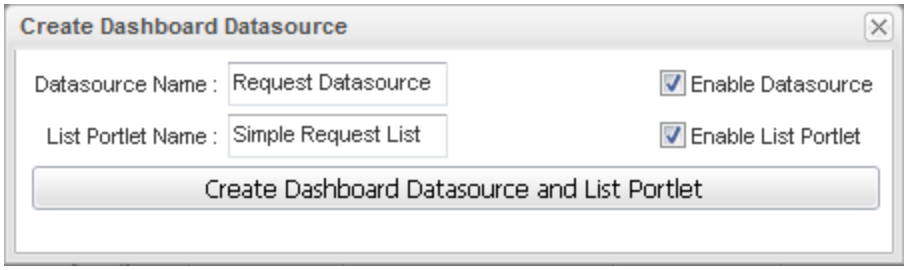
1. Log on to PPM.
2. From the menu bar, select **Open > Administration > Open Administration Console**.
3. Under the Administration Console node, select **Administration Task > SQL Runner**.
4. In the SQL Statement text box, run the SQL statement that you want to create a Dashboard Datasource with.
5. Click the **Create Dashboard Datasource** button.

   The Create Dashboard Datasource dialog opens.
6. Specify names of the datasource and the list portlet that you want to create as described in the table below.

Field/Option	Description
Datasource Name	Specify a name for the datasource that you want to create.
List Portlet Name	Specify a name for the list portlet that you want to create. The List portlet creation is optional. If you keep this field empty, no list portlet will be created, and button name changes accordingly.
Enable Datasource	Select to enable the datasource upon its creation.
Enable List Portlet	Optional. Select to enable the list portlet upon its creation. If you choose to enable the List Portlet, any PPM User can add it directly to their dashboard after the portlet creation.

> **Note:**
>
> - You CANNOT enable the list portlet but disable the datasource.
> - If you enter a name for the list portlet or datasource that already exists, you get an error message.

For example,



7. Click **Create Dashboard Datasource and List Portlet**.

   The Creation Successful confirmation pops up.

8. Click **OK**.
9. If you chose to enable the List portlet, any PPM User can now add it directly to their dashboard.

SQL Syntax Accepted and Limitation

Most of the SQL syntax is accepted as long as it is a `SELECT` statement.

You can use the * operator for columns, such as:

```
SELECT * FROM KNTA_USERS;
```

All columns and their types will be discovered automatically.

However, you cannot use bind variables in the SQL. If you need to use bind variables, first create the datasource with constants values set in place of variables, and then edit the datasource definition in the PPM Workbench to add the bind variables.

Gathering Information for HPE Software Support from the Administration Console

You can run the `kSupport.sh` tool from the Administration Console to gather information about the PPM Server node you currently access with your web browser, and then send that information to HPE Software Support to help diagnosing system problems. This helps to ensure that your issues can be resolved quickly, with minimal requests for information.

The kSupport tool is designed to serve as a troubleshooting knowledge system for PPM Center. Embedding into the tool the knowledge gathered from supporting PPM Center customers around the world, HPE expects the tool to make PPM Center self-diagnosable and self-healable. The flexible

interface of the tool makes it easy to absorb new knowledge as PPM Center develops and HPE Software Support's knowledge grows.

To collect support data for your PPM Server:

1. In the **Administration Console Actions** panel, select **Support Task > Generate Support Information.**
2. Provide the information described in the following table.

   > **Note:** Although none of the fields listed are required, to generate information for HPE Software Support, you must select at least one of the check boxes.

Field	Description
Company Name	Name of your organization
Incident Number	Number HPE has assigned to the incident you reported
System Information	Select this check box to generate log and report information. ◦ **Standard.** Select this option to collect information about the PPM Server<*PPM_Home*>/<*PPM Server*> level) log and PPM Server reports. ◦ **Full.** Select this option to collect information about the following logs: • Upgrade • Install • Deploy (<*PPM_Home*> level) • Server (<*PPM_Home*>/<*PPM Server*> level) • JDBC log **Note:** To see a list of the reports covered, move your cursor over the text "Collect Server (PPM_Home/server level) log and reports."
Super support information	Selecting this check box enables you to search logs by specifying time range and keywords. ◦ **Start time.** Click the Show Date Chooser icon to specify a start time. ◦ **End time.** Click the Show Date Chooser icon to specify an end time. ◦ **Logs search key.** Enter keywords for logs search. If specifying both the time range and the keywords, you can retrieve logs containing the keywords within the time range. This helps quickly locate the desired information from the massive logs and avoid getting outdated logs.

Field	Description
	Note: If the time zone of PPM Server has been changed, the logs cannot be extracted correctly by the specified time range.
Supper support modules	Selecting this check box enables you to gather information based on modules. Then, select specific modules that you want to gather information for. For example, if you want to gather information for the Demand Management module only, you can just select the **DM** checkbox. The retrieved information for the Demand Management module is stored in the *<kSupport_Zip_File>*/etc/DM directory. Note that the check boxes for the **Core**, **DBChangeCheck**, **FilesystemCheck**, and **SeedDataCheck** modules are always selected by default, no matter which modules you select. This means the script is always run for these four modules. Running the script on the **Core** module gathers common information not related to any specific modules. Running the script on the other three mandatorily selected modules collects data that users have modified. For more details, see Collect Modified Data below. The **DBChangeCheck** module is enhanced in 9.30, Running the script on the **Performance** module gathers information about PPM Center performance. For more details and usage of the **ErrorPageCollector** check box, see Collect HTML Source Codes for Error Pages below. In order to proactively detect some common configuration errors, data missing issues, potential data corruption issues, and so on, the kSupport tool can perform some sanity checks for several modules, especially for integration modules. The sanity check is designed based on best practices of PPM Center Support, so it may help find out the causes for some system issues in less time. **Note:** ◦ If an integration module is not enabled, you cannot retrieve support information for that module by selecting the check box for the corresponding module here. ◦ Specifying the time range in the **Super support information** check box section also helps retrieve database queries within the time range. For example, if you specify the time range and select the **DMS** check box, only the integration events within the time range will be retrieved.
Run SQLs	Select this check box to include SQL SELECT statement results in the support information you generate.
Run SQLs	Select this check box to specify either a single SQL file, or a Zip file that contains

Field	Description
from Zip file or single SQL file	multiple SQL files (or subdirectories). Then, click **Browse** to navigate to and select the file(s). **Note: Notes:** ◦ Only `SELECT` statements are retrieved and run. ◦ Make sure that the SQL statements you type in the text box do not exceed 1 MB in size.
Enter your SQL script in the text box below	Select this check box if you want to simply type one or more SQL `SELECT` statements to run. Then, type the statement(s) in the text box provided. **Note:** If you type multiple statements, make sure that you start each statement on a new line and end each statement with a semicolon (`;`).

3. Click **Generate.**

Note: Normally a copy of the `server.conf` file is included in the generated kSupport zip file (in the `<kSupport_zip_file>/` folder).

If your instance is in clustered environment and the `server.conf` file is located in the shared folder for the cluster, after generating the kSupport zip file, you can find a copy of the `server.conf` file in the `<kSupport_zip_file>/ppmc` folder.

Collecting Modified Data

Data that users have modified are collected everytime you run `kSupport.sh` on the following three mandatorily selected modules.

- **DBChangeCheck**. The script compares user's database objects against the baseline data, such as constraints, packages, triggers, and indexes. The retrieved comparison report is stored in the `<kSupport_Zip_File>/etc/DBChangeCheck` directory.

  The constraint comparison report can be found in the `<kSupport_Zip_File>/etc/DBChangeCheck/DBChangeReport.html` file. The report lists the following:

  ◦ Missing primary keys, foreign keys, and unique constraints

  ◦ Custom primary keys, foreign keys, and unique constraints

- **FileSystemCheck**. The script compares the baseline data with user's file system, such as `jsp`, `js`, and class files in the `<PPM_HOME>/server/<NODE>/deploy/itg.war` directory and `<PPM_HOME>/server/<NODE>/deploy/dashboard.war` directory. The retrieved comparison report is

stored in the <*kSupport_Zip_File*>/etc/FileSystemCheck directory, and the modified files are stored in the <*kSupport_Zip_File*>/etc/FileSystemCheck/modifiedfiles directory.

- **SeedDataCheck**. The script compares the baseline seed data with user's seed data, such as request status, workflows, and portlet definitions. The retrieved comparison report is stored in the <*kSupport_Zip_File*>/etc/SeedDataCheck directory.

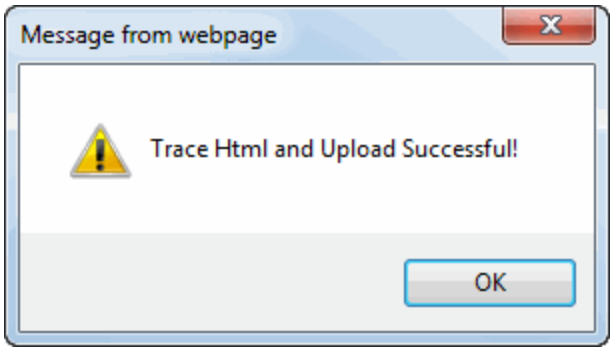
Collecting HTML Source Codes for Error Pages

Collecting HTML source codes for JS errors on PPM Center standard user interface can help address these errors more quickly.

To collect HTML source codes,

1. Log on to PPM Center.
2. Navigate to the page you want to trace.
3. From the menu bar, select **My Links > Trace Html and Upload**.

   A dialog pops up.

4. Click **OK**.
5. Navigate to the Generate Support Information page in the Administration Console.
6. Select **ErrorPageCollector** in the **Super support modules** section.
7. Click **Generate**.

   HTML source codes for error pages are stored in the <*kSupport_Zip_File*>/etc/ErrorPageCollector directory.

Note:

- After you generate support information in the Administration Console, the recently traced error pages will be cleaned.
- Not only error pages but also normal pages can be traced.
- If you are using Internet Explorer 9.0, and open the Developer Tools, you may not be able to trace HTML. To address this issue, do one of the following:
  - Do not use the Developer Tools when tracing HTML.
  - Refresh the page when Developer Tools is opened.

Generating Java Dumps

Starting from version 9.31.0001, you can use the Administration Console to generate Java dumps (heap dumps and thread dumps).

Note: Java dumps can be generated for started PPM server only.

To generate Java dumps in the Administration Console:

1. In the Administration Console Actions panel, select **Support Task** > **Generate Java Dumps**.
2. In the Java Dumps Generator page, click **Generate**.

Note: You can also generate Java dumps by running the command `sh .\kGenJavaDump.sh`. For more information, see "kGenJavaDump.sh " on page 502.

The generated Java dumps are collected in the `<PPM_HOME>\bin\support\javadumps` folder:

- The heap dump files are in the following format:

  `<NODE_NAME>_<PID>_<TIMESTAMP>_heap.hprof`

- The thread dump files are in the following format:

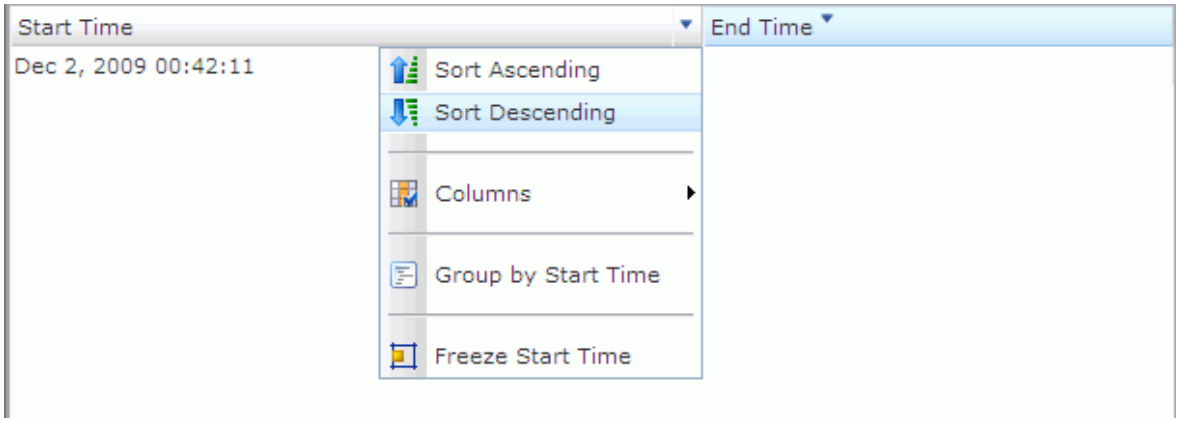
  `<NODE_NAME>_<PID>_<TIMESTAMP>_thread.hprof`

Changing Data Display in Administration Console Tables

This section addresses the various ways you can arrange the data displayed in the right panel of the Administration Console.

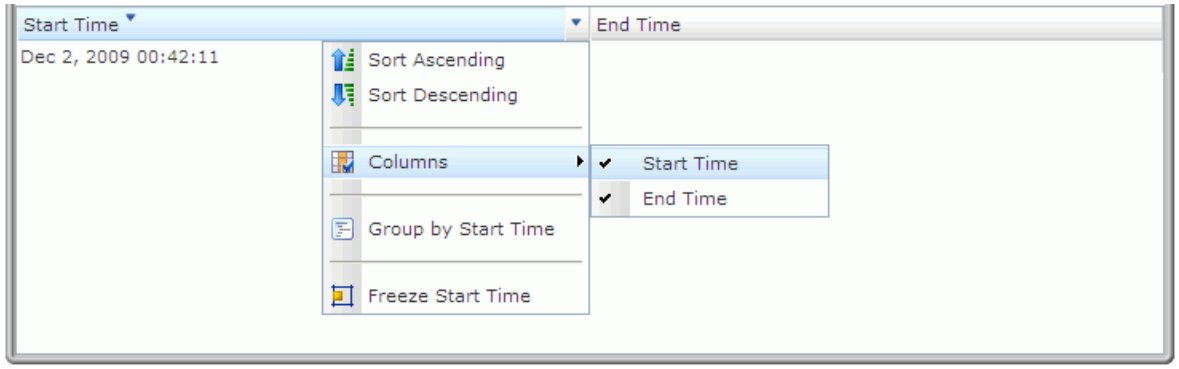
Changing Sort Order

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.**



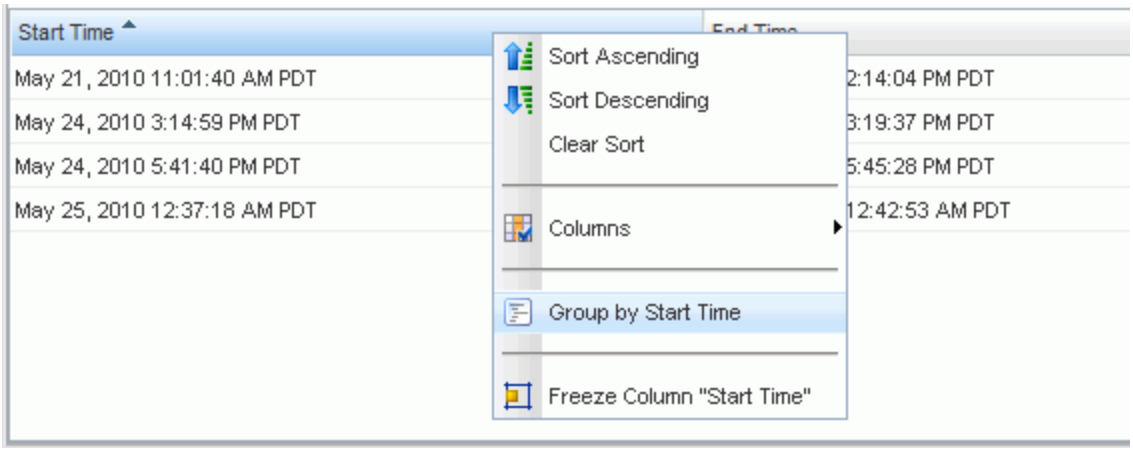
Toggling Column Display

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.



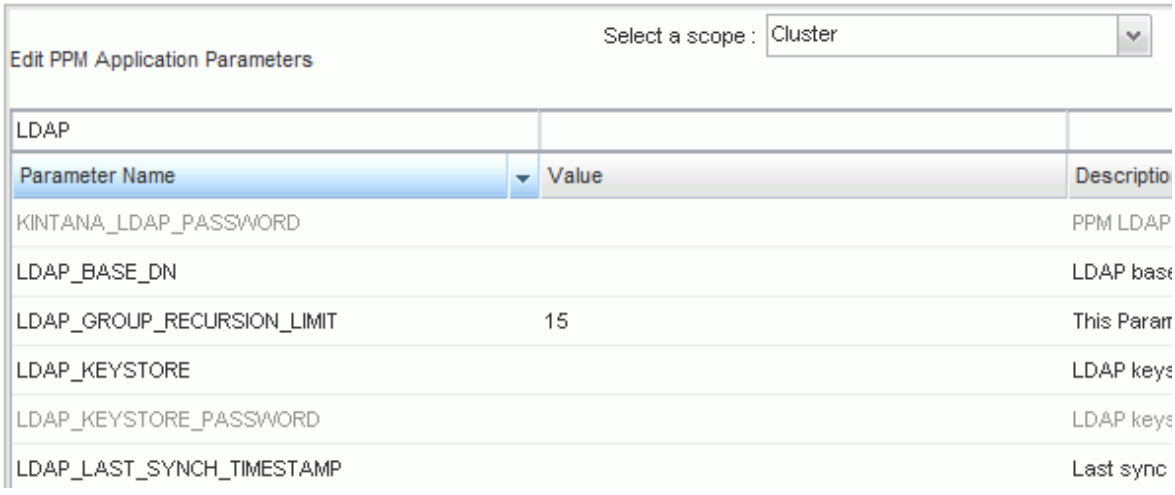
Grouping Displayed Data

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

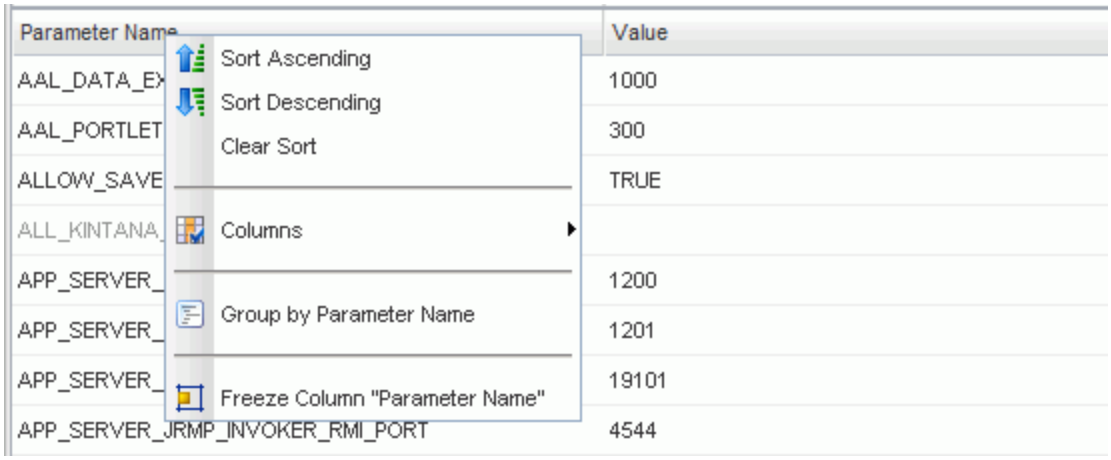
To filter displayed data based on a character string in parameter names, assigned values, or in descriptions, place your cursor in the filter field above a column heading, and then type the text for the filter.



The table displays all data that include the string you specified (specific to the column).

Freezing Column Width

To freeze a 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.



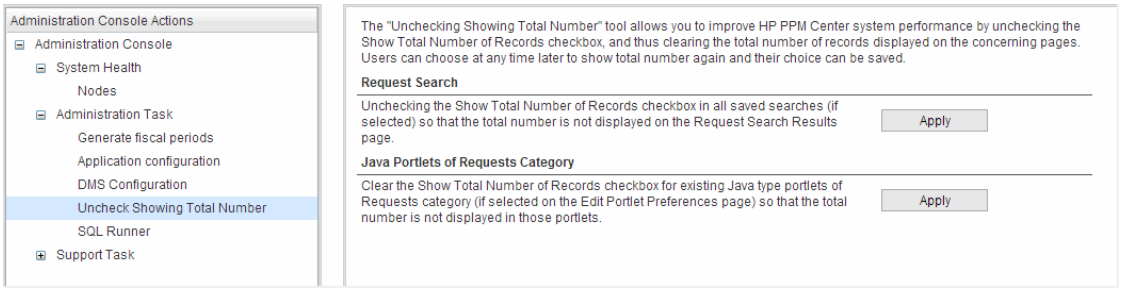
Using the Unchecking Showing Total Number Tool from Administration Console

The Unchecking Showing Total Number tool is added to the Administration Console. It allows you to clear the total number of records displayed on the concerning pages, and thus improve PPM Center system performance.

To use the tool,

1. Log on to PPM Center and launch the Administration Console.
2. In the left navigation pane of the Administration Console, expand the **Administration Task** section, and then select **Unchecking Showing Total Number**.

3. In the right panel, click **Apply** for **Request Search** or **Java Portlets of Requests Category**.



> **Caution:** Be careful when clearing the Show Total Number of Records checkbox. This batch operation is irreversible.

For more information about showing total number, see the *Demand Management User's Guide*.

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

The following table lists the names and descriptions of the three access grants that give users various levels of access to the Server Tools window.

Table 9-2. 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	Stop the PPM Server by using `kStop.sh` when you enable authentication with the REMOTE_ADMIN_REQUIRE_AUTH `server.conf` parameter set to `true`, send messages through `kWall.sh`, execute administration reports in the Admin Tools window, and view the SQL Runner window in the Server Tools Workbench.
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.
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.

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.

Server reports that you can generate from the Admin Tools window

Table 9-3. Server reports

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

Table 9-3. Server reports, continued

Report Name	Description
	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.
CacheManager Statistics	Displays useful statistics on the caching behavior of each cacheable entity in PPM, 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 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 HPE Deployment Management Extensions installed (if any).

Table 9-3. Server reports, continued

Report Name	Description
JVM Memory	Free and total memory in the PPM Server JVM.
Kintana RMI	All RMI connection threads.
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 "Viewing PPM Server Status from the Administration Console" on page 274.
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.

Selecting and running a server report

1. Log on to PPM.
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.
4. Expand the report list in the Admin Tools window and select a report.
5. Click **Submit.**

   The Admin Tools window displays the output of the selected report.

> **Note:** If you run a report on a PPM 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, 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 509.

Running SQL Statements from the PPM Workbench

You can use the SQL Runner window to run database queries directly against the PPM 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.

> **Note:** You can also run SQL queries from the PPM Administration Console.For details, see "Running SQL Queries from the Administration Console" on page 291.

Running 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.

   > **Caution:** 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.**

Controls in the SQL Runner window.

Table 9-4. 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.

Control Name	Control Type	Description
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.

Running an SQL Script with SQL*Plus on a Windows System

If your PPM 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`.

Setting Debugging and Tracing Parameters

Use the Debugging and Tracing Settings dialog box to set debugging and tracing parameters at both the user and server levels.

Opening the Debugging and Tracing Settings dialog box

1. Log on to PPM.
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 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.
   - **Normaldebugging information** is equivalent to the parameter value `INFO`. Errors and information that describes the normal tasks that the running server is performing are logged.
   - **Maximumdebugging 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. 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.

     > **Caution:** 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 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 applications use by calling the Oracle-supplied PL/SQL package `DBMS_PROFILER`.

  > **Note:** 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:" below.

  The profiling information is logged in a JDBC log file in the PPM`log` directory. Enabling the profiler can help you to identify performance bottlenecks.

  > **Note:** Because running the `DBMS_PROFILER` package might slow system performance and reduce storage space, HPE 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 applications) use.

  The output of the profiling information is saved to a JDBC log file in the <*PPM_Home*>/server/<PPM Server>/log directory.

  > **Note:** 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.

   > **Note:** 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.
   - **Normaldebugging information** is equivalent to the parameter value INFO. Errors and information that describes the normal tasks that the running server is performing are logged.
   - **Maximumdebugging 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. 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 "PPM Configuration Parameters" on page 401.

2. To have the PPM Server(s) maintain a Java Database Connectivity (JDBC) log file, select the **Enable JDBC Logging** checkbox.

SQL Debugging for All Product Areas

If you turn on SQL debugging, PPM now collects statistics for the following legacy product areas:

- Demand Management web pages
- Deployment Management web pages

To turn on SQL debugging, set the server parameter `SQL-Debug` to `true`.

The debugging console display includes information about the SQL statements executed by the legacy code. The web pages of the legacy areas now display the debugging console.

System Logging in PPM

Every error message logged in PPM includes a unique identifier that you can use to locate the corresponding error in the log file.

> **Note:** 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

Note: 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 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.

> **Note:** For descriptions of the parameters in the `logging.conf` file (located in the `<`*PPM_Home*`>/conf` directory), see "Logging Parameters" on page 480.

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)

Enabling Debugging On a Per-User Basis

You can turn on the debugging console and set a server logging threshold on a per-user basis. After you do that, a specific user logging on to PPM can toggle the debugging console by pressing and holding the `ALT` key and clicking the HPE logo, which is located above the menu bar.

Enabling debug logging to the serverLog.txt file for a specific user

1. Add the followingjiushi server configuration parameters to the server.conf file and set the values for both to **true** (case-sensitive):
   - ENABLE_DEBUGGING_PER_USER=true
   - SHOW_DEBUGGING_CONSOLE_PER_USER=true
2. (For "Tracing PPM Center Pages with the SQL Tracer Tool" on the next page only) Run the ppm930/sys/GrantSysPrivs.sql script from SQL*PLUS as the SYS DBA user to provide the necessary grants and permissions to the PPM Center users:

   ```
   grant execute on DBMS_MONITOR to &ppm_schema;
   ```
3. To configure finer-grained logging than that specified by the SYSTEM_THRESHOLD parameter, add the following to the logging.conf file:

   USER_THRESHOLD = <*Username*>, <*Log_Level*>

   The following table lists supported log levels.

Log Level	Description
DEBUG	Print fatal, status, error, warning information and debug log (used in debugging)
INFO	Print fatal, status, error, warning and information log
WARN	Print fatal, status, error and warning
ERROR	Print fatal, status, and error log (default)
STATUS	Print fatal and status messages (Not recommended)
FATAL	Print only fatal messages (Not recommended)

Enabling the debug logging to the serverLog.txt file for multiple users

Follow step 1 and step 2 before adding USER_THRESHOLD values on separate lines.

For example:

USER_THRESHOLD = gchu, DEBUG

USER_THRESHOLD = pchapin, DEBUG

USER_THRESHOLD = bkordon, DEBUG

If the log level value meets the threshold criteria of *either* the USER_THRESHOLD (set for that user) or the SYSTEM_THRESHOLD, then the message is logged.

Example 1

```
SYSTEM_THRESHOLD = ERROR
USER_THRESHOLD = ddalton, DEBUG
```

In Example 1, for a user logged in as ddalton, the system logs all DEBUG and higher (INFO, WARN, ERROR, STATUS and FATAL) messages. For any other user, the system logs only ERROR, STATUS and FATAL messages.

Example 2

```
SYSTEM_THRESHOLD = DEBUG
USER_THRESHOLD = ddalton, ERROR
```

In Example 2, the system logs all messages for all users. In this case, the USER_THRESHOLD value (ERROR) set for ddalton has no effect.

Tracing PPM Center Pages with the SQL Tracer Tool

You can trace PPM pages with the SQL tracer tool to help diagnose performance issues caused by poor SQL executions. For example, if you find a slow PPM page, you can turn on this tool to collect information from Oracle side about the SQLs executed on this page and generate support files. With these files, HPE Software Support can better identify the performance bottlenecks and provide suggestions on how to tune the performance.

> **Note:** The SQL tracer tool works by identifying and tracking threads that handle HTTP requests. If there are database actions happening outside the main threads that the Web server uses to handle HTTP requests, those actions are not captured. For example, in the Demand Management module, special commands are handled by threads spawned by request-handling threads, thus they would not be captured by the tool.

To trace SQLs executed on a PPM page,

1. Log on to PPM, and open a page. For example, the Search Request page.
2. Press and hold the ALT key and click the HPE logo located above the menu bar to enable the Debugging Console.

   The Debugging Console opens.
3. Set trace ID.

a. From the right end of the Debugging Console, click the **show** link.

   The DB Stats list displays.

b. Click the latest record from the DB Stats list.

   In this example, click **/itg/web/knta/crt/RequestSearchResults.jsp**.

   The DB Statistics page opens in a new window.

c. Provide a value in the **TraceID** field, and select the **Trace This Page** checkbox.

   Trace ID is the identification you define for collecting information. You can define any words or string as a trace ID, just make sure you include the following characters: alphabetic characters, numbers, and "_".

d. Close the DB Statistics details page.

4. Go back to the page you opened in step 1.

   In this example, go back to the Search Requests page, and run the search again.

5. Get information from the traced page.

   a. Repeat step a and step b of step 3 to reopen the DB Statistics page.

      Now the page displays on-screen instructions for DBAs.

   b. Follow the on-screen instructions to generate support files.

   c. Send the following files that contain the execution plans of SQLs to HPE Software Support.

      - `ppmtrace.trc`
      - `ppmprof.prf`
      - [traceid].xplan

      Before sending the files to HPE Software Support, check and make sure that there are no warnings in the `[traceid].report` file.

      The `ppmtrace.trc` and `ppmprof.prf` files are generated by Oracle commands. For more details, see Oracle documentation (http://docs.oracle.com/cd/B10500_01/server.920/a96533/sqltrace.htm).

      The `ppmxplan.sh` script of the tracer tool generates the `[traceid].xplan` and the `[traceid].report` files. The `ppmxplan.sh` script gets SQL plans from the Oracle shared pools. The `[traceid].report` file contains the information on whether the execution plans of SQLs are fully generated. If you have not run the script timely, the SQL plans might get aged out of the Oracle shared pools. As a result, you might see warnings when you open the `[traceid].report` file.

(Optional) Enabling Debugging Console on Customized Logo

If you have replaced the HPE logo located above the menu bar with a customized one, to enable the Debugging Console, do the following:

1. Make sure the following two files exist.
   - <*PPM_Home*>/server/<*NODE_HOME*>/deploy/itg.war/web/knta/admin/ToggleDebug.jsp
   - <*PPM_Home*>/server/<NODE_HOME>/deploy/itg.war/WEB-INF/jsp/common/Debug.jsp
2. Open your customized `header.jsp` file and add the following Java script to it.

   The `header.jsp` file is located in the <*PPM_Home*>/server/<*NODE_HOME*>/deploy/itg.war/web/knta/global directory. The Java script to be added is as follows:

```
<script type="text/javascript">
    function toggleDebugConsole(event) {
        //try {
            if(!event)
                event = window.event;
            if(!event)
                event = window.Event;

            if(event.altKey) {
                // toggle debug for this user in the user's session
                comLib.doHttpPost('/itg/web/knta/admin/ToggleDebug.jsp');

                // reload the page
                setTimeout("window.location.reload(true);",1000);
            } else {
                location.href = "/itg/dashboard/app/portal/PageView.jsp";
            }
        //} catch(c) {}
        return false;
    }
</script>
```

3. Add Java script event to your logo.

   The Java script event is as follows:

```
<img src="/itg/common/CorporateBrand.gif" style="color: white; cursor:pointer;"
onmousedown="return toggleDebugConsole(event);" alt="<fmt:message
key="CorporateBrand.Name" bundle="${coreResources}" />">
```

4. Include the Debugging Console section in your customized `header.jsp`.

   Locate the following lines:

   ```
   <div class="centerbanner navpathtext" style="width: 100%; padding-top: 5px;">
   <div style="padding-left: 12px;"><%= NavigationPath.getBreadcrumbsHTML(request)
   %></div>
   </div>
   ```

   Add the following script after them:

   ```
   <c:if test="${applicationScope.serverConfig.showDebuggingConsole ||
   sessionScope.showDebuggingConsole}">
     <jsp:include page="/WEB-INF/jsp/common/Debug.jsp" />
   </c:if>
   ```

5. Save and close the `header.jsp` file.

Logging of Physical Memory and Operating System Swap File Space at Server Startup

Total and free physical memory and operating system swap file space are logged during PPM Server startup.

The exception to this is AIX systems, on which this information is not available.

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>/logs` directory. Server log files are named `serverLog.txt` and `serverLog_timestamp.txt`. The log timestamp setting (see "Log TimeStamp Setting" on page 314) 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 HPE 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 HPE Software Support, increase the debug level.

Unless instructed otherwise by HPE 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.

> **Note:** Unless instructed by HPE 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.

Enable HTTP Logging

Caution: 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/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
```

Mail Notification for Specified Server Logs

Mail notification is available for specified server logs. As an administrator, you can decide the server logs information to be notified by email through specifying regular expression or the combination of log levels and regular expression.

To use the mail notification feature, you should first configure the SMTP related parameter in the `server.conf` file and then configure the logging parameters as described in the following table. These logging parameters are added in the `logging.conf` file located in the `<PPM_Home>/conf` directory.

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
ENABLE_SMTP_LOGGING	If set to `true`, mail notification for critical exceptions is enabled.	Default: `false` Valid values: `true, false`
**SMTP.To Required if ENABLE_SMTP_	The recipient of the notification.	Default: N/A Valid values: Email address

LOGGING is set to `true`		
SMTP.From	The sender of the notification. If not specified, the email address of the sender is derived from EMAIL_NOTIFICATION_SENDER in the `server.conf` file.	Default: N/A Valid values: Email address
SMTP.Subject	The subject of the notification mail.	Default: N/A
**SMTP.Filter.RegexToMatch Required if ENABLE_SMTP_LOGGING is set to `true`	The keyword or regular expression to be monitored.	Default: N/A Valid values: Regular expression For example, `OutOfMemoryError|\w+\d{2}|Exception`
SMTP.Filter.LevelMin	The lowest log level to be monitored.	Default: N/A Valid values: `trace/debug/info/warn/error/fatal`
SMTP.Filter.LevelMax	The highest log level to be monitored.	Default: N/A Valid values: `trace/debug/info/warn/error/fatal`
SMTP.delayBetweenChecksInSeconds	The time interval (in seconds) to check message queues.	Default: `10` Valid values: `> 0`
SMTP.SMTPDebug	If set to `true`, enables displaying debug information when mail notification is sent out.	Default: `false` Valid values: `true, false`
SMTP.BufferSize	The buffer queue length.	Default: `512` Valid values: `> 0`

Note:

- The same server log information generated several times within an hour is sent out only once per hour.
- If the information in the `logging.conf` file conflicts with this document, refer to this document for instructions.

Example

If you want to monitor the InfrastructureException issue only and receive emails about the issue, you can set the parameters as follows:

Parameter Name	Parameter Value
ENABLE_SMTP_LOGGING	true
SMTP.To	admin@yourdomain.com
SMTP.From	sender@youdomain.com or null
SMTP.Subject	Notification Report
SMTP.Filter.RegexToMatch	InfrastructureException
SMTP.Filter.LevelMin	debug
SMTP.Filter.LevelMax	fatal

You will receive emails with the content similar to the following:

```
ERROR :ppmLightServiceListenerContainer-1:(ProjectHealthService.java:88):2014/08/15-16:18:38.037 CST: Failed to calculate health for
some projects, rolling back transaction


com.mercury.itg.exceptions.InfrastructureException: org.hibernate.exception.SQLGrammarException: could not execute query
GUID=873B4FFD-23FB-FE54-3117-FB4E35860F06
Generated Time=2014/08/15-16:18:38.036 CST
        at com.mercury.itg.pm.dao.PendingProjectHealthCalculationDAO.getProjectsPending(PendingProjectHealthCalculationDAO.java:42)
        at com.mercury.itg.pm.service.impl.ProjectHealthService.calculateProjectsHealth(ProjectHealthService.java:70)
        at com.mercury.itg.pm.service.impl.ProjectHealthService.runServiceImpl(ProjectHealthService.java:53)
        at com.mercury.itg.core.server.mdServices.PluggableHibernateMDService.runService(PluggableHibernateMDService.java:39)
        at com.mercury.itg.core.jms.service.impl.GenericServiceMessageHandler.handleMessage(GenericServiceMessageHandler.java:71)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:606)
        at org.springframework.aop.support.AopUtils.invokeJoinpointUsingReflection(AopUtils.java:307)
        at org.springframework.aop.framework.ReflectiveMethodInvocation.invokeJoinpoint(ReflectiveMethodInvocation.java:182)
        at org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:149)
        at org.springframework.aop.aspectj.MethodInvocationProceedingJoinPoint.proceed(MethodInvocationProceedingJoinPoint.java:77)
        at com.mercury.itg.core.monitor.impl.BackgroundServiceMonitorAspect.monitorService(BackgroundServiceMonitorAspect.java:104)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:606)
        at org.springframework.aop.aspectj.AbstractAspectJAdvice.invokeAdviceMethodWithGivenArgs(AbstractAspectJAdvice.java:627)
        at org.springframework.aop.aspectj.AbstractAspectJAdvice.invokeAdviceMethod(AbstractAspectJAdvice.java:616)
        at org.springframework.aop.aspectj.AspectJAroundAdvice.invoke(AspectJAroundAdvice.java:64)
        at org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:160)
        at org.springframework.aop.interceptor.ExposeInvocationInterceptor.invoke(ExposeInvocationInterceptor.java:89)
        at org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:171)
        at org.springframework.aop.framework.JdkDynamicAopProxy.invoke(JdkDynamicAopProxy.java:204)
        at com.sun.proxy.$Proxy126.handleMessage(Unknown Source)
        at com.mercury.itg.core.jms.service.impl.ServiceMessageListenerImpl.onServiceMessage(ServiceMessageListenerImpl.java:357)
        at com.mercury.itg.core.jms.service.impl.ServiceMessageListenerImpl.onMessage(ServiceMessageListenerImpl.java:183)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:606)
        at org.springframework.aop.support.AopUtils.invokeJoinpointUsingReflection(AopUtils.java:307)
        at org.springframework.aop.framework.JdkDynamicAopProxy.invoke(JdkDynamicAopProxy.java:198)
        at com.sun.proxy.$Proxy128.onMessage(Unknown Source)
        at
org.springframework.jms.listener.AbstractMessageListenerContainer.doInvokeListener(AbstractMessageListenerContainer.java:543)
        at
org.springframework.jms.listener.AbstractMessageListenerContainer.invokeListener(AbstractMessageListenerContainer.java:482)
        at
org.springframework.jms.listener.AbstractMessageListenerContainer.doExecuteListener(AbstractMessageListenerContainer.java:451)
        at
org.springframework.jms.listener.AbstractPollingMessageListenerContainer.doReceiveAndExecute(AbstractPollingMessageListenerContainer.j
        at
org.springframework.jms.listener.AbstractPollingMessageListenerContainer.receiveAndExecute(AbstractPollingMessageListenerContainer.jav
        at com.mercury.itg.core.jms.service.impl.ServiceMessageListenerContainer.access$2000(ServiceMessageListenerContainer.java:40)
        at
com.mercury.itg.core.jms.service.impl.ServiceMessageListenerContainer$AsyncMessageListenerInvoker.invokeListener(ServiceMessageListene
        at
com.mercury.itg.core.jms.service.impl.ServiceMessageListenerContainer$AsyncMessageListenerInvoker.executeOngoingLoop(ServiceMessageLis
        at
com.mercury.itg.core.jms.service.impl.ServiceMessageListenerContainer$AsyncMessageListenerInvoker.run(ServiceMessageListenerContainer.
        at java.lang.Thread.run(Thread.java:724)
Caused by: org.hibernate.exception.SQLGrammarException: could not execute query
        at org.hibernate.exception.SQLStateConverter.convert(SQLStateConverter.java:67)
        at org.hibernate.exception.JDBCExceptionHelper.convert(JDBCExceptionHelper.java:43)
        at org.hibernate.loader.Loader.doList(Loader.java:2223)
        at org.hibernate.loader.Loader.listIgnoreQueryCache(Loader.java:2104)
        at org.hibernate.loader.Loader.list(Loader.java:2099)
        at org.hibernate.loader.hql.QueryLoader.list(QueryLoader.java:378)
        at org.hibernate.hql.ast.QueryTranslatorImpl.list(QueryTranslatorImpl.java:338)
        at org.hibernate.engine.query.HQLQueryPlan.performList(HQLQueryPlan.java:172)
        at org.hibernate.impl.SessionImpl.list(SessionImpl.java:1121)
        at org.hibernate.impl.QueryImpl.list(QueryImpl.java:79)
        at com.mercury.itg.pm.dao.PendingProjectHealthCalculationDAO.getProjectsPending(PendingProjectHealthCalculationDAO.java:40)
        ... 44 more
Caused by: java.sql.SQLSyntaxErrorException: ORA-00904: "PENDINGPRO0_"."ATTEMPT_COUNT": invalid identifier

        at oracle.jdbc.driver.SQLStateMapping.newSQLException(SQLStateMapping.java:91)
        at oracle.jdbc.driver.DatabaseError.newSQLException(DatabaseError.java:112)
        at oracle.jdbc.driver.DatabaseError.throwSqlException(DatabaseError.java:173)
        at oracle.jdbc.driver.T4CTTIoer.processError(T4CTTIoer.java:455)
        at oracle.jdbc.driver.T4CTTIoer.processError(T4CTTIoer.java:413)
        at oracle.jdbc.driver.T4C8Oall.receive(T4C8Oall.java:1030)
        at oracle.jdbc.driver.T4CPreparedStatement.doOall8(T4CPreparedStatement.java:194)
        at oracle.jdbc.driver.T4CPreparedStatement.executeForDescribe(T4CPreparedStatement.java:785)
        at oracle.jdbc.driver.T4CPreparedStatement.executeMaybeDescribe(T4CPreparedStatement.java:860)
        at oracle.jdbc.driver.OracleStatement.doExecuteWithTimeout(OracleStatement.java:1186)
        at oracle.jdbc.driver.OraclePreparedStatement.executeInternal(OraclePreparedStatement.java:3381)
        at oracle.jdbc.driver.OraclePreparedStatement.executeQuery(OraclePreparedStatement.java:3425)
        at oracle.jdbc.driver.OraclePreparedStatementWrapper.executeQuery(OraclePreparedStatementWrapper.java:1490)
        at org.jboss.resource.adapter.jdbc.WrappedPreparedStatement.executeQuery(WrappedPreparedStatement.java:236)
        at org.hibernate.jdbc.AbstractBatcher.getResultSet(AbstractBatcher.java:186)
        at org.hibernate.loader.Loader.getResultSet(Loader.java:1787)
        at org.hibernate.loader.Loader.doQuery(Loader.java:674)
        at org.hibernate.loader.Loader.doQueryAndInitializeNonLazyCollections(Loader.java:236)
        at org.hibernate.loader.Loader.doList(Loader.java:2220)
        ... 52 more
```

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 HPE 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 HPE Software Support, you can delete these log files on a regular basis.

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.

- "Changing PPM Data" below
- "Changing the Database Schema Passwords" on the next page
- "Maintaining Temporary Tables" on page 332

Changing PPM Data

Updating PPM master data directly in the database can cause various errors to occur. HPE highly recommends that you not make changes directly to the PPM database, and instead use the PPM 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 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 data model, see the *Data Model Guide*.)

Changing the Database Schema Passwords

If you must change the PPM database schema passwords, be sure to change them both in the database and in the `server.conf` file.

What to consider before you change all the database schema passwords

- 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.

  > **Note:** 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.

Changing the PPM 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 a Single-Server System" on page 77.)
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.

   > **Note:** 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
- User sessions
- 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 235 and in "PPM Configuration Parameters" on page 401) 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.

PPM_LOGON_SESSIONS

The PPM_LOGON_SESSIONS table stores information about the sessions created on the PPM server. It includes information such as session start and end time, session type, how uses logged on to PPM, PPM node on which the session was created, and how the session ended.

The data in the table is automatically purged by the Logon Attempts Cleanup service after the duration specified by the parameter `DAYS_TO_KEEP_LOGON_SESSIONS_ROWS`. By default, the value of the parameter is `60`, which means the table contains information about user sessions during the previous 60 days.

KNTA_DEBUG_MESSAGES Table

The `KNTA_DEBUG_MESSAGES` table contains any debugging text that HPE 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.

Purging Stale PPM Database Data

The increasing PPM database size is starting to impact database operations and affect the overall performance of PPM. This is especially true for organizations that have been using PPM for years.

This section provides complete information about the PPM Purge Tool, including an introduction of the tool, installation instructions, and usage information.

Overview of the PPM Purge Tool

The standalone PPM Purge Tool is designed only for the users who have both the SYS DBA and PPM Center application administrator access grants to permanently delete (purge) stale database data by specifying purging criteria.

For security reasons, it is highly recommended that this tool should be installed on a dedicated server that only the tool users with both the SYS DBA and administrator access grant can have access to, rather than on an end-user's machine.

The PPM Purge Tool offers the following:

- Dynamic statistics overview of all entities with statuses in the PPM Oracle database
- Enables administrators to define purging criteria based on their organizations business needs and to purge stale data to decrease database load
- Downloadable XML files for purging criteria that administrators defined and downloadable purge history reports

The PPM Purge Tool is located here: `<PPM_Home>/bin/purge/ppm-purge.zip`.

The `ppm-purge.zip` package contains the following:

- `<purge_home>/kPurgeStart.bat`: Execute this script to launch the PPM Purge Tool web server on a Windows system.
- `<purge_home>/kPurgeStart.sh`: Execute this script to launch the PPM Purge Tool web server on a Unix system.
- `<purge_home>/jetty` folder, contains all necessary scripts and files for the tool, including the `ppminfo.conf` file.

> **Note:** The PPM Purge Tool can be accessed from an IPv4 address only. It does not support a JDBC connection using IPv6 URL.

Prerequisites for Running the PPM Purge Tool

Review the following prerequisites:

- **The purge operation is irreversible. Make sure you back up the database before you run the purge tool.**
- Before you run the PPM Purge Tool, stop the PPM Server. If in clustered configuration, stop all PPM Server nodes.
- Collect business needs from your organization's stakeholders in order to clearly define your purge criteria.

Purge Stale Data Using the PPM Purge Tool

This section provides detailed instructions on how to use the PPM Purge Tool to purge stale database data.

To purge stale data,

1. Copy and extract the `<PPM_Home>/bin/purge/ppm-purge.zip` package to a different server than the DB server.

   For security reasons, it is highly recommended that this tool should be installed on a dedicated server that only the tool users with both the SYS DBA and administrator access grants can have access to, rather than on an end-user's machine.

2. (Required if you use BusinessObject on PPM) Stop the capture process of Oracle Streams.

   For information about how to stop a capture process, refer to https://docs.oracle.com/cd/B28359_01/server.111/b28324/tdpii_adcont.htm.

3. Stop the PPM Server. If in clustered configuration, stop all nodes in the cluster.

4. Set the JAVA_HOME value.

   - To set the JAVA_HOME value in Windows,

     > **Note:** The steps described in the following procedure are for Windows 7. The exact steps may differ, depending on your Windows operating system.

     i. Open the Control Panel.

     ii. Open the System window.

     iii. Click **Advanced system settings** in the navigation pane.

     The System Properties dialog box opens.

     iv. Click **Environment Variables** on the Advanced tab.

     v. Under **System Variables,** click **New.**

     The New System Variable dialog box opens.

     vi. In the **Variable name** box, type `JAVA_HOME.`

     vii. In the **Variable Value** box, type the full Java install directory path, and click **OK**.

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

     ```
     set JAVA_HOME=<JDK_Home>
     ```

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

     ```
     JAVA_HOME=<JDK_Home>; export JAVA_HOME
     ```

   For more information about setting JAVA_HOME value, see the *Preparing to Install PPM Center* section in the *Installation and Administration Guide*.

5. Navigate to the `<purge_home>` directory that you extracted in step 1, and start the purge tool

server.

Run the following script to star the purge tool server:

- On Windows system, run the `kPurgeStart.bat` file
- On Unix system, run the `kPurgeStart.sh` file

6. Open a web browser, type **`http://localhost:8080`** or a specified IP address with access control in the **Address** bar, and press **Enter**.

   > **Note:**
   > - HPE recommends you use domain name to access the Purge Tool web server, instead of using IP address.
   > - The default port is `8080`. To use a different port, you can configure the port value in the `<ppm_purge_home>/jetty/start.d/http.ini` file.
   > - By default, the Purge Tool allows access from the localhost only. However, you can control whether you allow remote access to the Purge Tool or not. For more information, see "Restrict Remote Access to the PPM Purge Tool to Specified IP Addresses" on page 342.

   The PPM Purge Tool login page opens in your browser.

7. Configure PPM Server and Oracle database related parameters.

   Go to the `<ppm_purge_home>/jetty/conf` directory and configure the parameters in the `ppminfo.conf` file as described in the table below.

   > **Note:** Only the DB account configured in the `ppminfo.conf` file has access to the PPM Purge Tool.

Parameter name	Description	Sample Value
DB_USERNAME	Username of the PPM database schema for connecting the Oracle database. It is usually the same as the value of the DB_USERNAME parameter in the `server.conf` file.	DB_USERNAME= `knta`

RMI_URL_LIST	RMI URL of the PPM Server. When there are multiple PPM Server nodes, the RMI URLs are separated by commas.	RMI_URL=
		`rmi://localhost`
		`:1099`
		`/KintanaServer`

Parameter name	Description	Sample Value
	See the RMI_URL parameter value in the `server.conf` file.	
JDBC_URL	JDBC URL for the Oracle database. See the JDBC_URL parameter value in the `server.conf` file.	JDBC_URL= `jdbc:oracle:thin :@localhost:1521 :ORCL`

8. Provide the DB account information you just configured in the **DB User Name** and **DB Password** fields.
9. Click **Login**.

   The PPM Purge Tool opens to its **Database Statistics Overview** tab.

   The Database Statistics Overview tab dynamically displays the current statistics of all entities found in the Oracle database.

   There are two date types available for all the entities: **Creation date** (default) and **Update date**. You can choose to display the database statistics overview by Creation Date or Update date.

   The entities found in the Oracle database are categorized into two statuses for simplification purpose: **Open** and **Closed**. The following table describes specific statuses categorized as Closed or Open for each entity type:

Entity Type	Closed	Open
Request (status)	◦ Cancelled ◦ Closed ◦ Closed (Approved) ◦ Closed (Not Approved) ◦ Closed - Failed ◦ Closed - Rejected ◦ Closed – Successful	All the other Request statuses
Project (status)	◦ Cancelled ◦ Closed ◦ Closed (Approved) ◦ Closed (Not Approved) ◦ Closed - Failed ◦ Closed - Rejected ◦ Closed – Successful	All the other Project statuses
Staffing Profile (status)	◦ Completed ◦ Cancelled	All the other Staffing Profile statuses

Entity Type	Closed	Open
Program (status)	◦ Completed ◦ Cancelled	All the other Program statuses
Scenario Comparison (active_flag)	N/A	All the other Scenario Comparison statuses
Timesheet (status)	◦ CANCELLED ◦ CLOSED	All the other Timesheet statuses
Package (status)	◦ Cancelled ◦ Closed [Failure] ◦ Closed [Mixed] ◦ Closed [Success]	All the other Package statuses
Release (status)	Closed	All the other Release statuses
Portfolio	N/A	N/A

10. Go to the **Run PPM Purge Tool** tab.

    You can define and add your purge criteria by specifying desired values for the fields on this page. You can add as many criteria as you like.

    > **Note:** In case there are any specific data that you want to retain, go to the **White List** tab to specify your retention criteria. Otherwise leave the **White List** tab empty.

11. On the **Purge Criteria** tab, define system level or entity level purging criteria as described in the following table.

Field	Description
Date type	Specify date type for all entities in the Oracle database by selecting **Creation Date** or **Update date** from the **Date type** drop-down list.
Criteria	Allows you to define a system level or an entity level purging criteria: To define a system level purging criterion, select **All**. To define an entity level purging criteria, select an entity from the list of supported entities: ◦ Request ◦ Project ◦ Timesheet ◦ StaffingProfile

Field	Description
	◦ Program ◦ ScenarioComparison ◦ Package ◦ Protfolio ◦ Release To specify purging criteria for multiple entities, you need to add them one at a time.
Entity Type	Allows you to select **All** or a specific entity type (if available) for the entities you specify in the **Criteria** field: **Request**, **Project**, **Program**, or **Package**. Note that the available entity type options vary with the entity you select. **Note:** This field is not applicable for the following entities: **Timesheet**, **StaffingProfile**, **ScenarioComparison**, **Protfolio**, and **Release**. To specify multiple entity types for an entity, you need to add them one at a time.
Status	Allows you to multi-select statuses for the entity type you want to purge. The available statuses vary with entities you specify in the **Criteria** field. You can click **Check all** or **Uncheck all** to select or deselect all options. **Note:** The **Status** field is not applicable to the **Portfolio** entity. The **Status** field is not applicable when you select **All** in the **Entity Type** field.
Date Before	Click the calendar icon to select a date. This defines that you want to purge data with their creation date or update date earlier than the specified date (not including data created or updated on the selected date). You can also manually enter a valid date value.

12. After you have specified a criterion, click **Add**.

    To add multiple entities and/or entity types, you need to add them one at a time. The screenshot below illustrates an example of multiple purging criteria being added.

    Note: If you add a criteria that was already added, the new one will override the existing one.

13. If there are any specific data that you want retain from the purging criteria you already specified, go to the **White List** tab and specify the retention criteria.

    Otherwise, leave the **White List** tab empty.

14. Review the purging criteria and click **Next**.

    The purge tool starts to identify and add tags to purgeable data.

15. When the purge tool finished identifying and tagging purgeable data, cilck **Next**.

    The summary page displays an overview of purgeable entities and non-purgeable entities.

    Click **View Details** to view entities that have dependencies.

16. (Optional) Run some SQL scripts in the database to view detailed lists of purgeable entities.

    Detailed lists of purgeable entities are not displayed here due to potential large amount of the purgeable entities. However, you can run some SQL scripts in the database to view the detailed lists.

    > **Note:** Skip this step if you do not need to view the detailed lists of purgeable entities.

    a. Log on to the database.

    b. Run SQL scripts as described in the table below to view the detailed list of purgeable entities for a concerning entity type.

Entity Type	Run the following script
Request	`SELECT * FROM KCRT_REQUESTS T1 WHERE EXISTS(SELECT 1 FROM P_REQUEST T2 WHERE T1.REQUEST_ID=T2.REQUEST_ID)`
Project	`SELECT * FROM PM_PROJECTS T1 WHERE EXISTS(SELECT 1 FROM P_PROJECT T2 WHERE T1.PROJECT_ID=T2.PROJECT_ID)`
Timesheet	`SELECT * FROM TM_TIME_SHEETS T1 WHERE EXISTS(SELECT 1 FROM P_TIME_SHEET T2 WHERE T1.TIME_SHEET_ID=T2.TIME_SHEET_ID)`
Staffing Profile	`SELECT * FROM RSC_STAFFING_PROFILES T1 WHERE EXISTS(SELECT 1 FROM P_STAFFING_PROFILE T2 WHERE T1.STAFFING_PROFILE_ID=T2.STAFFING_PROFILE_ID)`
Financial Summary	`SELECT * FROM FM_FINANCIAL_SUMMARY T1 WHERE EXISTS(SELECT 1 FROM P_FINANCIAL_SUMMARY T2 WHERE T1.FINANCIAL_SUMMARY_ID=T2.FINANCIAL_SUMMARY_ID)`
Portfolio	`SELECT * FROM PFM_PORTFOLIOS T1 WHERE EXISTS(SELECT 1 FROM P_PORTFOLIO T2 WHERE T1.PORTFOLIO_ID=T2.PORTFOLIO_ID)`
Program	`SELECT * FROM PGM_PROGRAMS T1 WHERE EXISTS(SELECT 1 FROM P_PROGRAM T2 WHERE T1.PROGRAM_ID=T2.PROGRAM_ID)`
Release	`SELECT * FROM KREL_RELEASES T1 WHERE EXISTS(SELECT 1 FROM P_RELEASE T2 WHERE T1.RELEASE_ID=T2.RELEASE_ID)`

Entity Type	Run the following script
Package	`SELECT * FROM KDLV_PACKAGES T1 WHERE EXISTS(SELECT 1 FROM P_PACKAGE T2 WHERE T1.PACKAGE_ID=T2.PACKAGE_ID)`

17. Go back to the pre-processing summary page of the PPM Purge Tool, click **Next**.

    The purge tool moves to **Step 3. Start Purge** page.

18. Follow the screen instructions to create the PPM_PURGE_DIR directory and grant read/write privileges to it.

    The PPM_PURGE_DIR directory will hold the external tables that the purge tool created on each run to store the purged data. For example, you can run the following to replace the `/home/oracle/oracle11g/product/11.2.0/dbhome_1/backup` directory with your own directory on the Oracle database server:

    ```
    CREATE DIRECTORY PPM_PURGE_DIR AS
    '//home/oracle/oracle11g/product/11.2.0/dbhome_1/backup';

    GRANT READ, WRITE ON DIRECTORY PPM_PURGE_DIR TO;
    ```

19. Check that PPM server or all nodes in the PPM cluster are stopped, and provide comments for the current purge operation in the **Messages** text box.

20. Click **Start Purging**.

    Wait for the current purging operation to finish.

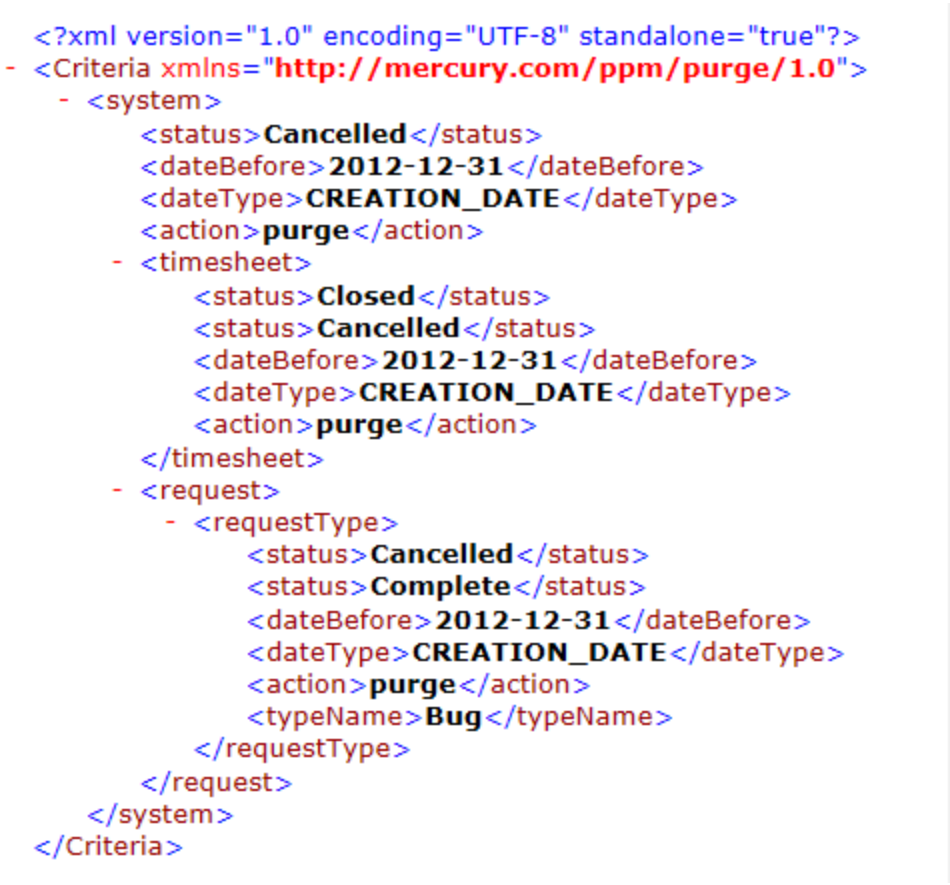
    > **Caution:** HPE strongly recommends you not to click **Logout** or move to other tab pages while the purging is in progress.

21. Click **Next** when purging finishes.

    The Purge results <*Purge_ID*> page displays.

22. To view or download historical purge criteria and summary reports, go to the **Purge History** tab.

    By clicking **Download XML**, you can download an XML file that contains all purge criteria that you or another DB administrator specified for that purge operation. You can find the XML file as follows: `C:/Users/<User>/Downloads/purgeCriteria_<purge_ID>.xml`. Below is the content of the XML file for purge ID 30020:



By clicking **Summary Report**, you can view a snapshot of the database statistics for a purge operation, like the following.

23. (Required if you use BusinessObject on PPM) After the purging is done, perform a full ETL.

    For information about how to perform a full ETL, see the *Operational Reporting Administrator's Guide*

Restrict Remote Access to the PPM Purge Tool to Specified IP Addresses

If you need to access the PPM Purge Tool remotely, you may want to restrict remote access to the PPM Purge Tool to specified IP addresses .

1. Go to the `<purge_home>/jetty/webapps/ROOT/WEB-INF` directory.
2. Open the `web.xml` file.

3. Add IP addresses you would allow remote access to the purge tool to the `<param-value>` line. The IP addresses shall be separated by commas.

   For example, if you want to allow remote access from the IP address of `126.1.1.1`, add it behind the default value for the `<param-value>` line as follows:

```
<filter>
    <filter-name>RemoteIpFilter</filter-name>
    <filter-class>com.kintana.purge.filter.RemoteIpFilter
</filter-class>
    <init-param>
        <param-name>allow</param-name>
        <param-value>127\.0\.0\.1,0\:0\:0\:0\:0\:0\:0\:1,126\.1\.1\.1</param-
value>
    </init-param>
</filter>
```

4. Save the file.

> **Note:** The default value of **127\.0\.0\.1,0\:0\:0\:0\:0\:0\:0\:1** for `<param-value>` means support for IPv4 and IPv6 localhosts.

Tracking User Sessions in PPM Using Database Table

Using the database table PPM_LOGON_SESSIONS, you can track the following information of users sessions created in PPM:

- Session start and finish time
- Session type
- How users logged on to PPM
- PPM node on which the session was created
- How the session ended

> **Note:** The table PPM_LOGON_SESSIONS does not record information when users fail to authenticate.

The data in the table is automatically purged by the Logon Attempts Cleanup service after the duration specified by the parameter `DAYS_TO_KEEP_LOGON_SESSIONS_ROWS`. By default, the value of the

parameter is `60`, which means the table contains information about user sessions during the previous 60 days.

For more information about this table, see the *Data Model Guide*.

> **Note:** PPM_LOGON_SESSIONS can be used in combination with the table KNTA_LOGON_ATTEMPTS. For example, if you want to find out the IP address of a user when connecting to PPM, refer to KNTA_LOGON_ATTEMPTS.

Backing Up PPM Instances

Backing up a PPM instance involves backing up both the file system and the database schema. HPE stores all PPM configuration and transaction data in its associated database schema.

Because this information is so important, HPE 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.

HPE recommends that you back up the `<PPM_Home>/logs` directory daily. This directory contains transactional history files for each migrated package or request.

> **Note:** Before you make critical changes to PPM, perform a full backup of the database schema and complete `<PPM_Home>` directory.
>
> It is not necessary to back up registry settings.

Protecting Backups

Because the backups you create may contain sensitive information such as cryptographic keys and payload data, HPE strongly advises that you protect the backups themselves. Oracle Advanced Security provides transparent data encryption of data stored in the database, the encryption of disk-based backups of the database, and network encryption for data traveling across the network between the database and client or mid-tier applications. It also provides a suite of strong authentication services to Oracle Database.

To use Enterprise User Security in Oracle Database Enterprise Edition, you must license Oracle Internet Directory (OID). If you want to use stronger authentication alternatives (such as Kerberos or PKI) for enterprise user security, you must license Oracle Advanced Security and the Oracle Internet Directory (OID). For more information, see the release notes for your Oracle software.

Checking PPM License Status

You can view information related to licenses on your organization's PPM instance in the Administration Console. The License tool provides the following licensing information:

- Which PPM products are licensed for use on your instance
- Expiration dates for licenses
- Number of licenses available for different PPM modules.

For more information, see "View A Summary of Autopass Licenses Purchased and Installed on the PPM Server " on page 276.

Compiling JSP Files at Runtime

If you have made changes to JSP files and want to make modified JSP files reloadable at runtime, do the following:

1. Back up your `server.conf` file, which is located in the <*PPM_Home*> directory.
2. Run the `kStop.sh` script to stop the PPM Server:

   ```
   sh ./kStop.sh
   ```
3. Open the `server.conf` file in a text editor, and set `com.kintana.core.server.JSP_RECOMPILE_ENABLED` to `true`.
4. Specify folders to exclude when compiling the modified JSP files.

   For example,

   ```
   com.kintana.core.server.JSP_COMPILE_EXCLUDE_FOLDERS=web/knta/rpt;web/knta/test
   ```
5. Save the `server.conf` file.
6. To regenerate the `server.conf` file and propagate the changes, run the `kUpdateHtml.sh` script from the <*PPM_Home*>\bin directory.
7. Run the `kStart.sh` script to restart the PPM Server.

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

Using PPM AntiSamy

This section describes the AntiSamy feature in PPM Center. This feature gains wisdom from the OWASP AntiSamy project. Generally speaking, AntiSamy is an HTML, CSS, and JavaScript filter that sanitizes user input based on a policy file. For more information about OWASP AntiSamy project, see https://www.owasp.org/index.php/Category:OWASP_AntiSamy_Project.

PPM AntiSamy makes sure user's HTML, CSS and JavaScript input strictly follows rules defined by the policy file `antisamy-ppm.xml`. For example, if you enable the AntiSamy feature, you cannot open hyperlinks on request details page or project details page. This is because the hyperlink-kind input by default does not meet the rules defined by `antisamy-ppm.xml`. To make hyperlinks accessible in PPM, you can configure the policy file as you demand.

Enabling/Disabling the AntiSamy Feature

You can enable or disable the AntiSamy feature by setting the server configuration parameter `ENABLE_ANTISAMY` in the `server.conf` file.

If you set the parameter to `true`, your enable the AntiSamy feature. User's HTML, CSS, and JavaScript input will be monitored by the policy file `antisamy-ppm.xml`.

If you set the parameter to `false`, you disable the AntiSamy feature. User's HTML, CSS, and JavaScript input will not be monitored.

By default, the AntiSamy feature is enabled. And HPE recommends that you keep the AntiSamy feature enabled.

Configuring AntiSamy Policy File

1. Open the policy file `antisamy-ppm.xml` located in the `<PPM_HOME>\conf` directory.
2. Configure the sections of the policy file as you want.
   - **Directives**

The following table shows the directives, their default values, and their impact on the AntiSamy filtering process.

Directive	Type	Default Value	Description
omitXmlDeclaration	boolean	true	When "useXHTML" is turned on, AntiSamy will automatically prepend the XML header. Enabling this feature will tell AntiSamy not to do that.
omitDoctypeDeclaration	boolean	true	When this feature is enabled, AntiSamy will automatically prepend the HTML doctype declaration.
maxInputSize	int	600000000	This directive specifies the maximum size (in bytes) of user input before it is validated.
useXHTML	boolean	true	When this feature is enabled, AntiSamy will output the sanitized data in XHTML format as opposed to just regular HTML.
formatOutput	boolean	true	When this feature is enabled, AntiSamy will automatically format the output according to some basic rules and indentation. Kind of like "pretty print."
embedStyleSheets	boolean	false	When the developer chooses to allow CSS, this directive will specify whether or not remote stylesheets found referenced in the user's input will be pulled down and embedded into the current user input.
connectionTimeout	int	5000	When "embedStyleSheets" is enabled, this timeout value (in milliseconds) will be used when fetching the offsite resource in question. This should be used to prevent validation threads from blocking when connecting

			to 3rd party systems that may purposefully act really slowly.
maxStyleSheetImports	int	3	This feature allows developers to specify how many remote stylesheets can be downloaded from any one input.

> **Note:** The `antisamy-ppm.xml` file only deploys some of the directives provided by the OWASP AntiSamy project. You can include more directives when configuring the policy file. For more information about other directives, see *AntiSamy User Guide*.

- **Common Regular Expressions**

  You can declare regular expressions here that can be used in the rest of the policy file.

  Example:

  ```
  <regexp value="[a-zA-Z0-9\:\-_\.]+" name="htmlId"/>
  ```

  This regular expression is used to determine whether text in an `id` attribute is valid or not.

- **Common Attributes**

  You can declare attributes here that are common to many different tags.

  Example:

  ```
  <attribute name="id" description="The 'id' of any HTML attribute should not
  contain anything besides letters and numbers">
     <regexp-list>
        <regexp name="htmlId"/>
     </regexp-list>
  </attribute>
  ```

  This is where the `id` attribute is mapped to the `htmlId` regular expression.

- **Global Tag Attributes**

  You can declare attributes here that are global to all different tags.

  Example:

  ```
  <attribute name="id"/>
  ```

  The `id` attribute is global to all different tags.

- **Tags to Encode**

You can declare tags that will not be removed, filtered, validated, or truncated, but encoded using HTML entities.

Example:

```
<tag>g</tag>
```

The g tag does not actually do anything, but it is not malicious either, so you can encode it, rather than remove it.

- **Tag Rules**

  You can define parsing rules here that will be used for each tag individually. What happens to tags depends on what actions AntiSamy has decided to perform on it. PPM's AntiSamy policy file by default includes the following actions for tags.

  - **Remove**: When the tag rule action is set to "remove" for a given tag, the tag is deleted with all of its child text.

    Example:

    ```
    <tag name="script" action="remove"/>
    ```

  - **Validate**: When the tag rule action is set to "validate" for a given tag, PPM verifies if its attributes and children elements follow rules defined in the policy file.

    Example:

    ```
    <tag name="a" action="validate">
      <attribute name="href">
        <regexp-list>
          <regexp name="ppm-report-token"/>
        </regexp-list>
      </attribute>
    </tag>
    ```

  - **Truncate**: When the tag rule action is set to "truncate" for a given tag, the element of the tag is kept, but all its attributes are removed.

    Example:

    ```
    <tag name="title" action="truncate"/>
    ```

  > **Note:** Apart from the above tag rules, you can also use "default" and "filter" to build you own tag rules. For information about more tag rules, see *AntiSamy User Guide*.

- **CSS Rules**

You can define parsing rules here that will be used for each CSS property individually. Only CSS defined in this section is allowed.

Example:

```
<property name="background-position" description="If a background image has
been specified, this property specifies its initial position.">
   <literal-list>
      <literal value="top"/>
      <literal value="center"/>
      <literal value="bottom"/>
      <literal value="left"/>
      <literal value="center"/>
      <literal value="right"/>
      <literal value="inherit"/>
   </literal-list>
   <regexp-list>
      <regexp name="percentage"/>
      <regexp name="length"/>
   </regexp-list>
</property>
```

The CSS background position property is allowed only when it matches these rules. Its value must be a percentage, length, or one of the literal values such as "top" and "center".

3. Save the changes.
4. Restart PPM Server.

Usage Example

The `antisamy-ppm.xml` file by default has the following tag rule:

```
<tag name="a" action="validate">
  <attribute name="href">
    <regexp-list>
      <regexp name="ppm-report-token"/>
    </regexp-list>
  </attribute>
</tag>
```

This means if an end user inputs a hyperlink in a field, the hyperlink cannot be opened from the PPM pages, unless the hyperlink is in conformity with the regular expression "ppm-report-token", which is defined as follows in the policy file.

```
<regexp value="\[\w+\.\S+\]" name="ppm-report-token"/>
```

If you want to open hyperlinks from PPM pages, you should delete or edit the regulation expression in the above tag rule. For example, you can change the tag rule into the followings:

> **Caution:** The regular expression "ppm-report-token" mitigates most attack vectors such as XSS. If you delete this regular expression, some PPM pages will not be protected from XSS. HPE highly recommends that you exercise caution when deleting or editing the regular expression.

```
<tag name="a" action="validate">
  <attribute name="href">
  </attribute>
</tag>
```

Or

```
<tag name="a" action="validate">
  <attribute name="href">
    <regexp-list>
      <regexp name="anything"/>
    </regexp-list>
  </attribute>
</tag>
```

where `<regexp value=".*" name="anything"/>`

Chapter 9: Migrating Instances

This chapter covers the following topics:

Overview of Instance Migration

Each PPM instance consists of a file system and an Oracle database, which can exist on Windows or UNIX machines. You can migrate PPM using one of the following methods:

- Copy an entire PPM 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 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 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 HPE 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. 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.

(Optional) Migrating Document Management

If your source machine has document management installed and integrated with PPM, 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 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 Autopass 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 version number
   - Machine IP address
   - Operating system (Windows or UNIX)
   - Server purpose (development, test, or production)
2. Go to the HPE Licensing for Software portal: http://enterpriselicense.hpe.com/redirector/home.
3. Click **Sign In**.
4. Provide your HPE Passport credentials and click **Sign in**.

   > **Note:** If you do not have an HPE Passport, click **Create an account**.

5. On the Enter Entitlement Order Number page, enter the Order number found on the Entitlement Certificate and click **Go**.
6. Complete the activation process to generate an Autopass license key file.

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 instance. For information about how to stop the server, see "Starting and Stopping the PPM Server on a Single-Server System" on page 77.

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" below
- "Migrating to a UNIX Machine" on page 357

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 353.
2. Stop the PPM Server.

   For information on how to stop the server, see "Starting and Stopping the PPM Server on a Single-Server System" on page 77.
3. Migrate the PPM 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 database schema.

   For information about how to migrate the database schema, see "Migrating the Database Schemas" on page 358.
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 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. Create a Windows service to start the new PPM instance.
    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>
    ```

    > **Note:** 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 a Single-Server System" on page 77.

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 353.
2. Stop the PPM Server.

   For information about how to stop the PPM Server, see "Starting and Stopping the PPM Server on a Single-Server System" on page 77.
3. Migrate the PPM file system.
   a. On the PPM Server host machine, navigate to the parent of the <*PPM_Home*> directory.
   b. Use an archiving utility (such as Tar or Zip) to create an archive file of the entire <*PPM_Home*> directory.

      Example:

      If the <*PPM_Home*> directory is named "PPM", run the command:

      ```
      $ tar cf ppm930.tar PPM
      ```
   c. Use FTP in binary mode to 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 ppm930.tar
      ```

      This creates the new PPM Server directory structure. A directory named PPM is created automatically.
4. Migrate the PPM database schema.

   For information about how to migrate the database schema, see "Migrating the Database Schemas" on the next page.
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. Install the new Autopass license key file using the `kLicenseInstall.sh` tool.

   The license file is installed and becomes effective right away, with a message popping up showing how many licenses are installed.

7. Start the new PPM Server.

For information about how to start the server, see "Starting and Stopping the PPM Server on a Single-Server System" on page 77.

Migrating the Database Schemas

This section provides the procedures used to migrate the PPM database schemas from one database to another.

What to consider before migrating the database schemas

- Export and import tools

  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. Using incompatible versions of export and import tools causes errors in instance migration.

- If you use the Extension for Oracle E-Business Suite

  If you have Deployment Management Extension for Oracle E-Business Suite, you must consider the location of your Primary Object Migrator Host when migrating the PPM database schema, because Object Migrator might reside in the same database, or even the same schema, as PPM.

  Migrating the schema does not require migrating the Object Migrator instance because the integration method in PPM can be refreshed to use the existing Object Migrator installation. If Object Migrator shares a database with PPM, and you intend to migrate it as well as PPM, the destination database must support Object Migrator. (For more information, see the *Object Migrator Guide*.)

  Unless PPM and Object Migrator share the same schema, the migration of Object Migrator is completely separate from the migration of PPM, and should be completed before you migrate the PPM database. Contact HPE Software Support Web site (https://softwaresupport.hpe.com) for instructions on how to perform this migration.

  If PPM and Object Migrator share the same schema and you want to migrate both, you must coordinate the migration activities. Contact HPE Software Support Web site (https://softwaresupport.hpe.com) for instructions.

  Regardless of the configuration, refresh the integration definition after you migrate the PPM schemas.

How 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 a Single-Server System" on page 77.

2. Export the PPM database schema to a file by running the `expdp` command as shown in the following example.

```
$ORACLE_HOME/bin/expdp USERID=system/<Password>@<DB> DUMPFILE=<Export_Filename>
DIRECTORY=<Dump_Dir> schema=<Source_SCHEMA> LOG=export_knta_920.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`).
<Dump_Dir>	represents database dump directory. To create the directory, run the following: **`create directory DUMP_DIR as `c:/dump_dir';`**
<Source_Schema>	represents the name of the PPM database schema to export.

3. Export the RML schema.
4. Create the new PPM database schema:
   a. Run the `CreateKintanaUser.sql` script (located in the *<PPM_Home>*`/install_930/ppm930/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 `ppm920/sys` directory) from SQL*PLUS as the SYS DBA user.
5. Create the new PPM RML database schema.

   To create a new, empty RML database schema in the target database, run the `CreateRMLUser.sql` script (located in the `ppm920/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 database schema, run the `impdp` command, as shown in the following example.

   ```
   $ ORACLE_HOME/bin/impdp USERID=system/<Password>@<DB> DIRECTORY=<Dump_Dir>
   REMAP_SCHEMA=<Source_Schema>:<Target_Schema> DUMPFILE=<Export_Filename>
   LOG=import_knta_920.log
   ```

where

<*Password*>	represents the password for the system user on the Oracle database
<*DB*>	represents the database connect string
<*Dump_Dir*>	represents database dump directory.
<*Source_Schema*>	represents the name of the PPM database schema previously exported.
<*Target_Schema*>	represents the name of the new PPM database schema
<*Export_Filename*>	represents the name of the file that contains the export. The filename must have the `dmp` extension (for example, `kntaExport.dmp`).

7. Import the RML export file.
8. Create the RML-related packages in the RML schema:
   a. `cd<PPM_Home>/install_920/rml`
   b. `sqlplus <RML_Username>/<RML_Password>@<SID>@rml_driver.sql`
9. Grant privileges to the PPM RML database schema:

   > **Note:** You can find the following scripts in the <*PPM_Home*>/install_920/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 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.

    > **Note:** If PPM and Object Migrator share the same database schema, the PPM database schema is referred to as the PPM account, and the Object Migrator schema is referred to as the Object Migrator account.

    To provide the necessary grants and permissions to the PPM 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 Object Migrator resides in the same schema as PPM, complete the Object Migrator migration.

For assistance, contact HPE 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.

> **Note:** To validate any invalid PPM database objects generated during link regeneration, run the `RecompileInvalid.sql` script, which is located in the `<PPM_Home>/install_920` directory. Run this script from SQL*Plus connected as the new PPM 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 "PPM Configuration Parameters" on page 401.

> **Note:** 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 a Single-Server System" on page 77).

Troubleshooting Instance Migrations

This section describes common problems that you might encounter as you migrate PPM 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>/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.

   > **Note:** 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 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 URL (for example, existing requests) are updated to point to the new URL.

Chapter 10: Migrating Entities

This chapter contains the following topics:

About Entity Migration

Entity migrators are Deployment Management object types. Each migrator is designed to migrate a specific kind of PPM entity and all of its dependent objects from one PPM instance to another.

You can use Deployment Management to manage configuration changes to PPM. Deployment Management comes with an out-of-the-box set of object types, or *entity migrators,* that you can use to move PPM configuration entities (workflows, request types, and so on) between PPM instances. If you maintain scratch instances for developing and testing PPM 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 implementation. You can build a workflow for every migrator object type, or create a single generic workflow for all migrator object types.

> **Note:** You can only migrate entities between PPM instances of the same version.

You can migrate the following PPM 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. 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 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 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 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 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.

   > **Note:** If you want to perform version control on changes to PPM configuration entities as they are migrated, you can version the compressed file that is extracted from the source instance.
   >
   > HPE 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 to a production instance.

> **Note:** 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. Create the `KINTANA_SERVER` environment If the environment definition for the PPM Server is not configured.

   > **Note:** 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.

      > **Note:** 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 SSH session.

   d. Define and enable the source environment.

      > **Note:** 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 *Deployment Management User's 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 *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 account on the source instance.
PPM dest password	In this field, type the password for your PPM account on the destination instance.
Request type	In this field, type the name of the request type that you want to migrate.

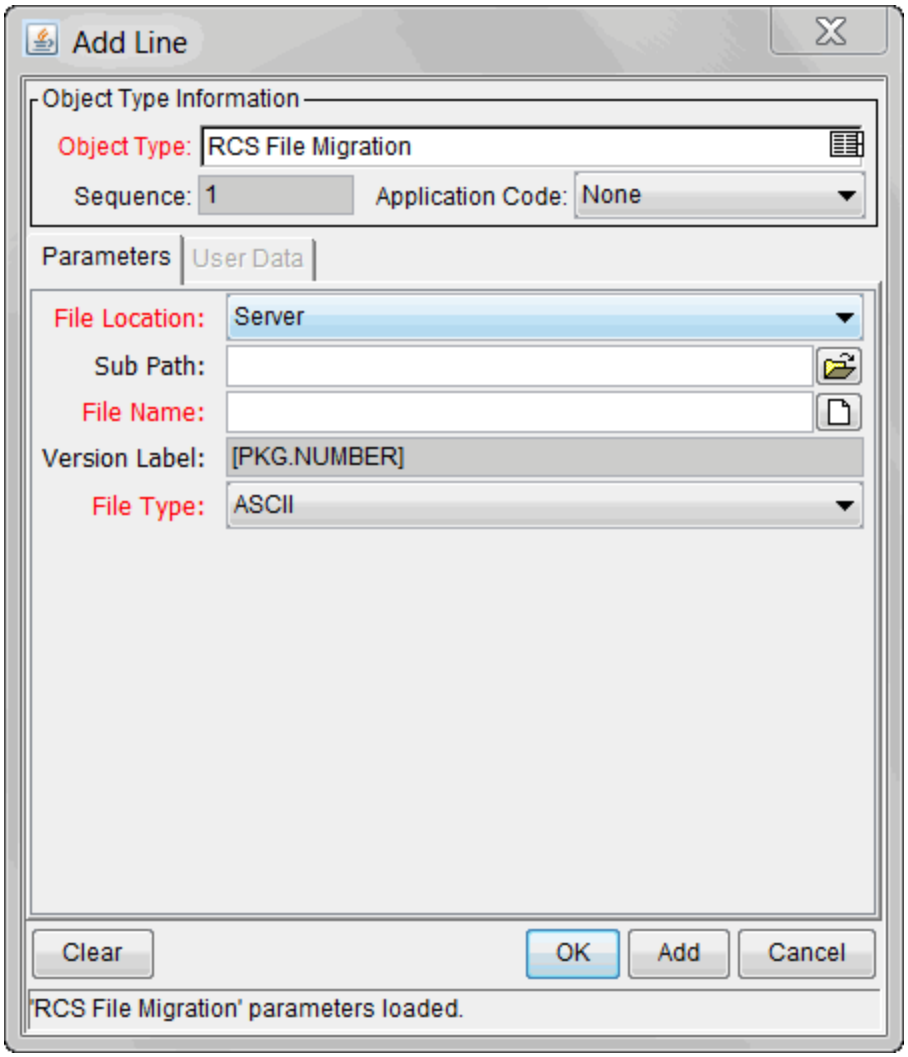
> **Note:** For information on **Migrator action** list dependencies, see "Migrator Action List" on the next page.

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 entity migrators has a set of parameters similar to those described in this section (and as shown in the previous example). The RCS File Migration shown in the following figure is an example.

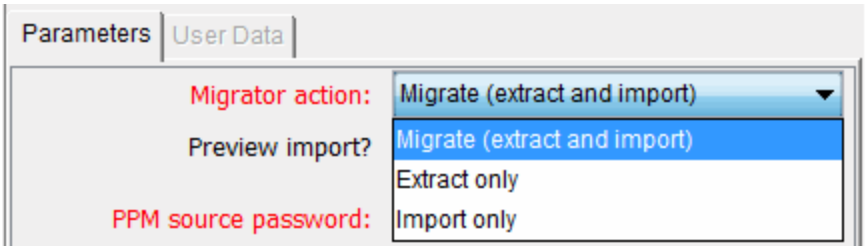
Figure 10-1. Add Line dialog box for the RCS File Migration



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. "Migrator Action List" above shows the **Migrator action** list.

Figure 10-2. Migrator action list



In the **Migrator action** list, you can select one of the following actions:

- **Migrate (extract and import)**
- **Extract only**
- **Import only**

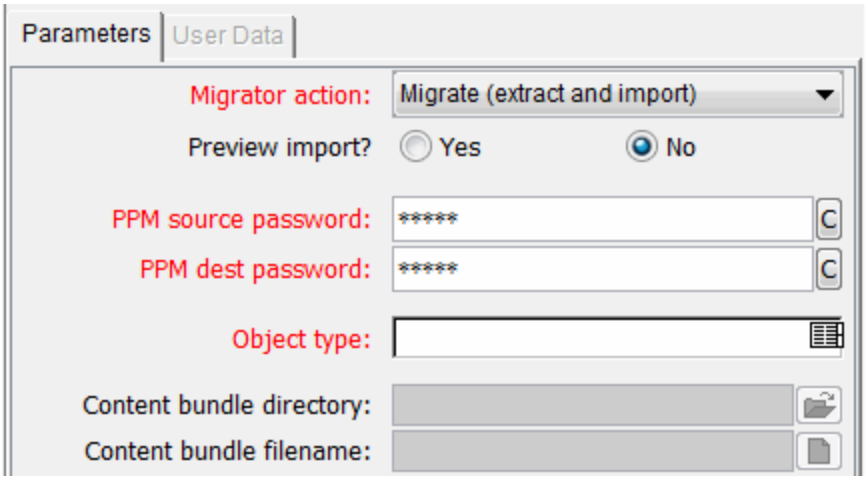
The following table lists the controls in the Add Line dialog box that are affected by the migrator action you select, and provides information about how each control is affected.

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 the following figure, the parameters are the entity name (in this case, the request type), content bundle directory, and content bundle filename.

Figure 10-3. Basic parameters



Content Bundle Controls

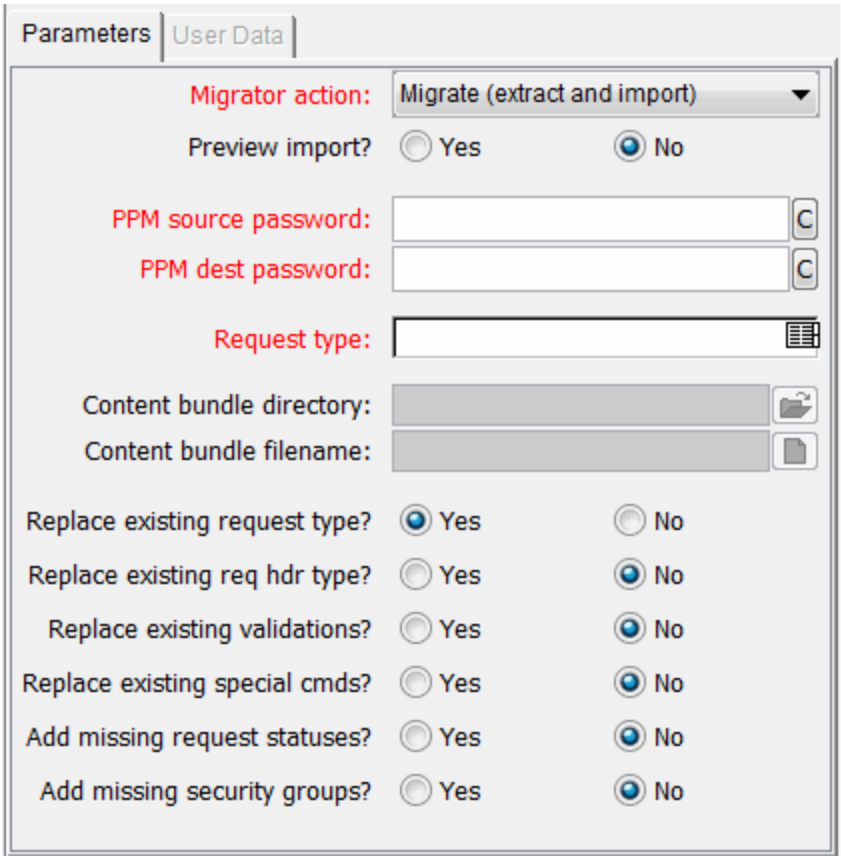
The behavior of controls related to the content bundle depends on the migrator action you select, as follows:

- 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 the following figure) to control migrator behavior.

Figure 10-4. 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 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 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 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 special commands (including parent and child special commands) that exist in the target PPM 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 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 ("Password Fields" above) are enabled.

Figure 10-5. Password fields

Source Password Field

When the migrator contacts the source server, it uses the credentials of the current PPM user to authorize the entity extraction. This user must be part of a security group that contains the access grant

"System Admin: Migrate HPE 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 user to authorize the entity import. This user must be part of a security group that has the "Sys Admin: Migrate HPE 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 databases involved have the same localization settings. However, if you must migrate configuration entities between PPM 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.
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 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, Deployment Management logs on to remote machines in the same way another user would (that is, using FTP, SCP, SSH). Deployment Management can log on to a remote server using any existing operating system user name and password.

HPE recommends that you generate a new user (for example, PPM) on every machine to which 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's command execution engine is to use to send and receive commands to a remote computer (via SSH or FTP/SCP). This setting is important if your PPM 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 uses to communicate with that server machine.

This is important if PPM 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 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 to be used to connect to the server or client. For information about connection protocols that PPM supports, see the *System Requirements and Compatibility Matrix* and the *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 *Deployment Management Configuration Guide.*

Setting the SERVER_ENV_NAME Parameter

The PPM migrators depend on the `SERVER_ENV_NAME` server configuration parameter. This parameter specifies the name of an environment definition in the PPM system that describes the host server running that PPM instance.

When you installed PPM, 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 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 users have ownership and control over different PPM 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 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.

> **Note:** 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. 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

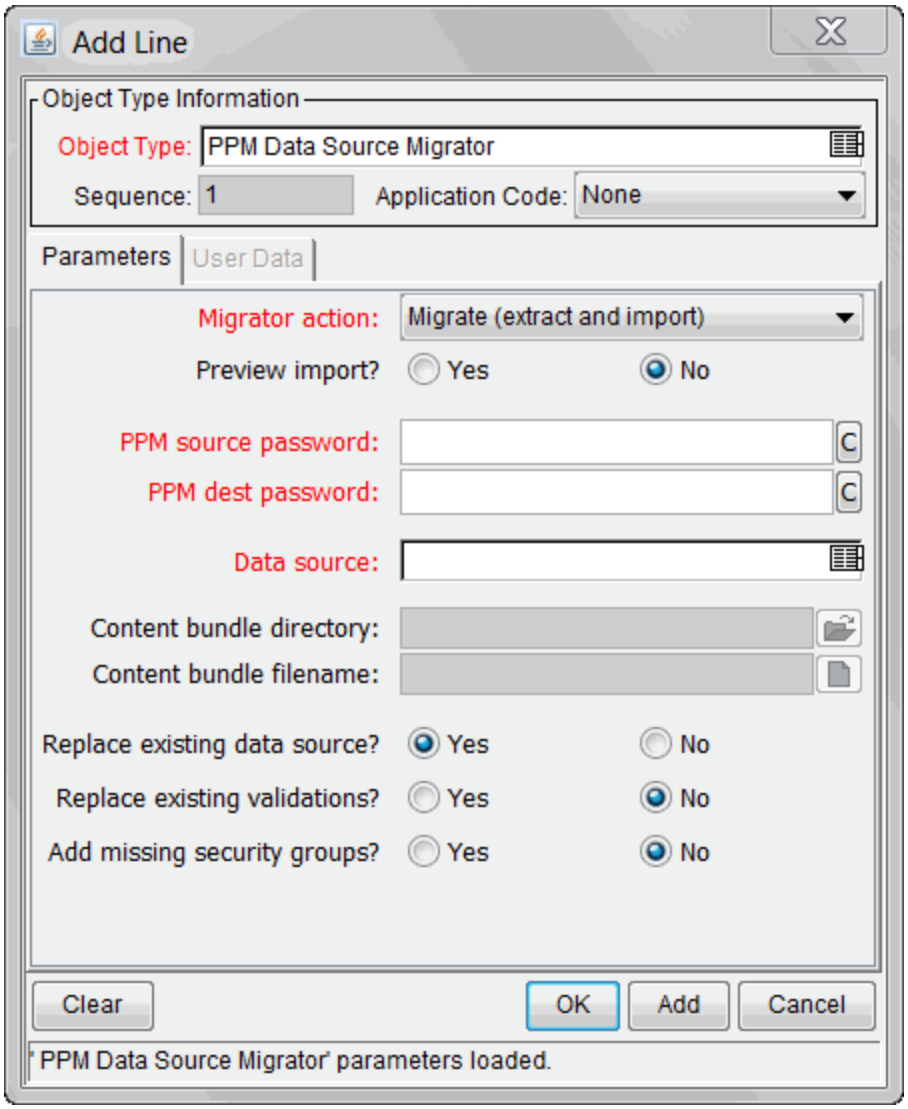
This section provides descriptions of PPM entity migrators.

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 instances. (Data sources provide data displayed in PPM Dashboard portlets.)

The following figure shows the parameters for the Data Source migrator as they are displayed during package line creation.

Figure 10-6. Data Source Migrator

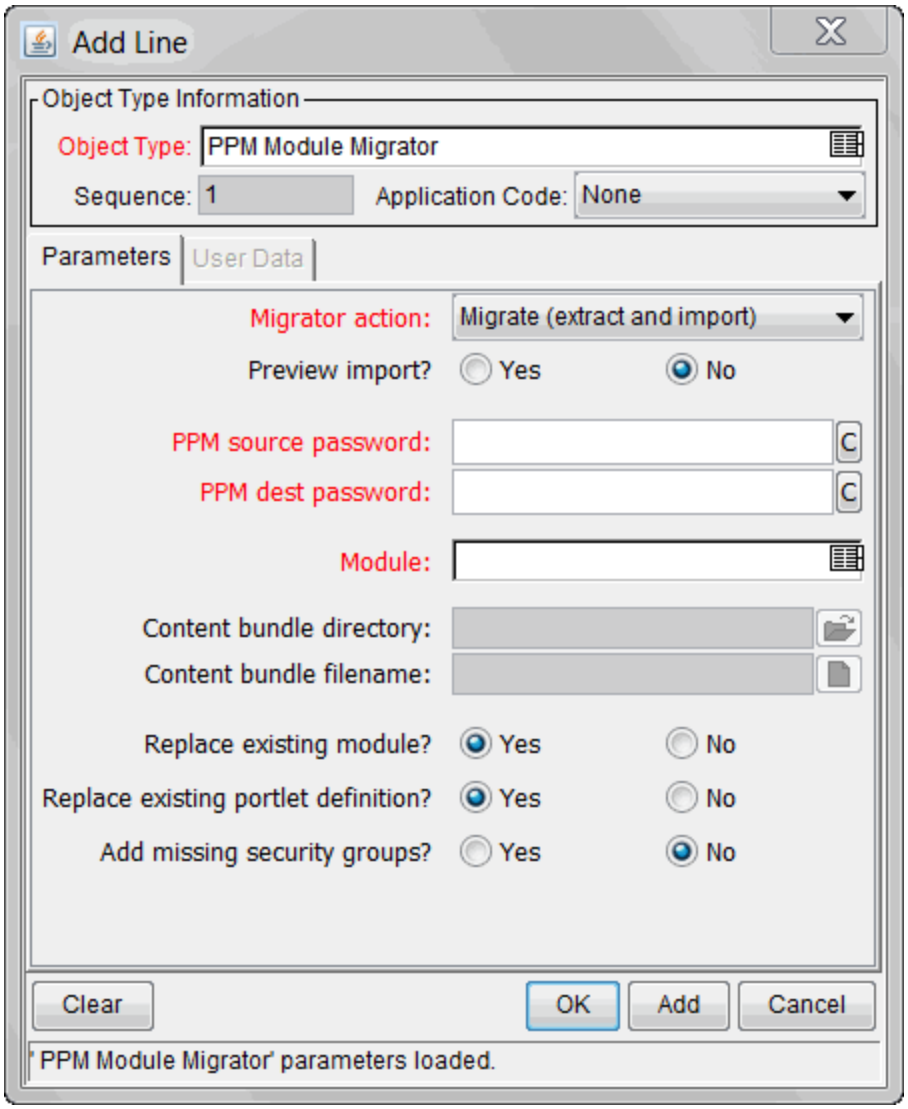


For information about the fields in this migrator, see "Defining Entity Migrators" on page 368. For information about how to create a portlet data source, see the *Creating Portlets and Modules*.

Module Migrator

In the PPM 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 modules from one PPM environment to another.

Figure 10-7. Module Migrator



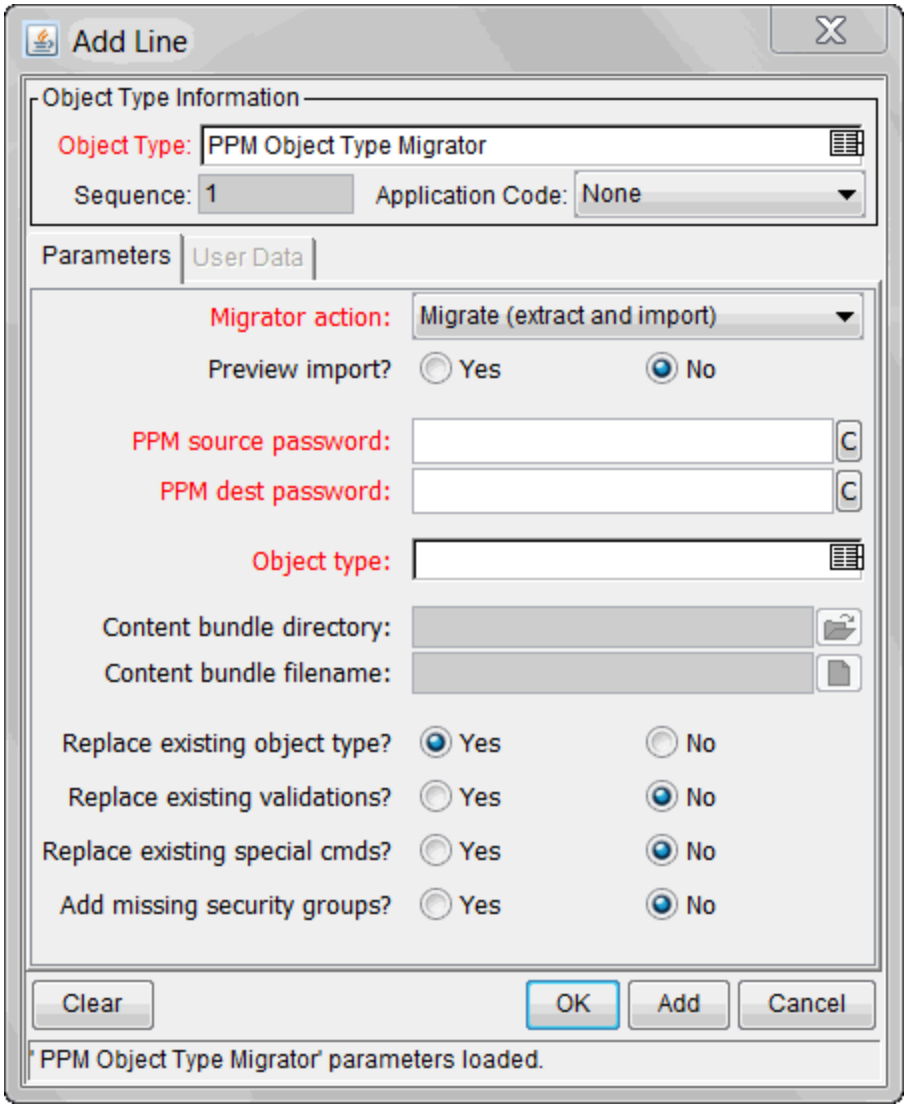
For information about the fields in this migrator, see "Defining Entity Migrators" on page 368. For information about how to create modules, see the *Creating Portlets and Modules* guide.

Object Type Migrator

The Object Type Migrator contains the additional option **Replace existing special cmds?** If the validation to be migrated references PPM special commands (including parent and child special commands) that exist in the target PPM 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 10-8. Object Type Migrator



For information about most of the controls in this migrator window, see "Defining Entity Migrators" on page 368.

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

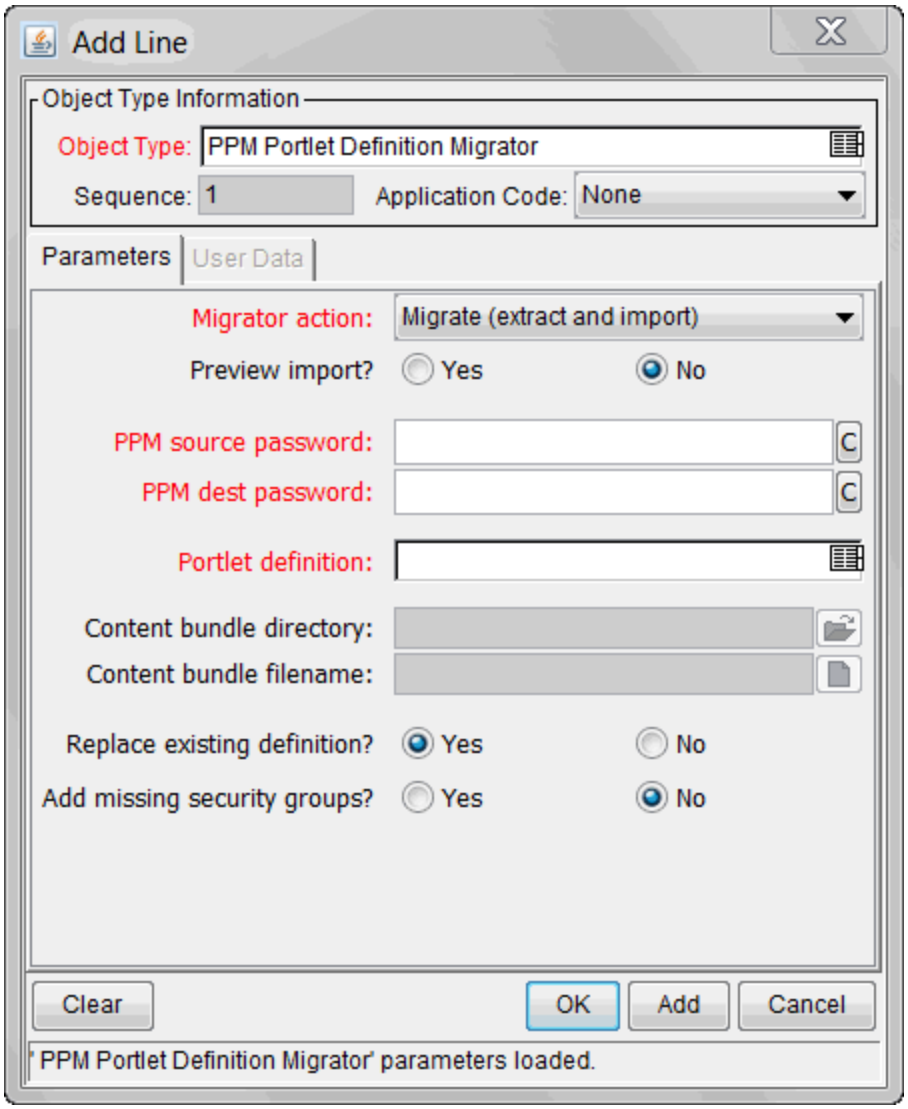
- Special commands referenced by other special commands
- Ownership group information for the entity

> **Note:** 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 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, the migrated changes are applied to all users who have added the same portlet to their PPM Dashboard pages.

Figure 10-9. Portlet Definition Migrator

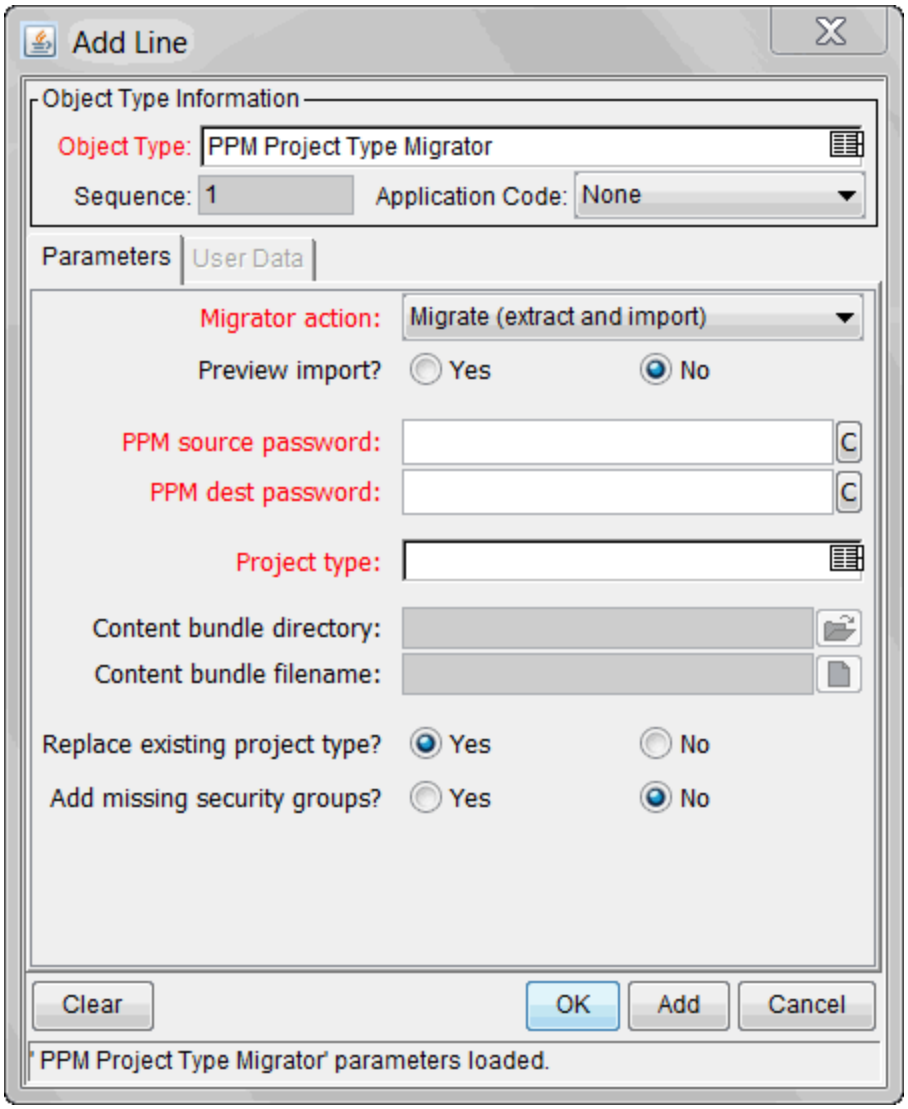


For information about the fields in this migrator, see "Defining Entity Migrators" on page 368.

Project Type Migrator

You can define project types in a development or testing instance of PPM, and then use the Project Type Migrator to migrate them to production after testing.

Figure 10-10. 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).

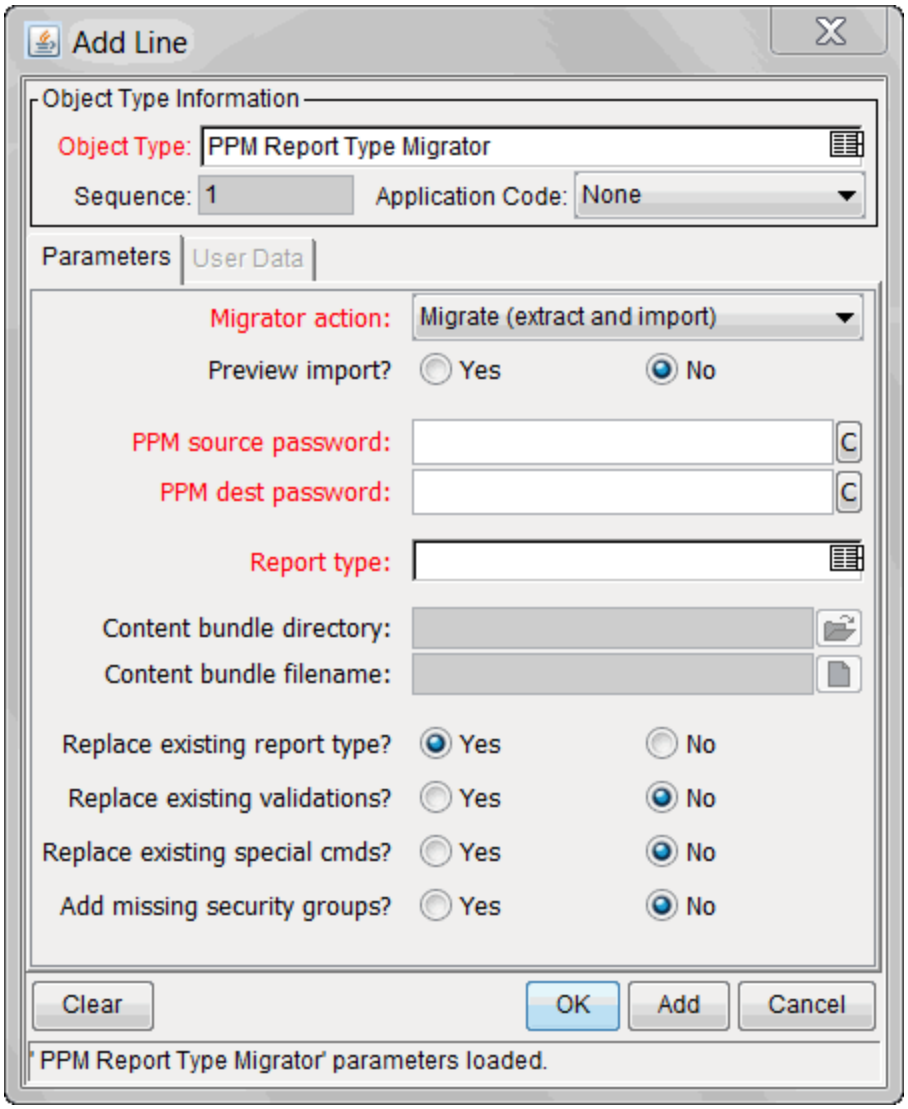
Because project types are useless without an associated project request, you must either migrate the associated request type first, so that the link to the project type is resolved when you migrate the project type is migrated, or edit the project type after you migrate it.

> **Note:** The Project Type Migrator does not transport secondary objects as dependencies.

Report Type Migrator

The Report Type Migrator contains the additional option **Replace Existing special cmds?** If the validation to be migrated references PPM special commands (including parent and child special commands) that already exist in the target PPM instance, you can choose to replace them (or not). (The default value is **No.**) Regardless of their values, PPM automatically re-creates special commands that are missing from the destination instance.

Figure 10-11. Report Type Migrator



For information about most of the fields in this migrator, see "Defining Entity Migrators" on page 368.

Configuration Considerations

The Report Type Migrator also transfers the following information:

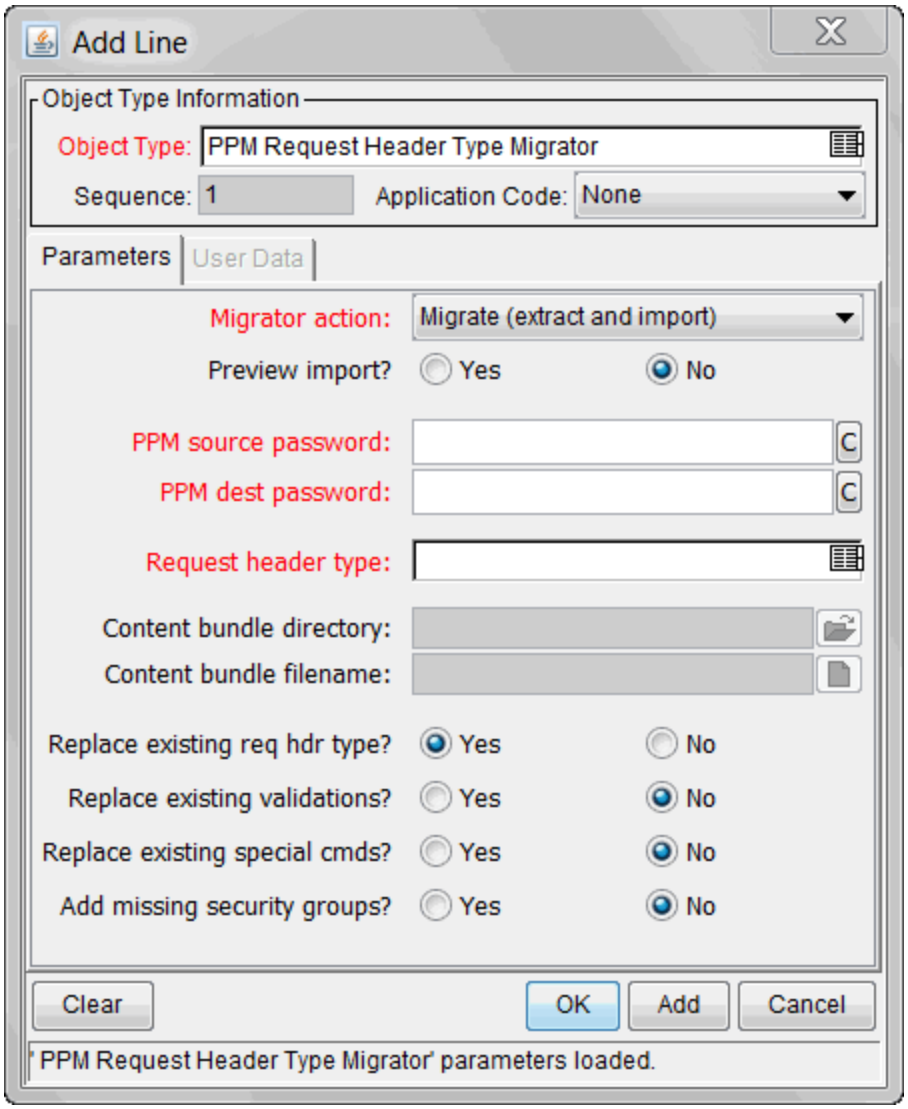
- Special commands referenced by command steps
- Validations referenced by fields
- Environments referenced by validations
- Special commands referenced by validations
- Special commands referenced by other special commands
- Ownership group information for the report type

> **Note:** 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 contains the additional option **Replace Existing special cmds?** If the validation to be migrated references PPM special commands that already exist in the target PPM 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 automatically re-creates special commands that are missing from the destination instance.

Figure 10-12. Request Header Type Migrator



For information about most of the fields in this migrator, see "Defining Entity Migrators" on page 368.

Configuration Considerations

The Request Header Type Migrator also transfers the following information:

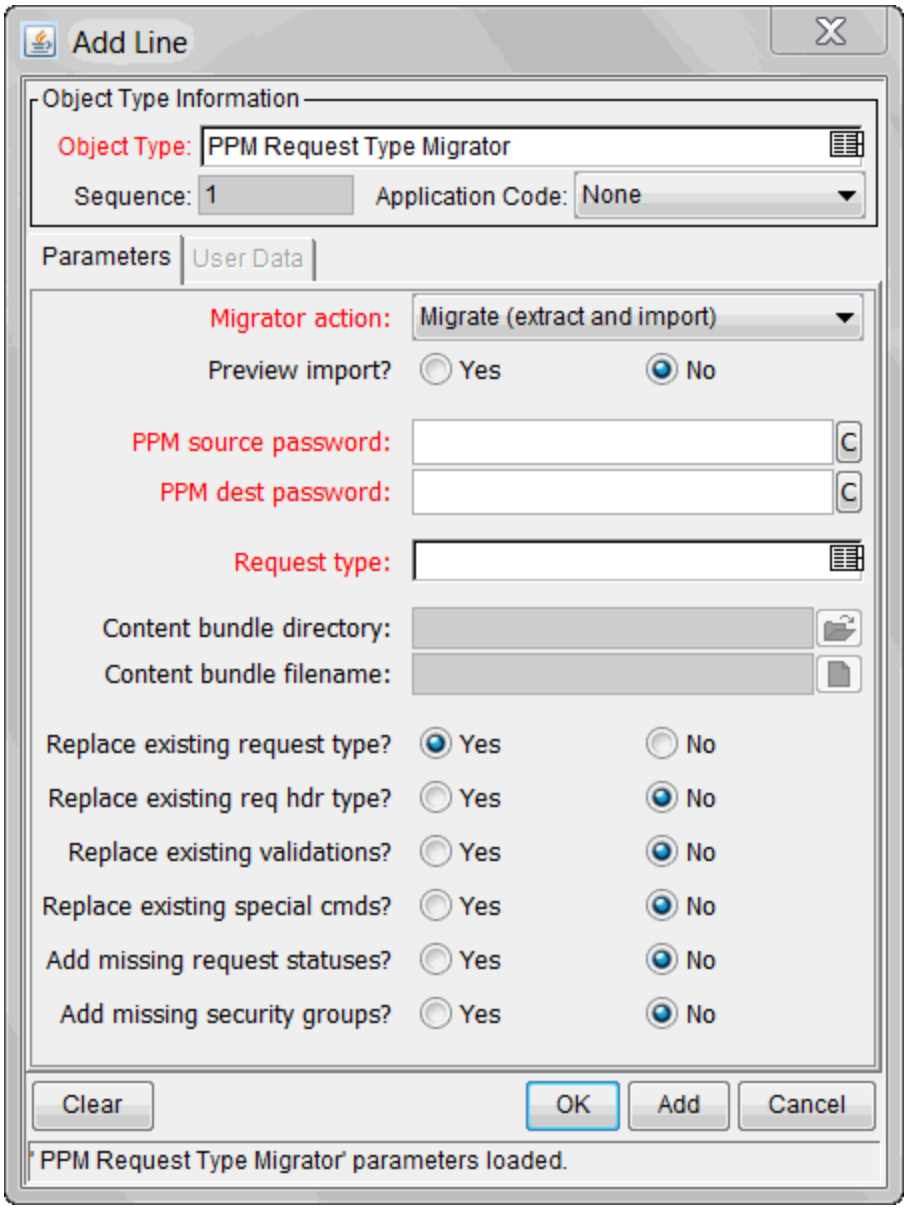
- Validations referenced by fields
- Environments referenced by validations
- Special commands referenced by validations
- Special commands referenced by other special commands
- Ownership group information for the request header type

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 has additional import behavior options from which to choose.

Figure 10-13. 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 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 special commands that already exist in the target PPM 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 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 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.

> **Note:** 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.

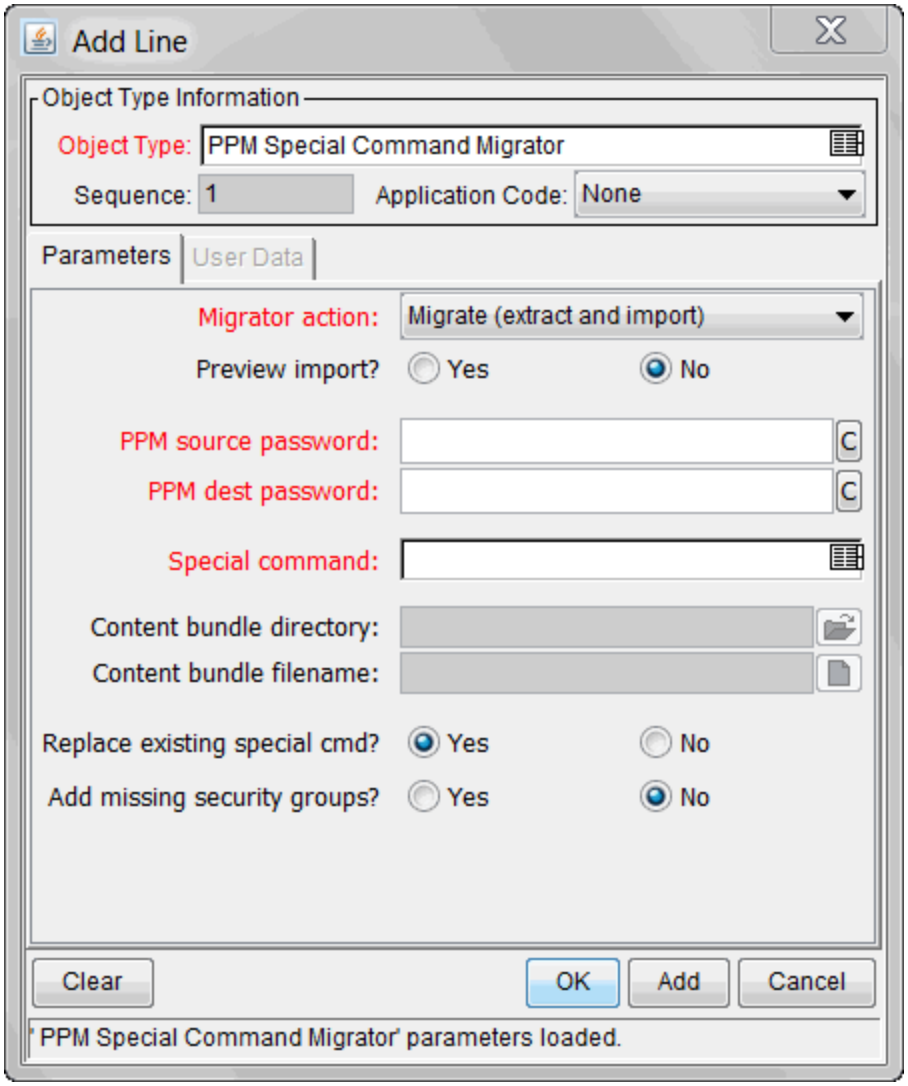
Circular references between request types and workflows could make it necessary to migrate either a request type or workflow twice:

- 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 to move the special commands between instances of PPM separately.

Figure 10-14. Special Command Migrator

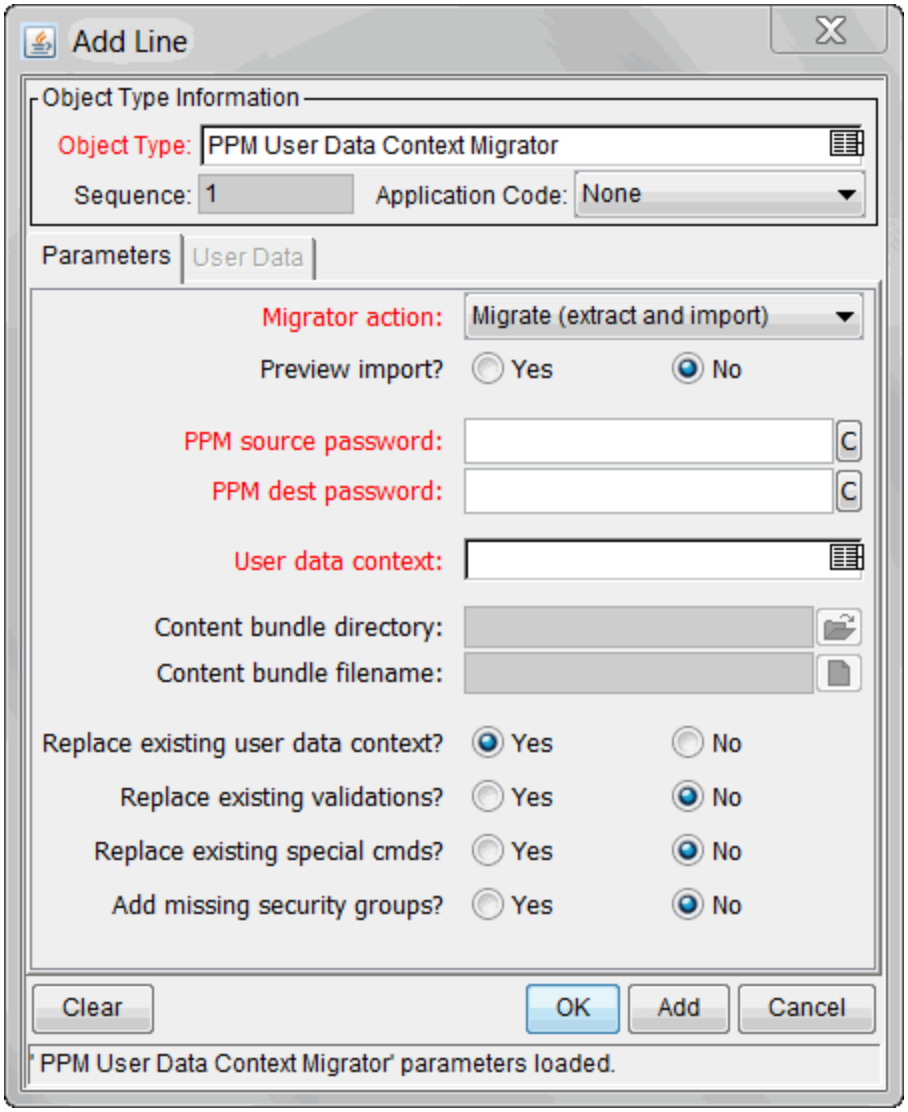


For information about the fields in this migrator, see "Defining Entity Migrators" on page 368.

User Data Context Migrator

The User Data Context Migrator contains the additional option **Replace Existing special cmds?** If the validation to be migrated references PPM special commands that already exist in the target PPM 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 automatically re-creates special commands that are missing from the destination instance.

Figure 10-15. User Data Context Migrator

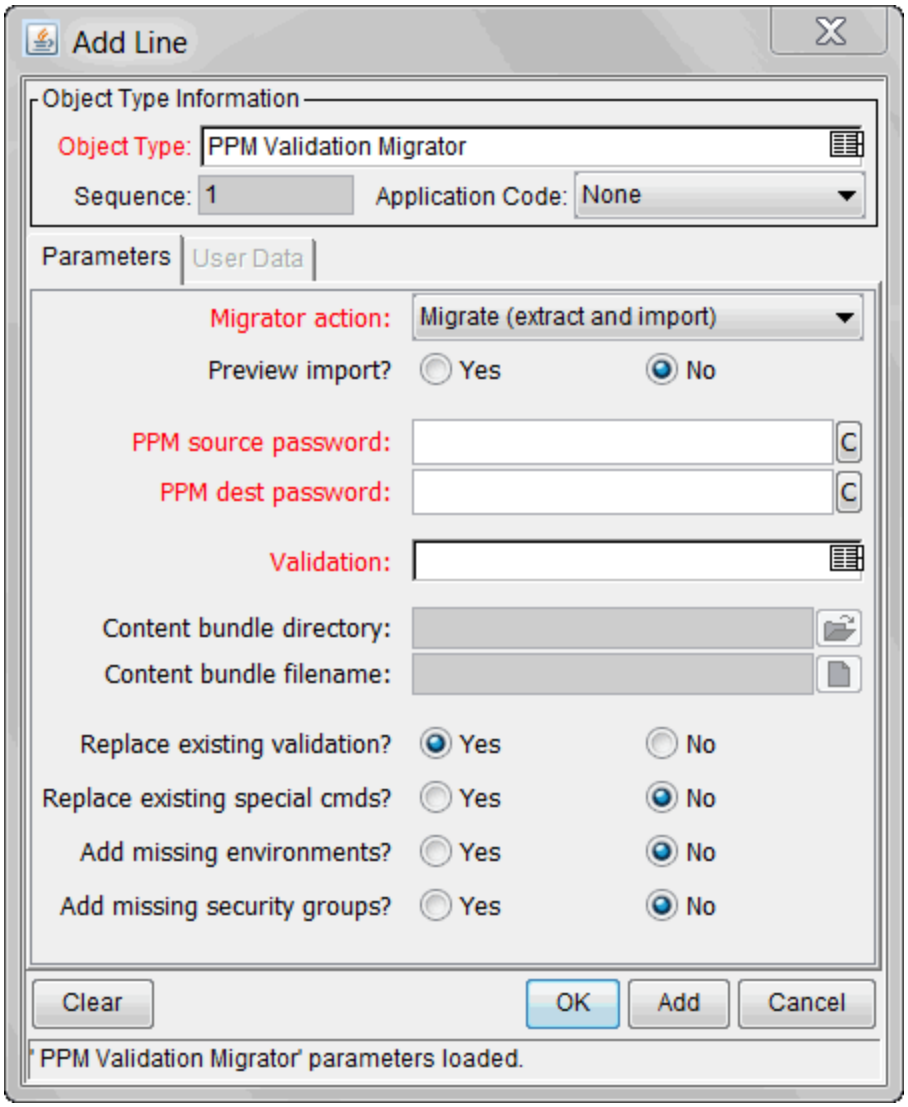


For information about most of the fields in the User Data Context Migrator, see "Defining Entity Migrators" on page 368.

Validation Migrator

The Validation Migrator is shown in the following figure.

Figure 10-16. Validation Migrator



This migrator contains the following two additional import behavior options:

- **Replace existing special cmds?**

  If the validation to be migrated references PPM special commands that already exist in the target PPM 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 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 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 368.

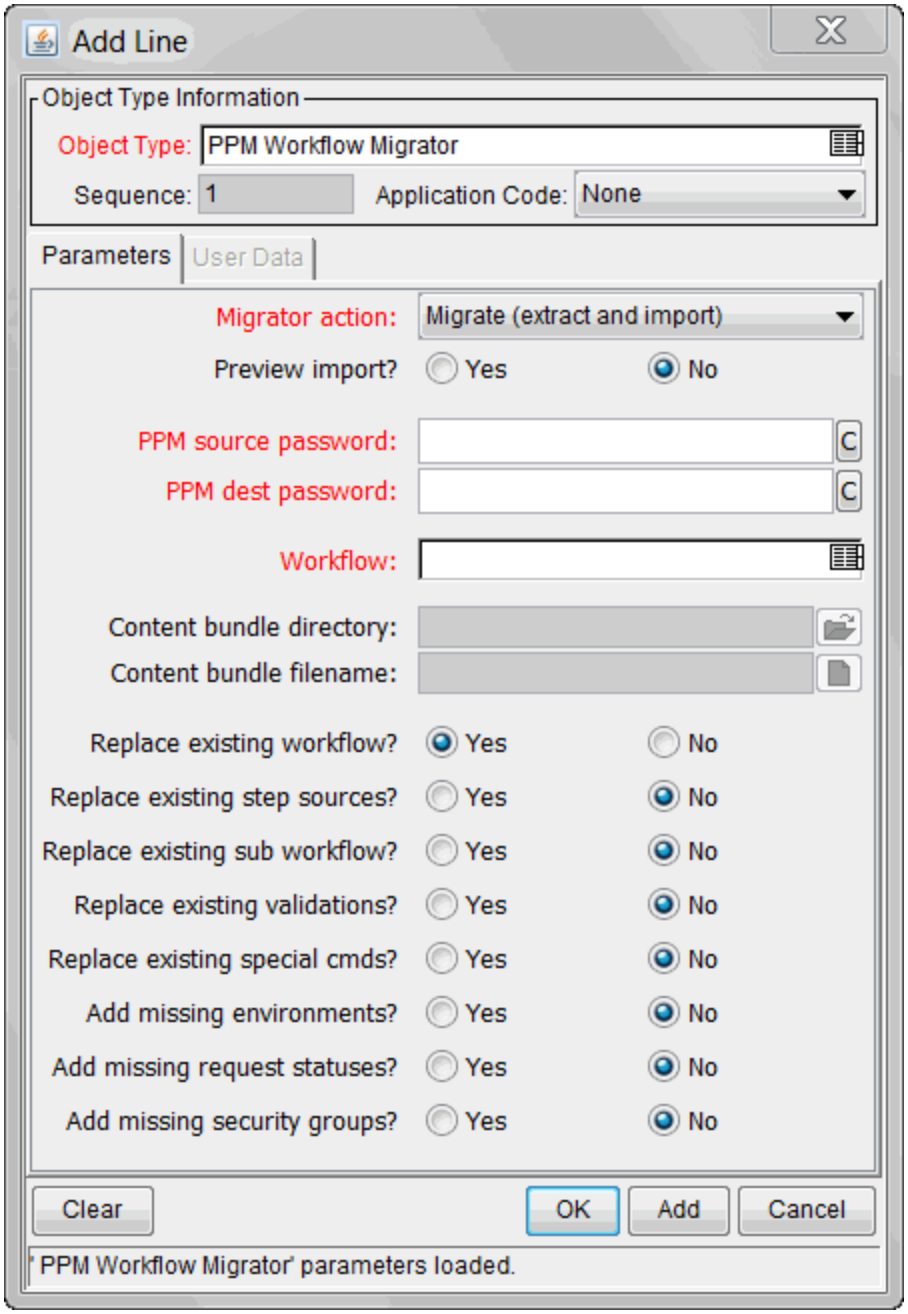
Configuration Considerations

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 the following figure.

Figure 10-17. Workflow Migrator



This migrator provides the following additional import behavior options:

- **Replace existing special cmds?**

  If the workflow to be migrated references PPM special commands that already exist in the target PPM 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 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 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 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 368.

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.

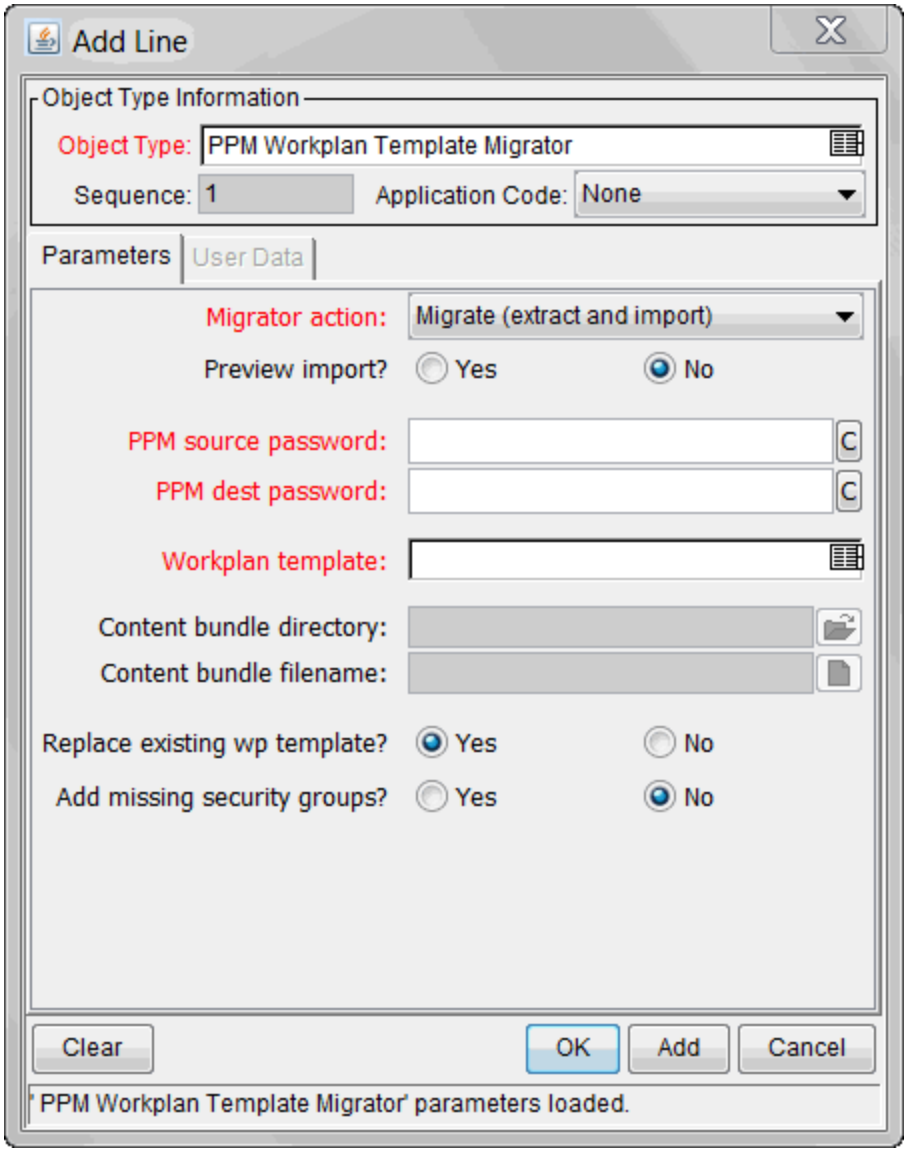
If there are circular references between workflows and request types, you may have to migrate either a workflow or request type twice:

- 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.

Work Plan Template Migrator

You can define work plan templates in a development or testing instance of Project Management, and then use the Work plan Template Migrator to migrate them to production after testing is completed.

Figure 10-18. 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.

Appendix A: PPM Configuration Parameters

This appendix contains the following topics:

Overview of Configuration Parameters

This appendix lists and describes the PPM 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 "Server Directory Structure and Server Tools" on page 490.

Determining the Correct Parameter Settings

For most PPM 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 165.

For information about how to specify your own parameters, see "Defining Custom and Special Parameters" on page 90.

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 automatically uses the appropriate path separators to communicate with Microsoft Windows. HPE 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 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.

> **Note:** HPE 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 284.

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 514.

   > **Caution:** 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.

   > **Note:** 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 307 and "Running server reports from the command line " on page 310.)

The following table 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

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 Resource Management. This acts as a safety valve to prevent PPM 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 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 the **Save Draft** button on the Create New Request page, which allows Demand Management users to save requests without automatically submitting them in the standard interface.	Default: `false` Valid values: `true, false`
APP_SERVER_ ALERT_TEXT	Alert text that displays on the application server logon page and headers.	
**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.
ARP_MAX_	Maximum number of resource pools that can be	Default: `30`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
RESOURCE_ POOLS	represented in the Analyze Resource Pool portlet in 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.	Valid values: Positive integer
ARP_MAX_ RESOURCES	Maximum number of resources that can be represented in the Analyze Resource Pool portlet in 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
ARP_PORTLET_ MAX_ RESOURCE_ POOLS	Maximum number of resources to be represented in the Analyze Resource Pool portlet in 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
ASSET_ROLLUP_ TIMESHEETLINE_ BATCH_SIZE	Specifies the batch size of time sheet lines for asset cost calculation.	Default: `800` Valid: 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`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
AUTHENTICATE_ REPORTS	If set to `true`, access to all reports requires user authentication. (A user must provide a PPM user login ID).	Default: `true` Valid values: `true, false`
*AUTHENTICATIO N_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`
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`
AUTOCOMPLET E_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)
BASE_ CURRENCY_ID	Dre: BASE_CURRENCY_ID: This is odd, default is not found in usual location even though parameter is used. Should this be configurable?	Default: 97
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, continued

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. If you want to specify a literal IPv6 address, make sure you enclose the literal address with "[" and "]" characters. For example, `http://[::1]:8080` **Note:** HPE strongly recommends you specify DNS name instead of literal IPv6 address.	Example: `http://www.mydomain.com:8080`
BASE_URL_QC_INTEG	Used in the process of enabling the integration of PPM with Quality Center when PPM is set up with an external Web server with HTTPS enabled. For detailed information on how to use this parameter, see the *Solution Integrations Guide*.	Default:`N/A` Valid values: `http://<Instance_Host_Name>:<HTTP_Port>/itg/`
BLOCK_PENDING_PKGL_FOR_ERROR	Specifies whether or not the system continues executing the subsequent package lines when a package line fails. If you set this parameter to `false`, when a package line fails, the system continues executing the subsequent package lines. If you set this parameter to `true`, the execution is blocked when a package line fails. When a package line fails and its subsequent package lines pass, the status of this execution is still successful in PPM Workbench. However, you should note that even when the status is successful, it does not mean all the package lines	Default: `true` Valid: `true`, `false`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	are executed successfully in logical relationship to meet your business needs. HPE suggests that you exercise caution when setting this parameter to false, unless you are absolutely sure about the consequences.	
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`
BYPASS_STARTUP_CHECKS	If set to true, prevents server checks at PPM startup. **Caution:**HPE strongly recommends that you leave this parameter set to `false`, unless HPE Software Support has advised that you set it to `true`.	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.	Valid value format: `http://<Host>:<Port>/ccm/`
CHANGE_MANAGEMENT_LICENSE_KEY	License key for Demand Management.	Default: N/A

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
CHECK_ COMMIT_ RESOURCES_ IN_RESOURCE_ FINDER	Flags whether or not the option **Commit selected resources** is checked in resource finder. The setting of this parameter is only meaningful when the parameter "ENABLE_COMMIT_ RESOURCES_IN_RESOURCE_FINDER" on page 416 is set to `true`.	Default: `false` Valid values: `true, false`
CHECK_ DISTRIBUTE_ RESOURCES_ IN_RESOURCE_ FINDER	Flags whether or not the option **Distribute demand evenly to selected resources** is checked in resource finder. The setting of this parameter is only meaningful when the parameter "ENABLE_DISTRIBUTE_ RESOURCES_IN_RESOURCE_FINDER" on page 416 is set to `true`.	Default: `false` Valid values: `true, false`
CLIENT_RMI_URL	Port on which the PPM Server listens to initiate RMI client/server communication via port forwarding. Must be a unique port, distinct from the Web server, SQL*Net, and the HTTP or HTTPS ports. Format: `rmi://<`*Public_Server_IP_Address*`>:<`*Public_ Server_Port*`>/KintanaServer`	Valid values: Port numbers higher than 1024 Example `rmi://gold.ppm .com:8082 /PPMServer`
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`
CMQC_QC_ VERSION	Specifies ALM/Quality Center version for CMQC solution.	Valid values: `10.00, 11.00`
CMQC_INIT_ STRING	The value for `initString` parameter for CMQC solution, for example, `INI:s4OF+cwwevEkcnJ9zWHwpE8ktxfl1pb5y8QoEN FQLs8=` You can get the value from Quality Center/ALM	Valid value: string

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	server. The `initString` value is stored in a properties file (ALM\jboss\server\default\deploy\20qcbin.war\WEB-INF\siteadmin.xml) or an XML configuration file (ALM\conf\qcConfigFile.properties) on the server where Quality Center/ALM is installed.	
COLOR_CACHE_SIZE	Used to extend the number of colors available in the cached copy of the `ColorPalette.css` file.	
**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, 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, you must specify a value. **Note:** You must use `kEncrypt.sh` to encrypt the password. For information on how to run the `kEncrypt.sh` script, see "kEncrypt.sh" on page 500.	Default: N/A Valid values: Encrypted password in the format `#!#<Encrypted_Password>#!#`
**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, you must specify value. Set the retrieval method (FTP or SCP). See CONC_LOG_TRANSFER_PROTOCOL.	Example `applmgr`
COST_CAPITALIZATION_ENABLED	Flag to enable cost capitalization globally. You can enable this parameter only in the `server.conf` file. **Note:**HPE strongly recommends that you *not* disable this parameter after you have enabled it.	Default: `false` Valid values: `true, false`
COST_ROWS_BATCH_SIZE	Batch size of rows for the Cost Rollup Service to process.	Default: `1000`
COST_RATE_	Batch size for the Cost Rate Rule Update Service	Default: `20`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
RULE_ UPDATE_ SERVICE_ COMMIT_BATCH_ SIZE	to process.	
CUSTOM_ SERVER_INFO	Enables customization of the text displayed in the PPM Workbench title bar. For example, `com.kintana.core.server.CUSTOM_SERVER_ INFO=Welcome to PPM Workbench`	Default: N/A Valid value: Text string
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 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 retains the record of the last logon, even if `DAYS_TO_KEEP_LOGON_ ATTEMPT_ROWS` is set to 14 days.	Default: `14`
DAYS_TO_KEEP_ LOGON_ SESSIONS_ ROWS	Number of days to keep records of all user sessions.	Default: `60`
**DB_ CONNECTION_ STRING (Required if RAC is used)	Oracle RAC (Real Application Clusters) service name.	Example `K92RAC`

Table A-1. Server configuration parameters, continued

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 tables. **Note:** You must use `kEncrypt.sh` to encrypt the password. For information on how to run the `kEncrypt.sh` script, see "kEncrypt.sh" on page 500.	Example `#!#<Password>#!#`
DB_POOL_ DEADLOCK_ CHECK_ INTERVAL	Specifies database pool deadlock check interval.	Default: 10
DB_ RECONNECTIO N_CHECK_ INTERVAL	Specifies database reconnection check interval.	Default: 240
DB_SESSION_ TRACKING	May have been replaced by ENABLE_DB_ SESSION_TRACKING	Valid values: `TRUE, FALSE`
*DB_USERNAME	Name of the database schema that contains the PPM tables.	Example `knta`
DEFAULT_ COMMAND_ TIMEOUT	Number of seconds the PPM Server can try to run commands before it times out.	Default: `90`
DEFAULT_ EXPENSE_TYPE_ FOR_POSITIONS	Specifies the default value of expense type for staffing profile positions when costs are capitalized. **Note:** If the Expense Type field is not editable (because SOP 98-1 is not enabled on the parent entity of the staffing profile), Capex is still the default value regardless of how the parameter is set.	Default value: `Capex` Valid values: `CAPEX`, `OPEX`, and `SPLITXX`, with `XX` being the percentage going to `CAPEX`.
*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).	Default: `50`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	In new installations, this defaults to the slow connection setting. HPE 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_ STAFFING_ EFFORT_EDIT_ MODE	Specifies the default editable view (Gantt view or Table view) in Staffing Profile New UI. If you do not specify the parameter value, the default editable view is Gantt view.	Default value: `Gantt` Valid values: `Gantt`, `Table`
DEFAULT_USER_ DEBUG_LEVEL		
*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 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 *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 for the demand set fields cache, expressed in seconds.	Default: `360000` (seconds)

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
TIMEOUT		
DEPLOY_BASE_PATH	Deployment destination. **Note:**HPE recommends that you leave the default value unless the PPM Server directory is renamed.	Default: `server/`
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:**HPE recommends that you not change this value. It does not affect performance.	Default: `5000` (milliseconds)
DMS_DB_ENABLE_FULLTEXT_SEARCH	To enable the database fulltext search feature in document management for PPM, set this parameter to `true.` **Note:** You must create and build database indexes in advance. For details, see the *Document Management Guide and Reference.*	Default: `false` Valid values: `true, false`
DMS_FILENAME_DISPLAY_LENGTH	Determines the number of characters shown in the **References** section for the names of files attached to PPM entities.	Default: `30` Valid values: Positive integer
DMS_FILENAME_SEARCH_MAX_RESULTS	Specifies maximum number of matching items before applying filters from other search criteria, such as creation date or "Closed" status. You may need to increase this value if too many filename matching items are filtered out by very selective search criteria.	Default: `1000` Valid values: integer
DMS_INSECURE_FILE_EXTENDSIONS	Defines the file extensions that you think insecure and cannot be uploaded to PPM.	Example: `exe, com, bat, reg, jar, cmd, lnk`
DMS_INSECURE_FILE_EXTENSION_CHECK	Flags whether or not to prevent uploading files of the extensions defined in the DMS_INSECURE_FILE_EXTENDSIONS parameter. If you set this parameter to `false`, files of all	Default: `false` Valid values: `true`, and `false`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	extensions are allowed to be uploaded. If you set this parameter to `true`, files of the extensions defined in DMS_INSECURE_FILE_ EXTENDSIONS cannot be uploaded.	
DMS_XSS_ CHECK	Flags whether or not to display the document download links. If you set this parameter to `true`, the document download links on PPM pages are hidden.	Default: `false` Valid values: `true`, and `false`
DMS_ MIGRATION_ DELAY_ BETWEEN_ DOCUMENT	Determines the duration (in seconds) that a thread will wait between two documents to migrate. To lighten the load of the migration process on the PPM Server, increase the value of this parameter.	Default: `0` (seconds) Valid values: Positive integer
DMS_ MIGRATION_ DOCUMENTS_ BATCH_ SIZE	Determines the number of documents to be queued for migration on a given PPM Service node. Every time the DMS Migration Engine Service runs on a Service node, the queue of documents to be migrated is filled up.	Default: `1000` Valid values: Positive integer
DMS_ MIGRATION_ THREAD_COUNT	Specifies number of threads that will be migrating documents on a given PPM Service node.	Default: `3` Valid values: integer
EDIT_ TIMESHEET_ TABLE_ MAX_ HEIGHT	The vertical size (in pixels) of the time sheet table before the vertical scroll bar appears in the table. For example, if you set it to `300`, the vertical scroll bar appears when the vertical size of the time sheet table exceeds 300 pixels.	Default value: `1,500` Valid values: positive integer
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_ALL_ PERFORMANCE_ MONITOR	If you set this parameter to `true`, you enable the UI monitor, the portlet monitor, and the Background Services monitor, regardless of whether these three monitors are enabled or not. If you set this parameter to `false`, the UI monitor, the portlet monitor, and the Background Services monitor are enabled or disabled according to their own	Default: `false`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	parameters.	
ENABLE_ ANTISAMY	If you set this parameter to `true`, you enable the AntiSamy feature to protect PPM from potential security issues. End users cannot open hyperlinks on request details page or project details page. If you set this parameter to `false`, you disable the AntiSamy feature. End users can open the hyperlinks on the request details page or project details pages. If you enable the AntiSamy feature and still want your end users to access to specific hyperlinks, you can configure the the `antisamy-ppm.xml` file located in `<PPM_HOME>\conf` to add those hyperlinks.	Default: `true` Valid values: `true, false`
ENABLE_ BACKGROUND_ SERVICE_ MONITOR	If set to `true`, enables the background services monitor.	Default: `true` Valid values: `true, false`
ENABLE_ COMMIT_ RESOURCES_IN_ RESOURCE_ FINDER	Flags whether or not the option **Commit selected resources** is available in the resource finder. This option is used to commit assigned resources.	Default: `true` Valid values: `true, false`
ENABLE_ DISTRIBUTE_ RESOURCES_IN_ RESOURCE_ FINDER	Flags whether or not the option **Distribute demand evenly to selected resources** is available in the resource finder. This option is used to distribute demand evenly to selected resources after they are assigned.	Default: `true` Valid values: `true, false`
ENABLE_CONC_ FILES_ RETRIEVAL	Flags whether PPM server retrieves log/output files from Oracle E-Business Suite server or not. With the Oracle Apps (the extension for Oracle E-Business Suite) and the extension for Oracle Technology deployed on PPM Server, you may encounter performance issue with PPM server retrieving log/output files from Oracle EBS server. For better performance, you can add this parameter into the `server.conf` file manually and set the parameter value to `false` to switch off the log/output files retrieval. Make sure you restart the	Default: `true` Valid values: `true, false`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	PPM Server for the server configuration parameter to become effective.	
ENABLE_ CONCURRENT_ REQUEST_ UPDATES	Related to requests in 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_COST_ UPDATE_ SERVICE		Default: 3600
ENABLE_ CONNECTION_ CORRELATION	The Connection Correlation page is intended for use by HPE Software Support for troubleshooting and is disabled by default in a production environment. If the page is enabled, you access it by selecting **Open > Administration > View connection correlation** from the PPM Dashboard. To enable the **Open > Administration > View connection correlation** menu item and the Connection Correlation page, add the `ENABLE_ CONNECTION_CORRELATION` parameter to the `server.conf` file, and set its value to `true`.	Default: `false` Valid values: `true, false`
ENABLE_DB_ SESSION_ TRACKING	If set to `true`, enables a stack trace to be reported in the PPM 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`
ENABLE_ DEBUGGING_ PER_USER	Add this parameter to the `server.conf` file and set it to `true` to enable debug logging to the `serverLog.txt` file for a specific user. For more information, see "Enabling Debugging On a Per-User Basis" on page 318. **Note:** The value is case-sensitive.	Default: `false` Valid values: `true, false` (case-sensitive)
ENABLE_IPV6	Add this parameter to the `server.conf` file and set it to `true` to enable support for IPv6. **Note:** If this parameter is not present in the `server.conf` file, the system uses IPv4 by default.	Default: `false` Valid values: `true, false`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
ENABLE_ GENERIC_SSO_ TIMEOUT	If you set this parameter to `true`, the session timeout property works fine in generic SSO implementation. PPM redirects you to the sign-in page when it times out and forwards you to the Dashboard page after you click **Sign-in**. **Note:** In MLU environment, a window pops up for you to select the session language before you click **Sign-in**. If you set the parameter to `false`, the session timeout property does not work in generic SSO implementation.	Default: `false` Valid values: `true, false`
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 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 is integrated with an SSO provider such as SiteMinder, then set this	Default: `false` Valid values: `true, false`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	parameter to false. If PPM is *not* integrated with an SSO provider, HPE recommends that you keep the parameter set to true.	
ENABLE_LOG_ SUCCESS_ SERVICE_LIST	Specify the refernces codes of services. If these services are run successfully, PPM will record the success logs for them. Reference codes are separated by semicolon.	For example: `_COST_RATE_RULE_ UPDATE_SERVICE; _ COST_ROLLUP_SERVICE`
ENABLE_LW_ SSO_UI	If added to the `server.conf` file and set to `true`, enables the lightweight single sign-on (LW-SSO) user interface	Default: N/A Valid values: `true, false`
ENABLE_LW_ SSO_WEB_ SERVICE	(For integration of PPM Tasks with Service Manager RFCs only) If LW-SSO authentication is enabled, add this parameter to the `server.conf` file and set it to `true` to specify that PPM is to always use the current user to call the Service Manager Web service.	Default: N/A Valid values: `true, false`
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 `PORTLET_MONITOR_THRESHOLD`), the captured information is output to the `thresholdLog.txt` log file, which resides in same directory as the server log.	Default: `false` Valid values: `true, false`
ENABLE_	If set to `true`, enables users with the required	Default: `true`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
PROJECT_ LAUNCH_FROM_ ACTION_MENU	permission to open the PPM Workbench as a stand-alone application using Active X.	Valid values: `true, false`
ENABLE_ PROJECT_COST_ ROLLUP	Setting this parameter to `false` prevents the Cost Rollup service from running when you do the following: • create a work plan from template • Schedule summary tasks • Schedule work plans • Indent/outdent tasks • Import a work plan from another work plan Disabling this parameter helps save time and memory in the above actions.	Default: `false` Valid values: `true, false`
ENABLE_ PROJECT_ DETAIL_ VALIDATE	When you set it to `true`, the system will check the fields on **Project Details** tab before you switch to another tab. If the fields are updated without saving, a prompt will pop up for your confirmation. When you set it to `false`, the system will not do the check, and the prompt will not appear.	Default: `true` Valid values: `true,false`
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	(For integrating with Quality Center version 10.00 only) If no XML mapping file has been generated and deployed to both PPM and Quality Center, set this value to `false`. If a mapping has been deployed, to enable the integration, set the value to `true`.	Default: `false` Valid values: `true, false`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
ENABLE_QUERY_BUILDER	If set to `true`, enables the advanced "query builder" capability for searching 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`
ENABLE_RESTRICTIONS_ON_PROJ_SEARCH	To enable a restriction on the number of projects returned on project searches, set this parameter to `true`. To specify the maximum number of projects returned on project searches, set the `MAX_RESULTS_ALLOWED_ON_PROJ_SEARCH` parameter.	Default: `false` Valid values: `true, false`
ENABLE_ROADMAP_EXT_ATTR	If set to `true`, administrators can define the database view APM_ROADMAP_ENTITY_EXT_ATTR_V to add customized columns in APM roadmap hierarchy.	Default: `false` Valid values: `true, false`
ENABLE_SITE_MAP	If set to `true`, enables the **Site Map** link in the header of each page.	Default: `false` Valid values: `true, false`
ENABLE_SKIP_NAVIGATION	If set to `true`, enables the **Skip Navigation** link in the header of each page.	Default: `false` Valid values: `true, false`
ENABLE_SP_LABOR_COST_UPDATE	If set to `true`, PPM calculates staffing profile labor cost when you save project settings. PPM suggests that you do not change it to `false` unless you meet the following conditions: • The project has a large staffing profile which contains more than 200 positions. • It takes more than 2 minutes to save the project settings.	Default: `true` Valid values: `true, false`
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:	Default: `false` Valid values: `true, false`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	• 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.	
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`
ENABLE_STAFFING_PROFILE_LEGACY_VIEW	Flags whether or not Staffing Profile Legacy UI is enabled so that you can switch staffing profile page between New UI and Legacy UI.	Default: `false` Valid values: `true, false`
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`
ENABLE_TM_ALLOW_EMPTY_ITEM	Default setting used by time sheet policies to specify what occurs when users try to submit time	Default: • `RESTRICT` when

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	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 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 requires the user to enter data in the empty lines or delete those lines before submitting the time sheet. > **Note:** To improve PPM performance and to make it easier for approvers to review submitted time sheets, HPE strongly recommends using the `RESTRICT` setting. For more information about time sheet policies, see the *Time Management Configuration Guide.*	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`
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 *Time Management Configuration Guide.*	Default: `true` Valid values: `true, false`
ENABLE_TM_ WORK_ITEM_ EXTERNAL_DATA	If set to `true`, the **External Data** option is available in the following PPM Workbench windows: • On the Work Items an Activities tabs of the Tim Sheet Policy window • In the Work Item Type drop-down list (under the Dependencies section) of the Override Rule window	Default: `true` Valid values: `true, false`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
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 *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 *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 *Time Management Configuration Guide.*	Default: `true` Valid values: `true, false`
ENABLE_TMPM_ SYNC_SERVICE	If set to `true`, enables the TM-PM Sync Service, which synchronizes time sheet updates from Time Management to project work plan tasks in 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: `false` Valid values: `true, false`
ENABLE_WEB_ ACCESS_ LOGGING	If set to `true`, tells Tomcat to log all http requests received. **Note:** If this is enabled on a busy system, Web access logging can generate many log files in `<PPM_Home>/logs`. This will occupy some disk space.	Default: `true` Valid values: `true, false`
ENABLE_WEB_	Enables the PPM Web services interface.	Default: `false`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
SERVICES		Valid values: `true, false`
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 *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`
ENABLE_ WORKBENCH_ HTTP	If set to true, PPM Workbench communicates with PPM Server via HTTP(S). If set to false, PPM Workbench communicates with PPM Server via RMI(S).	Default: `true` Valid values: `true, false`
ETL_END_DATE	If PPM's Operational Reporting solution is implemented, this determines the end date for the PPM 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's Operational Reporting solution is implemented, this determines the start date for the PPM 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`
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`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
EXCEL_EXPORT_POSITIONS_LIMIT	Specifies the maximum number of positions that can be exported to Excel on the Forecast Planning page.	Default: 2,000
EXCEL_EXPORT_RESOURCES_LIMIT	Specifies the maximum number of resources that can be exported to Excel on the Search Resources page.	Default: 3,000
EXCEPTION_DETAIL_LEVEL	Determines the level of detail to include in internal error exceptions displayed in PPM. 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_PROCESSING_THREAD_COUNT	You can use this parameter to specify the thread number of the Exception Rule Service.	Default: 1 Valid values: Positive integer
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
EXCEPTIONS_RETAIN_PERIOD	Number of days non-service PPM exceptions are to be retained. Also see LOG_EXCEPTIONS_TO_DB.	Default: 14 Valid Values: Zero or higher integer
EXTERNAL_WEB_PORT	If you are using an external Web server to serve PPM 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.	Valid value: Any available port number

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	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"Using the Server Configuration Utility to Modify Server Configuration Parameters" on page 403. This port is used internally by PPM reports. There is no need to make it accessible to the network.	
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`
FONT_SIZE_OF_ GRAPHIC_ WORKFLOW	Specifies a font size for displaying larger or smaller characters in workflow layout images.	Default: `9`
FORECAST_ PLANNING_ PAGE_SIZE	Number of staffing profile positions displayed on a Forecast Planning page in Resource Management.	Default: `50`
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.

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
		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`
GZIP_ENCODING_ENABLED	Determines whether HTTP responses are compressed before they are sent to PPM 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 standard (HTML) interface, because less overall data is carried across the Internet between the client and the PPM Server. If all PPM 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. **Caution:** There is a known issue related to GZIP compression in the Oracle Java Virtual Machine (for details, see http://bugs.java.com/bugdatabase/view_bug.do?bug_id=8028216). If you see GZIP exceptions in the server log, you should set this parameter to `false` until Oracle fixes this issue.	Default: `true` Valid values: `true, false`
*HEAVY_QUEUE_CONCURRENT_CONSUMERS	Number of listeners per node to execute heavy background services.	Default: 1 Valid values:

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and 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
*HEAVY_QUEUE_REDELIVERY_LIMIT	Number of times messages are to be redelivered to the heavy service queue.	Default: `5` Valid values: Positive integer
HIDE_COST_TAB_ON_PROJECT_PAGE	This parameter is used to hide or display the Cost tab in the Project Overview page. Setting it to true, you hide the Cost tab. Otherwise, the tab is shown.	Default value: `false` Valid values: `true`, `false`
HIDE_STAFFING_TAB_ON_PROJECT_PAGE	This parameter is used to hide or display the Staffing tab in the Project Overview page. Setting it to true, you hide the Staffing tab. Otherwise, the tab is shown.	Default value: `false` Valid values: `true`, `false`
*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
HIGHLIGHT_NONWORKING_DAYS_IN_TIMESHEET	If set to `true`, for time sheets on which time is logged on a daily basis, the columns for non-working days are highlighted in color.	Default: `false` Valid values: `true, false`
*HISTORY_MENU_SIZE	Number of links to display in the **History** menu in the PPM standard interface.	Default: `10` Valid values: Positive integer
*HOURS_TO_	Number of hours to keep rows in the `KNTA_DEBUG_MESSAGES` table.	Default: `48`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
KEEP_DEBUG_ MESSAGE_ROWS	For high-volume PPM installations, a large number of rows may be generated in this table. For such installations, decrease this value accordingly.	Valid values: Positive integer
*HTTP_PORT	Port to use to communicate with the built-in HTTP server. If PPM is in stand-alone mode (that is, it is not integrated with an external Web server), then PPM clients must have access to the HTTP_PORT. If PPM 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 clients (and thus, the port need not be exposed outside of the PPM Server). **Note:** If you are integrating PPM 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.
HTTP_PROXY_ URL	URL of the HTTP proxy server used for PPM to connect to the Internet. It can be used by integration solutions or other PPM functionality.	
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`. **Note:** You must use `kEncrypt.sh` to encrypt the password. For information on how to run the `kEncrypt.sh` script, see "kEncrypt.sh" on page 500.	Default: N/A Valid values: Encrypted password in the format `#!#<Encrypted_ Password>#!#`
HTTPS_	Full path location of the keystore. This parameter	Default: N/A

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
KEYSTORE_ LOCATION	setting is required if the ENABLE_SSL_LOGIN parameter is set to `true`.	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 (TLS) the PPM Server uses. **Note:** Starting from version 9.31, SSLv3 as the HTTPS protocol is disabled in order to enhance security.	Default: `TLS` Valid values: `TLS`
HTTPS_ ENABLED_ PROTOCOLS	Specifies the HTTPS protocols in details.	Default: `TLSv1`, `TLSv1.1`, `TLSv1.2`
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
IGNORE_ NEGATIVE_ UNMET_DEMAND	If you set this parameter to `true`, PPM ignores negative unmet demand value in the calculation of forecast labor cost. If you set this parameter to `false`, PPM does not ignore negative unmet demand value (regards the negative unmet demand value as it is) in the calculation of forecast labor cost.	Default: `true` Valid values: `true`, `false`
I18N_CARET_ DIRECTION	Caret position on input fields (for example, text fields). If unspecified, the system uses the value specified for I18N_SECTION_DIRECTION.	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 standard interface.	Default: `UTF-8`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
I18N_LAYOUT_DIRECTION	Default layout direction of HTML pages in the PPM 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.	Default: `WE8ISO8859P15` Valid values: Any character set names that Oracle recognizes
I18N_REPORTS_ENCODING	Character encoding to use to generate reports in PPM. HPE 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.	Default: `ltr` Valid values: `ltr,rtl`

IMPACT_ANALYSIS_REPORT_CATEGORY	If you have enabled the attachment of Universal CMDB Impact Analysis Reports for CIs in PDF format to requests in PPM, use this parameter to specify the default category value for impact analysis reports, for example, change or operation.
	For details, see the *Demand Management Configuration Guide.*
		Default: N/A
		Valid values:
		`NO CHANGE`
		`PLAN`
		`NEW`
		`CANCEL`

IMPACT_ANALYSIS_REPORT_LANGUAGE	If you have enabled the attachment of Universal CMDB Impact Analysis Reports for CIs in PDF format to requests in PPM, use this parameter to set the default language code.
	For details, see the *Demand Management Configuration Guide.*
		Default: N/A
		Valid values:
		`de (German)`
		`en (English)`
		`es (Spanish)`
		`fr (French)`
		`it (Italian)`
		`ja (Japanese)`
		`ko (Korean)`
		`pt (Brazilian Portuguese)`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
		ru (Russian) zh (Simplified Chimese)
IMPACT_ ANALYSIS_ REPORT_NAME	Specifies a name for the UCMDB impact analysis report.	Default: DefaultImpactAnalysisReportName
IMPACT_ ANALYSIS_ REPORT_ SEVERITY	If you have enabled the attachment of Universal CMDB Impact Analysis Reports for CIs in PDF format to requests in PPM, use this parameter to set the default severity value for the impact analysis report. For details, see the *Demand Management Configuration Guide.*	Default: N/A Valid values: NORMAL WARNING(1 WARNING(2 MINOR(3 MINOR(4 MINOR(5 MINOR(6 MAJOR(7 MAJOR(8 CRITICAL
*INSTALLATION_ LOCALE	Language and country code of the PPM installation. The language code must match the PPM installation language.	Default: en_US Valid values: PPM installation language code
JAVA_CLASSES_ LOC	JRE classes location.	Example: C:/Java/j2sdk1.7/jre/lib/classes.zip
JAVA_PLUGIN_ CLASSID	Class ID for the Java plugin used for the PPM Workbench. The value of this parameter is automatically set to CAFEEFAC-0018-0000-FFFF-ABCDEFFEDCBA if the value of the parameter WORKBENCH_PLUGIN_ VERSION is JRE 8.	Default: CAFEEFAC-0017-0000-FFFF-ABCDEFFEDCBA
JAVA_PLUGIN_ PATH_IE	Web location for downloading the cross-platform Java plug-in bundle for Internet Explorer browsers. **Note:** If the parameter WORKBENCH_ PLUGIN_VERSION has a value, but you do not want the JRE packages to be automatically	Default: //java.sun.com/update/1.7.0/jinstall-7u72-windows

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	downloaded, you can set the value of this parameter to `NA`. Ed users will be directed to the link specified in JAVA_PLUGIN_PATH_NS to manually install JRE bundle. If the value of the parameter WORKBENCH_PLUGIN_VERSION is JRE 8, the values of this parameter is empty because Oracle does not provide address for automatically downloading JRE 8.	`-i586.cab`
JAVA_PLUGIN_PATH_NS	Web location for downloading the cross-platform Java plug-in installer for Netscape browsers. The value of this parameter is automatically set to `http://java.com/dt-redirect` if the value of the parameter WORKBENCH_PLUGIN_VERSION is JRE 8.	Example: `http://www.oracle.com/technetwork/java/javase/downloads/index.html`
JAVA_PLUGIN_VERSION	Earliest version of the Sun Java plug-in used to start the PPM Workbench.	Default: `1.7.0_72`
JAVA_PLUGIN_XPI_PATH	Web location for downloading the cross-platform Java plug-in installer for Firefox and Chrome browsers. **Note:** If the parameter WORKBENCH_PLUGIN_VERSION has a value, but you do not want the JRE packages to be automatically downloaded, you can set the value of this parameter to `NA`. Ed users will be directed to the link specified in JAVA_PLUGIN_PATH_NS to manually install JRE bundle. If the value of the parameter WORKBENCH_PLUGIN_VERSION is JRE 8, the values of this parameter is empty because Oracle does not provide address for automatically downloading JRE 8.	Example: `http://javadl.sun.com/webapps/download/GetFile/1.7.0_11-b21/windows-i586/xpiinstall.exe`
JDBC_DEBUGGING	Enables debugging of the Java database calls.	Default: `false` Valid values: `true, false`
*JDBC_URL	Locator for the database that contains the PPM database schema.	Default: `1521` Example:

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
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.	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. **Note:** If you want to specify a literal IPv6 address, make sure you enclose the literal address with "[" and "]" characters. For example, `jdbc:oracle:thin:@[::1]:1521:SID` **Note:**HPE strongly recommends you specify DNS name instead of literal IPv6 address.	`jdbc:oracle:thin:@DBhost.domain.com:1521:SID`
*JOB_STATUS_CLEANUP_INTERVAL	Timeout (in minutes) for cleaning the `JOB_STATUS` table.	Default: `720` (minutes) Valid values:
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`
*JSP_COMPILE_EXCLUDE_FOLDERS	If JSP_RECOMPILE_ENABLED is set to `true`, changes made to the JSP files inside the folders specified by the JSP_COMPILE_EXCLUDE_FOLDERS parameter are picked up on a running server, therefore making them visible quickly.	Default: `web/knta/rpt;web/knta/test`
JVM_OPTIONS	For HPE internal use only. Do not alter its value unless directed to do so by HPE Software Support for PPM.	Default: N/A

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
KEEP_ALIVE_INTERVAL	Frequency with which the client sends keep-alive messages to the PPM Server.	Default: `2`
KEY_STORE_FILE	Specifies the keystore file for Secure RMI. (See "Enabling Secure RMI" on page 91.)	Default: N/A
KEY_STORE_PASSWORD	Password for the keystore created for Secure RMI. (See "Enabling Secure RMI" on page 91.) **Note:** You must use `kEncrypt.sh` to encrypt the password. For information on how to run the `kEncrypt.sh` script, see "kEncrypt.sh" on page 500.	Default: N/A Valid values: Encrypted password in the format `#!#<Encrypted_Password>#!#`
**KINTANA_LDAP_ID Required if AUTHENTICATION_MODE = LDAP	PPM 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 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: `#!#<Encrypted_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:**HPE recommends that you not reset this parameter.	Example: `kintanaHome.html`
KINTANA_SERVER_DIRECTORY	Server directory location. Set this value if you have a clustered server setup.	Default: `PPM_930_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	Default: `PPM_930_ALL`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	match the name of the Windows service name.	
*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 uses the value set for LDAP_BASE_DN.	Default: N/A Examples: `CN=Users,DC=PPMAD,DC=com`
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 uses the value set for LDAP_BASE_DN.	Default: N/A Examples: `CN=Users,DC=PPMAD,DC=com`
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. **Note:** You must use `kEncrypt.sh` to encrypt the password. For information on how to run the `kEncrypt.sh` script, see "kEncrypt.sh" on page 500.	Default: N/A Valid values: Encrypted password in the format `#!#<Encrypted_Password>#!#`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
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.	Default: `636`
**LDAP_URL (Required if AUTHENTICATION_MODE = LDAP)	Comma-delimited list of LDAP URLs, which the PPM Server queries in the order specified. **Note:** • In version 9.30, make sure you specify a port number even when you use the default port number 389 for LDAP server or when you use SSL-enabled LDAP server. Otherwise, an error occurs. • 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 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:**	Example: `com.kintana.core.server.LDAP_URL_FULL=ldap://host.yourdomain.com/CN=Users,DC`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	• 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.	=yourdomain,DC =com ldap://host. yourdomai.com /OU=Users2,DC =yourdomain ,DC=com
LICENSE_KEY	License key required to use PPM core functionality.	Example 36ha5b993c1776k c6g03gjct5k7hv5c3
*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:
LOCAL_IP	Used to construct the servlet URL for use by PPM Server when the PPM Server internally invokes one of its own servlets. An example of this	Defaults to the IP address extracted from the server configuration

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	is when a report is executed. Ordinarily, this parameter should not need to be specified.	parameter *RMI_URL Example: 10.1.101.64
LOG_ EXCEPTIONS_ TO_DB	The menu item **Search > Administrative > Exceptions,** which opens the Search Exceptions page, is hidden by default. To enable it, set the LOG_ EXCEPTIONS_TO_DB parameter to true. You must have the administration license and the "server tools: Execute admin tools" access grant to access the Search Exceptions page. Except for service exceptions, all exceptions more than 14 days old are purged automatically. To reduce the load on the PPM Server, the purge is triggered for every 100th exception created. The value of 14 days can be configured using the EXCEPTIONS_RETAIN_PERIOD parameter.	Default: false Valid values: true, false
LOGIN_COOKIE_ MAX_AGE	Maximum age (and thus the expiration) of cookies used to start a PPM session.	Default: 180 (days) Valid values:
LOGON_METHOD	Method used to log on to PPM.	Default: USER_NAME
LOGON_PAGE	URL for the PPM 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
LW_SSO_CLEAR_ COOKIE	If LW-SSO authentication is enabled, add this parameter to the server.conf file to specify that PPM must clear the LW-SSO token when a user logs out of PPM. **Note:** For security purposes, HPE recommends that you always keep this parameter set to true.	Default: N/A Valid values: true, false
LW_SSO_	If LW-SSO authentication is enabled, add this	Default: N/A

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
DOMAIN	parameter to the `server.conf` file to specify the LW-SSO domain.	Valid values: LW-SSO domain Example: `xyz.com`
LW_SSO_ EXPIRATION_ PERIOD	The token for validating user logon has an expiration value that determines an application's session validity. If LW-SSO authentication is enabled, add this parameter to the `server.conf` file to specify the LW-SSO token expiration period in minutes. Configure a token expiration for each HPE application that uses LW-SSO. HPE recommends that you set the value to 60 (minutes). **Note:** The expiration value must be at least as high as that of the application session expiration value. For more information, see "Configuring PPM for LW-SSO" on page 207.	Default: N/A Valid value: Integer (minutes) Recommended value: `60`
LW_SSO_INIT_ STRING	If LW-SSO authentication is enabled, add this parameter to the `server.conf` file to specify the value of the `initString` parameter. For information about the `initString` parameter, see "LW-SSO Security Warnings" on page 201.	Default: N/A Valid value: String value at least 12 characters long
LW_SSO_ TRUSTED_ DOMAIN	If LW-SSO authentication is enabled, add this parameter to the `server.conf` file to specify one or more LW-SSO trusted domains. To separate multiple domains, use semicolons (;).	Default: N/A Valid value: Example: `xyz.com`; `abc.net`
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
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	Default: None Valid values: Integer

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	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).	
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_ CONCURRENT_ AGM_REST_ CALL_ FOR_ TIMESHEET	Specifies the maximum number of concurrent users to import time information from Agile Manager.	Default: `10` Valid values: `integer` Recommended range: `1~50`
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)
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_	Maximum number of concurrent executions allowed to run on the server. If your system is heavily	Default: `15`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
MANAGERS	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.	
MAX_QC_ALM_RELEASES_NUMBER	Specifies the maximum number of releases from ALM to be displayed on the Project Details page in PPM	Default: `500`
*MAX_LOGON_TRIES	Maximum number of logon attempts in the time interval specified for *LOGON_TRIES_INTERVAL.	Default: `3`
*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`
*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_RESULTS_ALLOWED_ON_PROJ_SEARCH	Specifies the maximum number of projects returned on project searches. To set this restriction, you must also set the ENABLE_RESTRICTIONS_ON_PROJ_SEARCH parameter.	Default: `1000` Valid values: Positive integer
MAX_RESULT_OF_AUDIT_EVENT_QUERY	Specifies the maximum number of audit event query results.	Default: `1000` Valid values: Positive integer
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` Valid values: Positive integer
MAX_RSC_IN_SG_TMAPPROVERS	Specifies the maximum number of resources allowed in a security group that serves as time approvers.	Default: `10,000` Valid values: Positive integer
MAX_WEB_ATTACHMENT_SIZE_IN_MB	Specifies maximum attachment size (in MB) for files uploaded using PPM web interface.	Default: `20` Valid values: Integer

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
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_HIDE_ INITIAL_ MESSAGE	Controls whether to hide or display the initial text in an Mobility Access email notification.	Default: `false`
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`
MSP_NOTES_ SIZE_LIMIT	Maximum size of Microsoft Project notes in PPM. Notes larger than the size specified for this parameter are truncated when MSP data is synchronized with PPM.	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 uses the Text30 field at the project level in Microsoft Project to store information about the PPM 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 to use. **Caution:** 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	Default: none

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	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.	
MULTICAST_ CLUSTER_NAME	Unique name of a PPM Server cluster. This parameter, along with the MULTICAST_IP and MULTICAST_PORT parameters, determines whether PPM starts in server cluster mode, or stand-alone mode. **Caution:** Do not configure two clusters with the same name running on the same subnet. **Caution:** 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	IP address used for exchange of heartbeat messages, cache synchronization, and cluster communication. For information about the IP addresses that you can use for `MULTICAST_IP`, see the IPv4 Multicast Address Space Registry Web page (http://www.iana.org/assignments/multicast-addresses/multicast-addresses.xml), which is maintained by the Internet Assigned Numbers Authority (IANA). **Note:** The IP address you specify for `MULTICAST_IP` must *not* include the text string `"http://"`. **Caution:** If you want to specify a literal IPv6 address here, do not enclose the literal address with `"["` and `"]"` characters. To specify a valid IPv6 multicast IP address, see http://www.iana.org/assignments/ipv6-multicast-addresses/ipv6-multicast-addresses.xhtml.	Default: N/A Valid values: `224.0.0.0` through `239.255.255.255`
MULTICAST_	Amount of time that must elapse after the	Default: `60000`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
LEASE_MILLIS	PPM Server heartbeat stops, before the PPM Server is considered terminated.	(milliseconds)
MULTICAST_NIC_IP	If the MULTICAST_NIC_IP parameter is specified in the `server.conf` file, the JGroup and MULTICAST Channel multicast sockets will bind to the NIC that you specified by using the MULTICAST_NIC_IP parameter. The value of the MULTICAST_NIC_IP parameter can be a host name or an IP address.	Valid value: A host name or an IP address
MULTICAST_PORT	Multicast port used by PPM'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
MULTICAST_TTL	Specifies TTL value in all multicast channels.	Default: `8`
**MULTICAST_WARNING_MINUTES	PPM logs warnings in the server logs if it does not detect multicast traffic from a node after a specified time interval has elapsed (even though the node can be reached on its JMS connection factory and the PPM_SERVER_INSTANCE table shows that the node is running). Use this parameter to configure the amount of time that must pass before PPM determines that a node is down. This parameter is required if you are configuring a clustered environment.	Default: `3` Valid values: An integer value greater than `0` and less than `2147483648`
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 system. You can specify NLS_DATE_<*NLS_LANGUAGE*> using all languages installed on a PPM 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, you would add the following to the `server.conf` file: `com.kintana.core.server.NLS_DATE_`	Valid values: Any Oracle-supported values

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	`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).	
NLS_SORT_<*NLS_LANGUAGE*>	Sort order of search results displayed on a multilingual PPM system. You can specify NLS_SORT_<*NLS_LANGUAGE*> values for all of the languages installed on a PPM 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, 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
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: `7` Valid values: Integer greater than 1
NUMBER_OF_FUTURE_YEARS_TO_SHOW_ON_FINANCIAL_SUMMARY	Number of future fiscal years that can be viewed in a financial summary. Time Management uses this value to determine the latest future fiscal year that users can specify in the **Fiscal Year** and **Data Range** fields on a financial summary. **Note:** This parameter is also applied to	Default: `5` Valid values: Integer between 0 and 20

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	snapshots of financial summaries, but is applied relative to the fiscal year each snapshot was taken, and not relative to the current fiscal year.	
NUMBER_OF_PAST_YEARS_TO_SHOW_ON_FINANCIAL_SUMMARY	Number of past fiscal years that can be viewed in a financial summary. Time Management uses this value to determine the earliest previous fiscal year that users can specify in the **Fiscal Year** and **Data Range** fields on a financial summary. **Note:** This parameter is also applied to snapshots of financial summaries, but is applied relative to the fiscal year each snapshot was taken, and not relative to the current fiscal year.	Default: `2` Valid values: Integer between 0 and 20
ONLINE_IMPACT_ANALYSIS_REPORT	Enable or disable the attachment of Universal CMDB Impact Analysis Reports for CIs in PDF format to requests in PPM Center. For details, see the *Demand Management Configuration Guide.*	Default: `true` Valid values: `true, false`
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 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 is to be integrated with Oracle Apps.	Default: `false` Valid values:

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	You must set this parameter to `true` for installations using Deployment Management to integrate with Oracle Apps through Object Migrator or GL Migrator. If you use Object Migrator with PPM, you must specify a value.	`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
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 database schema.	Example `d:/orant`
PACKAGE_LOG_DIR	Directory to which PPM 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`
PARTITION_NAME	Logical group name assigned to a cluster of Tomcat application servers. Any node in the server cluster	Default: N/A

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	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.	
*PASSWORD_ EXPIRATION_ DAYS	Expiration period of passwords for new users. A value of 0 indicates no expiration.	Default: `90` (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: `366` 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: integer
PLUGINS	Semicolon-separated list of plugins used by the PPM Server.	Default: N/A
*PM_CAN_ ROLLUP_ ACTUALS_ON_ SAVE	If set to `true`, enables the user to change the mode on 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 Project Management.	Default: `200`
PORTFOLIO_ MANAGEMENT_ LICENSE_KEY	License key required to use Portfolio Management. This key is available in the Autopass license key file.	Default: N/A

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
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)
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: 10000 (milliseconds)
PORTLET_ MONITOR_ PERSIST_STATE	If set to `true`, information captured by portlet monitor logs performance is saved in the table PPM_PERFORMANCE_LOG.	Default: `false` Valid values: `true, false`
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)
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 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`	Default: `false` Valid values: `true, false`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	`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.	
*QUARTZ_ WORKER_ THREADS	Number of threads to be invoked per PPM Server node to send messages.	Default: `10` Valid values: Integer greater than zero
REMOTE_ADMIN_ REQUIRE_AUTH	Determines whether user authentication is required for remote administration of the PPM instance. If set to `true`, users running `kStop.sh` to shut down the PPM Server must supply a valid PPM 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> /930/aeon /reports/`
REPORT_LOG_ DIR	Directory in which the PPM 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> /930/aeon/logs /reports/`
REPORTING_ BASE_URL	If Operational Reporting is deployed on your system, this is the base URL for your BusinessObjects server.	Example `http:\\corpqa25 :8080`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
REPORTING_ JDBC_URL	If Operational Reporting is deployed on your system, this is the locator for the database that contains the Operational Reporting database schema.	NA
REPORTING_DB_ USERNAME	If Operational Reporting is deployed on your system, this is the username for the Operational Reporting database schema.	NA
REPORTING_DB_ PASSWORD	If Operational Reporting is deployed on your system, this is the password for the Operational Reporting database schema.	NA
REPORTING_ STATUS_ REFRESH_RATE	Frequency with which report status is refreshed and displayed to the user.	Default: `5` (seconds)
REQUEST_AND_ PROJECT_ KEEPALIVE_ MAX_IDLE_TIME	Specifies the duration (in minutes) that you can stay idle on the Request Details page and Project Details page before your session starts to time out. • If the value of this parameter is grater than 0, users can stay idle in the Request Details page, Project Details page, and Edit Time Sheet page for the specified duration. After the duration ends, the session times out when the time specified in the parameter `KINTANA_SESSION_ TIMEOUT` is up. • If the value of this parameter is 0, the parameter `KINTANA_SESSION_ TIMEOUT` works in the Request Details page and Project Details page as it does in other PPM pages: your session times out if no requests are sent from the browser to the server for a duration equal to or greater than the `KINTANA_SESSION_TIMEOUT` value.	Default value: `600` Valid value: integer
REQUEST_LINK_ ALM_ ENTITY_ STATUS	Setting the parameter value to false disables the PPM Center defect workflow driven synchronization between PPM Center request status and ALM entity status. The default value is `true`.	Default value: `true` Valid values: `true`, `false`
REQUEST_LOG_ DIR	Location for Request execution log output. (Logs directory directly under the directory specified by the BASE_PATH parameter.)	Example `D:/PPM/930`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	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.	`/aeon/logs/`
REQUEST_TYPE_CACHE_TIMEOUT	Determines the stale check timeout frequency for the cache that holds mapping between parameter and tokens for Request Type and Request Header Type. **Note:** Do not change the value of this parameter.	Default: 3600 (seconds)
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. You can increase or lower the value to meet your needs. If you want to restrict the maximum number of records returned for the current search only, you can change the value in the **Limit Rows Returned To** field directly.	Default: `1000`
REQUEST_SEARCH_RESULTS_SHOW_TOTAL_NUMBER_OF_RECORDS	Flags whether or not to show total number of records for the request search results pages. The default value is `false` for better system performance.	Default value: `false` Valid values: `true`, `false`
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:** HPE 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_	Maximum number of resources that can be targeted	Default: `100`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
FINDER_ SEARCH_MAX_ USERS	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.	
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`
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 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`
*RM_DEFAULT_ PERIOD_TYPE	Default period type used to display staffing profiles and resource pool information.	Default: `month`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
		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`
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. Any user with “Edit/View Staffing Profiles” or “Edit/View All Staffing Profiles” access grant can then directly perform the following actions: • Assign any resource (in a resource pool) to the staffing profile or the work plan, or to both. • Reject unmet resource demand for one or more staffing profile lines. • Forward unmet resource demand for one or more staffing profile lines to a different resource pool. The user 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	Default: `1099`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	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` **Note:** If you want to specify a literal IPv6 address, make sure you enclose the literal address with "[" and "]" characters. For example, `rmi://[::1]:1099/` **Note:** HPE strongly recommends you specify DNS name instead of literal IPv6 address.	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. **Note:** You must use `kEncrypt.sh` to encrypt the password. For information on how to run the `kEncrypt.sh` script, see "kEncrypt.sh" on page 500.	Valid values: Encrypted password in the format `#!#<Encrypted_Password>#!#`
*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
RMO_MAX_POSITION_AMOUNT	The maximum number of positions allowed by RMO when it assigns resources to positions	Default: `50` Valid Values: Positive integer
RMO_MAX_PERIOD_AMOUNT	The longest duration (in days) of positions allowed by RMO when it assigns resources to positions	Default: `365*2` Valid Values: Positive integer
RMO_OPTIMIZE_	The maximum number of concurrent threads	Default: `1`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
CONCURRENT_ THRESHOLD	supported by RMO For example, if the value of this parameter is set to `1`, it means only one user is allowed to use the feature at a time.	Valid Values: Positive integer
RMO_OPTIMIZE_ AMOUNT_ THRESHOLD	The maximum amount of data to be handled by RMO The amount of the data for calculation = (number of positions) x (number of resources in the specified resource pool) x (number of days)	Default: `11,000,000` (100 positions, 100 resources, 3 years) Valid Values: Positive integer
RMO_OPT_ CONCURRENT_ TIMEOUT	The timeout (in milliseconds) of RMO	Default: `10,000` Valid Values: Positive integer
SCHEME_ BASED_ REDIRECT_ FILTER_ ENABLED	If set to `true`, enables the SchemeBasedRedirectFilter filter. The SchemeBasedRedirectFilter generates a correct redirect URL by adding the BASE_URL value as prefix to the redirect URL, such that every redirect URL starts with scheme (http/https) and the same base URL, and then sends to the correct target. If the parameter is not present in `server.conf`, the system would treat it as a `true` condition by default. If you do not use https or reverse proxy, you can disable the SchemeBasedRedirectFilter by setting the parameter to `false`.	Default: `true` Valid values: `true, false`
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)

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
SEARCH_ TIMEOUT	Number of seconds after which searches time out. It controls the timeout of both request search and time sheets search. 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)
SEARCH_ALL_ REQUEST_TYPE	Setting this parameter to true, you can find all request types listed in the Request Type auto-complete list on the Search Request page. However, you are still not able to see the requests that you have no access to.	Default: `false` Valid values: `true, false`
SERVER_ENV_ NAME	Name of the PPM environment that contains information about the PPM Server machine (for example, host name, user name, and password). Must be set before PPM entity migrators or commands involving secure copy can run.	Default: `KINTANA_SERVER`
SERVER_ LOCALE_ COUNTRY_CODE	Country code for the default regional settings. Add this parameter to the `server.conf` file manually and set a country code value to specify the default regional settings. The default value is `null`. Valid values are any two-letter abbreviation of a country in uppercase. For example, if you want to set the default regional settings to United States, set this parameter value to `US`. This parameter works together with SERVER_ LOCALE_LANGUAGE_CODE to ensure that PPM Center groups all PPM Center users and non-PPM Center users together when sending notifications.	Default: `null` Valid values: Two-letter abbreviation of a country in uppercase. Example, `US`.
SERVER_ LOCALE_ LANGUAGE_ CODE	Language code for the default display language. Add this parameter to the `server.conf` file manually and set a language code value to specify the default display language. The default value is `null`. Valid values are any two-letter abbreviation of a language in lowercase. For example, if you want to set the default display language to English, set the `SERVER_LOCALE_LANGUAGE_CODE` server	Default: `null` Valid values: Two-letter abbreviation of a language in lowercase. Example, `en`.

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	configuration parameter to `en`. This parameter works together with SERVER_LOCALE_COUNTRY_CODE to ensure that PPM Center groups all PPM Center users and non-PPM Center users together when sending notifications.	
SERVER_MAX_HEAP_SIZE		
SERVER_MAX_PERM_SIZE	For HPE internal use only. Do not change its value unless directed to do so by HPE Software Support for PPM.	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. If you want to specify a literal IPv6 address, make sure you enclose the literal address with "[" and "]" characters. For example, `[::1]` **Note:** HPE strongly recommends you specify DNS name instead of literal IPv6 address.	Default: `kintana` Valid values: Any valid machine name
SERVICE_RECORDS_EXPIRATION_DAYS	Specify the duration (in days) of the success logs for the background services defined in `ENABLE_LOG_SUCCESS_SERVICE_LIST`. The logs that expire the duration will be removed automatically from the Service Records page.	Default: `14`
SERVER_TYPE_CODE	Operating system on which the PPM Server is installed.	Valid values: `UNIX, WINDOWS`
SERVICE_LIST_SOURCE	Source of the available list of services that users can associate with a request (through Demand Management) or a project task (through Project Management): • If set to `uCMDB`, PPM retrieves the list of services for requests from the integrated Universal CMDB application in real time.	Default: `Validation` Valid values: `uCMDB,lookup`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	• If set to `lookup`, PPM retrieves the list of services for requests and tasks from the PPM Server.	
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 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 and Universal CMDB CIs. For more information, see the *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:
SERVICE_LIST_UCMDB_WS_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	A group of PPM Center users that no users in the system outside of this group can modify. This prevents these users from being locked out of the system and ensures that they always maintain a specific set of access rights. For HPE internal use only. Do not change its value unless directed to do so by HPE Software Support for PPM.	Default: N/A
SERVICE_RECORDS_	Determines the number of the most recent service runs to be retained.	Default: `50` Valid values:

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
RETAIN_COUNT		Integer
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 background services and instructions on how to configure them, see "PPM Background Services" on page 266.	Default: `true` Valid values: `true, false, light, heavy`
SHOW_BASE_ URL_ON_ NOTIFICATIONS	Determines whether the URL for the PPM logon window is displayed at the top of each email notification.	Default: `true` Valid values: `true, false`
SHOW_ DEBUGGING_ CONSOLE_PER_ USER	If set to `true`, allows user access to the debugging console and sets the server logging threshold on a per-user basis. Note: The value is case-sensitive.	Default: `false` Valid values: `true, false` (case-sensitive)
SHOW_ PARAMETERS_ AT_STARTUP	If set to `true`, enables the display (and logging) of all PPM server configuration parameters used during startup.	Default: `false` 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`
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 "Implementing User Authentication" on page 188.	Example `com.kintana .sc.security .auth.Site MinderSingle SignOn`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
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`
SKIP_REQUEST_CREATE_CONFIRMATION_PAGE	If you set this parameter to `true`, you can skip the Request Creation Confirmed page and directly go to the request details page after clicking **Submit** on the Create New Request page.	Default: `false` Valid values: `true, false`
SM_RFC_INTEGRATION_ENABLED	Enables the integration of PPM tasks and Service Manager requests for change (RFCs).	Default: `true` Valid values: `true, false`
SM_PASSWORD	Password that PPM Center uses to access Service Manager. You must encrypt this password by using the `kEncrypt.sh` script, which is located in the bin directory of the PPM Server. Then remove #!# from the beginning and the end of the encrypted password. **Note:** For information on how to run the `kEncrypt.sh` script, see "kEncrypt.sh" on page 500.	Default: N/A
SM_URL	Host name or IP address of Service Manager.	Default: N/A Example: `http://<Host_Name>:13080`
SM_USERNAME	User name that PPM uses to access Service Manager. This user name must include only single-byte characters.	Default: N/A Example: `admin`
SM_WEB_URL	Address of Service Manager Web tier.	Example: `http://<Host_Name>:<Port>/<WebTier_Package_File_Name>/index.do`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
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_AUTH_ PASSWORD	Encrypted password of SMTP authentication. **Note:** You must use `kEncrypt.sh` to encrypt the password. For information on how to run the `kEncrypt.sh` script, see "kEncrypt.sh" on page 500..	Default: N/A Valid values: Encrypted password in the format `#!#<Encrypted_ Password>#!#`
SMTP_AUTH_ USERNAME	Username of SMTP authentication.	Default: N/A
SMTP_PORT	Port used to connect to the SMTP server when sending notifications.	Default: `25` if SMTP_USE_SSL is `false`; `465` if SMTP_USE_ SSL is `true`. Valid values: Any available port number.
SMTP_RFC_ COMPLIANCE	If set to `true`, formats PPM 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_USE_SSL	Specifies whether or not to connect to SMTP server using SSL. **Note:** SSL is not used by default.	Default: `false` Valid values: `true, false`
SMTP_USE_ STARTTLS	Specifies whether or not to connect to SMTP server using STARTTLS. **Note:** STARTTLS is not the same as SSL. By default, STARTTLS is not used.	Default: `false` Valid values: `true, false`
SMTP_	If set to `true`, and if a Windows SMTP server is	Default: `true`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
WINDOWS_ADD_PERIOD	detected, PPM appends a period (.) to email notifications.	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
SP_RESOURCE_ROLE_RATE_ENABLE	If set to true, the forecasted labor cost of a committed resource is calculated by the cost rate of the resource role. If set to false, the forecasted labor cost of a committed resource is calculated by the cost rate of the position role. **Note:** When the resource has no role defined, and you set the parameter to true, the cost rate of the position role is used.	Default: `true` Valid values: `true, false`
*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 reports that run from command-line SQL*Plus calls. If you encounter problems running PL/SQL-based reports in PPM, set this parameter.	Example `com.kintana.core.server.SQLPLUS_VERSION=10.1.0.2`
SSH2_JSCH_DISABLE_STRICT_HOST_KEY_ CHECKING	If set to `true`, the SSH2 client will connect to the remote host even when its key is not in the list of the trusted hosts (`known_hosts` file). **Caution:** This parameter should not be set to true on a production environment.	Default: `false` Valid values: `true, false`
SSH2_JSCH_	When a value is defined in this parameter (valid file	Default: N/A

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
KNOWN_ HOSTS_ FILE_PATH	path), PPM Center will use it as `known_hosts` file to validate keys of trusted hosts it connects to. The remote servers you will connect to should be included in the `known_hosts` file (using OpenSSH format), otherwise the client cannot connect to them (unless the `SSH2_JSCH_DISABLE_STRICT_HOST_KEY_CHECKING` parameter is set to `true`, in which case PPM Center does not search for a `known_hosts` file). If this parameter is left empty, PPM Center first checks if there is a `<PPM_HOME>/known_hosts` file to use. If no, PPM Center then checks known standard locations for `known_hosts` file: • `.../.ssh/known_hosts` and `.../etc/ssh/ssh_known_hosts` under UNIX • `%USERPROFILE%\ssh\known_hosts` and `%USERPROFILE%\.ssh\known_hosts` under Windows **Note:** When a Linux user connects to a remote server using the `ssh` command on the command line and then accepts the host key when prompted, this remote machine key will be automatically added to the trusted hosts list in `.../.ssh/known_hosts`.	
SYNCH_EXEC_ INIT_WAIT_TIME	Amount of time (in seconds) after which the intermediate Request Working page opens.	Default: `4`
SYNCH_EXEC_ MAX_POLL_ TRIES	Maximum number of times to poll for completion of a request before a final message is sent to the user.	Default: `4`
SYNCH_EXEC_ POLL_INTERVAL	Time interval (in seconds) at which to poll for completion of a request after the intermediate Request Working page opens.	Default: `15`
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_	Maximum number of packages to run	Default: `10`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
MAX_THREADS	simultaneously within a release distribution. If a large number of packages in a distribution are processing, increasing this value can improve performance.	
THREAD_POOL_ MIN_THREADS	Minimum number of packages to be run simultaneously within a release distribution. See also THREAD_POOL_MAX_THREADS.	Default: `5`
**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 307. 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".
SSL_CLIENT_ SOCKET_ ENABLED_ PROTOCOL	Enables you to specify the TLS protocol of LDAPs connections.	Default: `TLS` Valid values: `TLS, TLSv1, TLSv1.1, and TLSv1.2`
TIMESHEET_ KEEPALIVE_ MAX_ IDLE_TIME	Specifies the duration (in minutes) that you can stay idle on the Edit Time Sheet page before your session starts to time out. • If the value of this parameter is greater than 0, users can stay idle in the Edit Time Sheet page for the specified duration. After the duration ends, the session times out when the time	Default value: `180` Valid value: integer

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	specified in the parameter KINTANA_SESSION_TIMEOUT is up. • If the value of this parameter is 0, the parameter KINTANA_SESSION_TIMEOUT works in the Edit Time Sheet page as it does in other PPM pages: your session times out if no requests are sent from the browser to the server for a duration equal to or greater than the KINTANA_SESSION_TIMEOUT value.	
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. HPE recommends setting this parameter to true.	Default: true Valid values: true, false
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:** HPE recommends that you use the default value of 80 characters.	Default: 80 Valid values: Integer
TM_ENABLE_GRID_RESIZE_CONTROL	If set to true, enables users to vertically resize Time Breakdown tables on time sheets.	Default: false Valid values: true, false

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
TM_ENABLE_ POLICY_ EXTENSION	Enable or disable the Time Management Custom Policy Extension.	Default: `false` Valid values: `true, false`
TM_ENABLE_ REQ_ACTIVITY_ CONTROLS	In time sheet policies, you can require users to specify activities on time sheet lines for requests of particular request types. To enable this activity restriction based on request types, you must set this parameter to `true`. (You must also select the **Use Time Management to track actuals** option from the PPM Workbench.)	Default: `false` Valid values: `true, false`
TM_ENABLE_ SHOW_WHATS_ NEW	The first time a user opens any time sheet, PPM displays a message that describes the saving of the time sheet. After this, the message is not displayed to that user again for any time sheet. To disable the display of this message, set this parameter value to `false`. **Note:** On PPM instances that support multiple languages, the message is displayed only to users who select English at log-in.	Default: `true` Valid values: `true, false`
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
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
TM_MAX_TIME_ APPROVER_ WARNING_ COUNT	The maximum number of time approvers of time sheet lines allowed by the system before it throws warning.	Default Value: `80,000,000` Valid Value: Positive integer
TM_MAX_	The maximum number of billing approvers of time	Default Value:

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
BILLING_ APPROVER_ WARNING_ COUNT	sheets allowed by the system before it throws warning.	`160,000,000` Valid Value: Positive integer
TM_MAX_TIME_ APPROVER_ ERROR_COUNT	The maximum number of time approvers of time sheet lines allowed by the system before it throws error.	Default Value: `100,000,000` Valid Value: Positive integer
TM_MAX_ BILLING_ APPROVER_ ERROR_COUNT	The maximum number of billing approvers of time sheets allowed by the system before it throws error.	Default Value: `200,000,000` Valid Value: Positive integer
TM_ REEVALUATE_ COUNT	Specifies the amount of the Reevaluate Time Sheet information handled by the Evaluate TM Approvers service.	Default Value:`1,000,000`
TMG_ CONFIGURABLE_ FILTERS_REF_ CODE	Stores the reference code for the 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:** HPE 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 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 Deployment Management. In a server cluster, all servers must be able to	Example `D:/<PPM_Home> /930/ionia /transfers/`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	access and share the specified directory.	
TZ_IS_TIME_ ZONE_ DEFAULTED		Default:
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 *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 *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.
UCMDB_ PASSWORD	Encrypted password for the Universal CMDB user. For more information, see the *Solution Integrations Guide*. **Note:** You must use `kEncrypt.sh` to encrypt the password. For information on how to run the `kEncrypt.sh` script, see "kEncrypt.sh" on page 500.	Valid values: Encrypted password in the format `#!#<Encrpted_ Password>#!#`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values

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 *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 *Solution Integrations Guide.*	Default: N/A Example: `Admin`
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	Universal CMDB user password for logging in through Web service. **Note:** You must use `kEncrypt.sh` to encrypt the password. For information on how to run the `kEncrypt.sh` script, see "kEncrypt.sh" on page 500.	Default: N/A Valid values: Encrypted password in the format `#!#<Password>#!#`
UCMDB_WS_	Used for integration with Universal CMDB for	Default: N/A

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
USER	service list retrieval. Universal CMDB user name for logging in through Web service.	Example: `Admin`
UI_MONITOR_ THRESHOLD	If ENABLE_UI_MONITOR is set to `true`, this parameter determines the time 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: `true` Valid values: `true, false`
USE_REGION_ OF_ RESOURCE_ POOL _FOR_ POSITION_ FORECAST_ COST	If you set this parameter to true, PPM uses the region of the resource pool specified for the position to pick up cost rule when calculating its forecast labor cost. Otherwise, PPM uses the region of the staffing profile to pick up the cost rule. If you change the parameter value, the new value will be used in the calculation of forecast labor cost the next time when Staffing Profile Financial Summary Sync service runs.	Default: `false` Valid values: `true, false`
USER_SEARCH_ RESULTS_MAX_ ROWS	The maximum number of users displayed in User Management Console.	Default: `200` Valid values: positive integer
USE_SERVER_ LOCALE_FOR_ NOTIFICATIONS	Flags whether or not to check notification recipient regional settings. Setting the value to `true` ignores users' regional settings when sending email notifications, and uses the values of server configuration parameters SERVER_LOCALE_ COUNTRY_CODE and SERVER_LOCALE_ LANGUAGE_CODE instead. This ensures that recipients are not split into different groups according to their regional settings.	Default: `false` Valid values: `true, false`
USER_ PASSWORD_ MAX_LENGTH	Maximum number of characters in user passwords.	Default: `16`
USER_ PASSWORD_	Minimum number of digits in user passwords.	Default: `1`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
MIN_DIGITS		
USER_ PASSWORD_ MIN_LENGTH	Minimum number of characters in a user password.	Default: `8`
USER_ PASSWORD_ MIN_SPECIAL	Determines the minimum number of non-alphanumeric characters that user passwords must contain.	Default: `0`
USER_ PASSWORD_ MIN_ UPPERCASE_ LETTERS	Minimum number of uppercase letters in a user password.	Default: `1`
USER_ PASSWORD_ MIN_ LOWERCASE_ LETTERS	Minimum number of lowercase letters in a user password.	Default: `1`
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> /930/aeon /logs`
VISUALIZATION_ EXEC_TIMEOUT	Length of time (in seconds) that resource management visualizations can run before they time out.	Default: `180`

**WEB_CACHE_ DIR	Specifies web cache directory.
	Add this parameter to the `server.conf` file if it is not there and make sure to use a shared folder in clustered configuration so that whenever a chart in portlet is loaded by PPM Dashboard, it creates charts pictures into this shared folder. Then, the Directory Cleanup Service can automatically clean up the older pictures files.
	Required only if all the nodes of the cluster are not located on the same physical machine AND in the
		Example:
		`//<IP_Address> /<Local_Drive_ Letter>$ /<Shared_Folder>/`
		or
		`<File_Server_Name>$ /<Shared_Folder>`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
	same <*PPM_Home*> folder.	
WF_DISABLE_DELEGATE_DECISION_STEP	By adding this parameter to the `server.conf` file and setting it to `true`, you can hide the **Delegate Decision** button for workflow steps where the **Decision Required** option is set to **At Least One** or **All**. If this parameter is not set or set to `false`, the **Delegate Decision** button will be displayed.	Default: `false` Valid values: `true`, `false`
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`
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 `1.7.0_72`
WORKBENCH_MAX_HEAP_SIZE	Specifies maximum available memory (in MB) for the PPM Workbench.	Default: `256`
WORKBENCH_SERVICE_URL	The address of PPM Server that PPM Workbench communicates with via HTTP(S). If not set, PPM Workbench communicates with the server specified in BASE_URL. The format of this parameter is the same as BASE_URL. This parameter is only effective when "ENABLE_WORKBENCH_HTTP" on page 425 is set to `true`.	Default: null
WORK_PLAN_RESOURCE_AVAILABILITY_DAYS_LIMIT	Specifies resource availability days limit for a task, including the resource's non-working days. A schedule warning appears if the total of the value entered in the **Scheduled Duration** field and the resource's non-working days is bigger than the limit you specified.	Default: `2000` Valid values: An integer value greater than `0` and less than `2147483648`
WS_UPDATE_CLOSED_AND_	If set to `true`, lets Web services update closed and canceled requests.	Default: `false`

Table A-1. Server configuration parameters, continued

Parameter Name (*Required, **Required If)	Description, Usage	Default and Valid Values
CANCELED_ REQUESTS		Valid values: `true, false`

a. For details about this parameter, see the *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

The following table lists and provides descriptions of the PPM Server configuration parameters related to the PPM Dashboard in the PPM standard interface. These parameters, like those listed in "Using the Server Configuration Utility to Modify Server Configuration Parameters" on page 403, 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

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.	Valid values: Same value as set for the *BASE_URL parameter
Character-Encoding	Specifies the coding to use for text displayed in the PPM Dashboard.	Default: `UTF-8`
Chart-Width-Restriction-Enabled	Setting this to `true` enables PPM Dashboard chart portlets to maintain their size, even with large numbers of data points. A chart maintains its size as long as the data in the chart is readable. For very large numbers of data points,	Default: `true` Valid values: `true, false`

Table A-2. PPM Dashboard-related server configuration parameters, continued

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
	the chart eventually grows.	
DSH - Center Name	PPM Dashboard center name.	Default: PPM
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`
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`
dashboard.Asynchronous-Loading-Enabled	Setting the value to `true` enables asynchronous rendering of the Dashboard page.	Default: `true`
dashboard.Pivotdataset-Max-Cells	Determines the maximum number of cells (number of rows * number columns) in the List display mode of a pivot table portlet.	120,000
dashboard.Pivotdataset-Max-Distinct-Cells	Determines the maximum number of unique aggregations (product of numbers of unique values in each column) in a pivot table.	10,000
dashboard.Pivotdataset-Max-Distinct-In-Column	Determines the maximum number of unique values in any column of a pivot table.	50
dashboard.Pivotdataset-Max-Rows	Determines the maximum number of rows in the List display mode of a pivot table portlet.	25,000
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	Default: `false`

Table A-2. PPM Dashboard-related server configuration parameters, continued

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
	Favorites option.	Valid values: `true, false`
Fonts-Directory-Path	Directory path of the fonts used by PPM Dashboard.	Example: `C:/Fonts/opt/fonts`
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 in a web browser. (Because only the left-to-right (ltr) page layout direction is supported, you can leave the value unspecified.)	
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. In order for the PDF export feature to work when the PPM Server is configured to use SSL, the PPM Dashboard must have an HTTP port that can be accessed locally by ICE browser.	Default: N/A
pdf-render-timeout	Specifies the timeout (in seconds) for the export-to-PDF mechanism. This timeout is used while calling ICEbrowser. Should be used only for customers who cannot export to PDF because of database queries that return a large volume of data.	Valid values: Any integer
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

Table A-2. PPM Dashboard-related server configuration parameters, continued

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
Page-Auto-Refresh-Disabled	If set to `true`, disables the auto-refresh capability in the PPM Dashboard.	Default: `false` Valid values: `true, false`
Page-PDF-Export-Disabled	If set to `true`, removes the Export to PDF icon from 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
Printout-Banner-Text	Determines the text to be displayed in the header of printouts (for example, the PDF header banner). If you specify no value, the center name is used.	Valid values: Any text string
SQL-Debug	If set to `true`, enables SQL debugging in the PPM Dashboard. **Note:** Unless you require detailed PPM Dashboard debug logs for support, keep this parameter set to false.	Default: `false` Valid values: `true, false`
Smtp-Port	Specifies an SMTP port for the PPM Dashboard. **Note:** HPE recommends that you leave the value for this parameter unspecified.	Default: N/A
Smtp-Server	Specifies an SMTP server for the PPM Dashboard. **Note:** HPE recommends that you leave the value for this parameter unspecified.	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 version

Table A-2. PPM Dashboard-related server configuration parameters, continued

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
Time-Zone	Specifies the time zone used in the PPM Dashboard. **Note:** HPE recommends that you leave the value for this parameter unspecified.	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`
Web-Server	If the export-to-PDF functionality does not work and PPM is configured with an IIS or Apache Web server, set this value to try to restore the export-to-PDF functionality.	Valid values: IIS, Apache

Logging Parameters

The following table 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.

Note: 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 below 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

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:

Table A-3. Logging parameters, continued

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
		`true, false`
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. **Caution:** Use of class filtering degrades performance.	Default: `DEBUG`
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)	Default: `0` Valid values: `3, 0, 001,` and `010`

Table A-3. Logging parameters, continued

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
	• 001 Include context information, if provided • 010 Include stack trace for log messages without exception	
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 312.	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
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 312.	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, continued

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_SMTP_ LOGGING	If set to `true`, mail notification for specified server logs is enabled.	Default: `false` Valid values: `true, false`
ENABLE_WEB_ ACCESS_LOGGING	Determines whether or not information sent to the internal PPM Web server (Tomcat) is logged.	Default: `false` Valid values: `true, false`
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`

Table A-3. Logging parameters, continued

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
LOG_LAYOUT	Determines the layout format of the log files.	Default: `XML` Valid values: `TEXT, XML`
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, HPE 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.SSS z}: %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, continued

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>` **Caution:** This parameter is designed for advanced troubleshooting. *Do not* modify its value unless HPE Software Support engineers advise you to do so.	Default: `com.kintana. crt.request, DEBUG` Valid values: `DEBUG, INFO, ERROR`
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`
**SMTP.To Required if ENABLE_ SMTP_LOGGING is set to `true`.	The recipient of the notification.	Default: N/A Valid values: Email address
SMTP.From	The sender of the notification. If not specified, the email address of the sender is derived from EMAIL_ NOTIFICATION_SENDER in the `server.conf` file.	Default: N/A Valid values: Email address

Table A-3. Logging parameters, continued

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
SMTP.Subject	The subject of the notification mail.	Default: N/A
**SMTP.Filter.RegExToMatch Required if ENABLE_SMTP_LOGGING is set to `true`.	The keyword or regular expression to be monitored.	Default: N/A Valid values: Regular expression For example, `OutOfMemoryError|\w+\d{2}|Exception`

SMTP.Filter.LevelMin	The lowest log level to be monitored.	Default: N/A
		Valid values:
		`trace`
		`debug`
		`info`
		`warn`
		`error`
		`fatal`

SMTP.Filter.LevelMax	The highest log level to be monitored.	Default: N/A
		Valid values:
		`trace`
		`debug`
		`info`
		`warn`
		`error`
		`fatal`

SMTP.delayBetweenChecksInSeconds	The time interval (in seconds) to check message queues.	Default: `10` Valid values: `> 0`
SMTP.SMTPDebug	If set to `true`, enables displaying debug information when mail notification is sent out.	Default: `false` Valid values: `true, false`
SMTP.BufferSize	The buffer queue length.	Default: `512` Valid values: `> 0`
SYSTEM_THRESHOLD	Determines the system-wide logging threshold. IF statements wrapping the log requests check this setting. **Caution:** This parameter is designed for	Default: `ERROR` Valid values: `INFO, ERROR, DEBUG`

Table A-3. Logging parameters, continued

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
	advanced troubleshooting. *Do not* modify its value unless HPE Software Support engineers advise you to do so.	
USER_ THRESHOLD:	Used to set logging threshold for individual users, as follows: `<Username>, <Log_Level>` To enable the debug console for multiple users, add `USER_THRESHOLD` on a separate line of the `logging.conf` file for each user.	Default: N/A

LDAP Attribute Parameters

The following table 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.

Caution: 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`	Format: `ColumnName = LDAPAttribute`

Table A-4. LDAP attribute parameters, continued

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
	• EMAIL_ADDRESS = mail • PHONE_NUMBER = telephoneNumber • DEPARTMENT_MEANING = departmentNumber • LOCATION_MEANING = locality • MANAGER_USERNAME = manager • USER_DATA1 = mail • VISIBLE_USER_DATA1 = mail • DISTINGUISHED_NAME= distinguished name • LDAP_USERNAME= LDAP username 2. Add a placeholder parameter to the LdapAttribute.conf file that will add a value to the **FIRST_NAME** and **LAST_NAME** fields.	
RSC_ RESOURCES_ 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 • 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: USERNAME = sAMAccountName
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'

Table A-4. LDAP attribute parameters, continued

Parameter Name (*Required)	Description, Usage	Default, Valid Values, Example
LDAP_USER_ OBJECTCLASS	Object class attribute for a user on the LDAP server.	Default: `person`

Appendix B: Server Directory Structure and Server Tools

This appendix addresses the ppm940 and <*PPM_Home*> directories and the scripts and tools they contain. The ppm940 directory (the installation directory) contains two subdirectories that relate to the Oracle database schemas: ppm940/sys and ppm940/system.

The <*PPM_Home*> directory (the install directory for PPM) 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.

ppm940/system

The ppm940/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 tables
<*Index_Tablespace*>	represents the tablespace used to store PPM 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 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.

kBuildStats.sh

In old versions of PPM, the kBuildStats.sh script was used to instruct Oracle to gather statistics about the PPM database schema.

Since Oracle offers sufficient tools to gather statistics about the database schema, this script is kept in code, but not used anymore.

kCalculateHealth.sh

The kCalculateHealth.sh script computes the classpath for accessing the PPM 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 users' names.

Run the script as follows:

```
sh ./kChangeNameDisplay.sh [-full_name_format] <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.

> **Note:** 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.

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), HPE 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 514) 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
    ```

> **Note:** Starting from PPM version 9.20, you can no longer use this tool to migrate DMS. To migrate your DMS, use the DMS Configuration tool from the Administration Console instead.

kConfigCheck.sh

The `kConfigCheck.sh` script performs sanity checks on a specified PPM Server instance defined in the `server.conf` file. The script returns a status of either `"Node <PPM Server> sanity failed"` or `"Node <PPM Server> sanity passed"`.

Run the script as follows:

```
kConfigCheck.sh -name <PPM Server>
```

If you run the script without specifying a PPM Server, the script performs a sanity check on the first node defined in the `server.conf` file.

If the sanity check fails, the PPM Server instance does not start up.

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 Deployment Management Extensions, PPM Best Practices, language packs, hot fixes, and PPM 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 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.

   > **Note:** 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
   ```

The following table 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 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.
-hotfix	Deploys hotfixes.
-f	Reinstalls an existing deployment.
-lang	Installs a language pack after PPM installation or upgrade. For information about how to install a language pack, see the *Multilingual User Interface Guide*.
-k	Includes the PPM database schema password in the command. Automates command execution but may be a security risk.
-u	Includes the PPM user name in the command.

Table B-1. Key command-line options for kDeploy.sh, continued

Option	Description
-p	Includes the password for the PPM 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` file, if a new version exists.

The `kDeploy.sh` script performs conflict check before installing a hotfix:

- If the hotfix files do not exist on the current instance, running the deployment command successfully installs the hotfix.
- If the hotfix is checked to be conflicting with some files on the current instance, the installation would fail.

  To address the conflict issue and to continue the installation, you can contact HPE Software Support who would provide a `.lst` file that contains the latest MD5 code. You can then run the following command:

  `sh ./kDeploy.sh –supersede <hotfix_bundle_name> <hotfix_bundle_name>.lst`

  If you encounter further conflict issues when running this deployment command, contact HPE Software Support for solutions.

After the hotfix is deployed successfully, the `kDeploy.sh` tool continues to verify whether or not the hotfix is deployed correctly.

Deploying Hotfixes with kDeploy.sh

To deploy a hotfix, run the following command:

`sh ./kDeploy.sh -hotfix <Deployment_Hotfix>`

where `<Deployment_Hotfix>` is the hotfix bundle name.

For example, if the bundle name is `912-debug-QCIM1L12345.jar`, then the corresponding deployment command is as follows:

`sh kDeploy.sh -hotfix 912-debug-QCIM1L12345.jar`

Viewing hotfix deployment information

To view the hotfix deployment information after you run `kSupport.sh`:

1. Navigate to and open the `<kSupport_zip_file>/ppm/index.html` file.
2. Click the **Patches Information** link.

   The `patches.html` file opens and displays the following information:

   - Version of PPM instance
   - Service packs applied in current PPM version
   - All files in the PPM file system
   - Additional files applied in the current PPM instance, including SQL files and imported entities (packages, request types, requests, and so on)
   - Service packs deployed in your current PPM version
3. Check the details in the section(s) of interest.

Undeploying Hotfixes

Running the command `sh ./kDeploy.sh –undeploy <Deployment_Hotfix>` undeploys a hotfix, and this command would perform files dependency check before undeploying the hotfix:

- If the hotfix files are not included in any other hotfix, running the command undeploys it directly.
- If some files in the hotfix are also included in the hotfixes that are installed later than it, you should undeploy those hotfixes first by running the above command.

> **Note:** In undeploying a hotfix, running the undeploy command rolls back only DB changes and file changes. It does not roll back DML (Data Modification Language) changes.

Redeploying Hotfixes

If you have successfully deployed a hotfix, and you want to redeploy it, you can run the deployment command `sh ./kDeploy.sh –hotfix <hotfix_bundle_name>` again.

When running this command for redeploying purpose, it would:

1. Perform conflict check.
2. Perform files dependency check.

3. Undeploy the hotfix.
4. Deploy the hotfix again.

Limitations

The `kDepoly.sh` script contains the following limitations:

- Only DDLs which are used to update a view or a package are fully supported. Other SQLs, like DML or DDL used for altering a table can be deploy successfully but cannot be rolled back by this tool.
- The files contained in the `fs_home` directory are not fully supported by this tool. You can deploy those files using the script, but cannot roll back them.
- This script does not support a patch that is completely composed of SQLs. You cannot use this tool to deploy, undeploy, or redeploy such patches.

kDevMigratorExtract.sh

The `kDevMigratorExtract.sh` script uses the content migrator to extract a content bundle from the PPM instance.

> **Note:** 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 administrator user name
-password	PPM administrator user password
-dbpassword	System database password
-itghome	Directory where PPM is installed (`<PPM_Home>`)
-action	Action to perform. Specify either `search` or `bundle`.
-entityId	Entity ID
-keyword	Search keywords
-filename	Content bundle filename

Option	Description
-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 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 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 Portfolio Management, Financial Management, and 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 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 Financial Management

For more detailed information and instructions on how to set up these functions, see the *Generating Fiscal Periods* document.

kGenJavaDump.sh

The `kGenJavaDump.sh` script is used to generate Java dumps for starting PPM server only.

The generated Java dumps are collected in the `<PPM_HOME>\bin\support\javadumps` folder:

- The heap dump files are in the following format:

  `<NODE_NAME>_<PID>_<TIMESTAMP>_heap.hprof`

- The thread dump files are in the following format:

  `<NODE_NAME>_<PID>_<TIMESTAMP>_thread.hprof`

kGenTimeMgmtPeriods.sh

The `kGenTimeMgmtPeriods.sh` script is used in 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 <number_of_periods> <start_date>`

where

`<number_of_periods>`	represents the number of time periods to create for a specific period type. Valid value: integer
`<start_date>`	represents the date from which the periods are to be populated. Date format: `MM/DD/YY`

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 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 user requests a page in the PPM 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 interface.

If any JSP is modified, you need to stop PPM, run the `kJSPCompiler.sh` script, and then restart PPM.

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:

- Detailed information for each license installed on your machine, such as
  - PPM Center module that is licensed for use, including the ID, version, and description
  - IP address of the license Start date and expiration of the license
  - Availability capacity of the license
- A summary of all the licenses installed on your machine
  - Capacity for different PPM Center modules
  - Expiration dates of licenses for different PPM Center modules
  - Licensed IP for different PPM Center modules

To use the license reader, run the following command:

`kLicenseReader.sh [-filename <`*`Autopass_License_File_Name`*`>] [-filepath <`*`Autopass_License_File_Path`*`>] [-help]`

If you do not specify the file name, the license reader provides information of all the licenses installed on your instance. If you do not specify the file patch, the license reader uses the default file path `<`*`PPM_Home`*`>/conf`.

> **Note:** You can only use the script to read the license information. You cannot use it to modify the

> license information.

kLicenseInstall.sh

The `kLicenseInstall.sh` script is added in PPM version 9.30.

After you have activated and generated an Autopass license (a .dat file or several .dat files that HPE sent to you) from the HPE Licensing for Software portal (http://enterpriselicense.hpe.com/redirector/home), and that you have successfully installed or upgraded to PPM version 9.30, use the `kLicenseInstall.sh` script to install the Autopass license key file(s). The newly installed Autopass license keys override the trial or evaluation licenses.

To install an Autopass license key file, run the command as follows:

`sh ./kLicenseInstall.sh <Autopass_License_File_Path>`

where, `<Autopass_License_File_Path>` is the full path of the Autopass license key file you save on your machine.

For example,

`sh ./kLicenseInstall.sh C:\AutopassLicense.dat`

> **Note:** Before you run the `kLicenseInstall.sh` script, make sure you already activated and generated an Autopass license from the HPE Licensing for Software portal: http://enterpriselicense.hpe.com/redirector/home.
>
> For instructions about activating and generating an Autopass .dat license file, see "Activating and Generating Autopass License" on page 86.

kMigratorExtract.sh

The script `kMigratorExtract.sh` is used in PPM 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 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 NNNNNNNNNNYYNNNNNNN
```

> **Caution:** 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>`

`nochange:`	Do not change reference data
`install:`	Install 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 Time Management to Project Management. Any actuals not entered into Project Management using Time Management are replaced by actuals from 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 *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.

Run the script as follows:

```
sh ./kRunCacheManager.sh
```

Select the number for the corresponding entity cache (request types, validations, and so on) that you want to flush. Running this script on any one node clears out the cache on all nodes. 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.

> **Note:** 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.

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 a Single-Server System" on page 77.

> **Note:** For PPM Servers participating in the same cluster, HPE 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>][-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.

> **Note:** 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 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 a Single-Server System" on page 77.

kSupport.sh

Use the `kSupport.sh` script to gather information useful to HPE Software Support in diagnosing system problems, and create a Zip file with a timestamp in the `support/zipfiles` directory.

The `kSupport.sh` script gathers the following information:

- Install and upgrade logs
- Server logs (with the option for a date range)
- JDBC logs
- Deploy logs (for the installation of patches and Deployment Management Extensions)
- Configuration files
- Server reports
- Database information for troubleshooting
- Invalid PPM schema objects and database indexes
- File system information
- Invalid PPM Center schema objects and invalid database indexes
- Bill of Materials (BOM) information

You can run the `kSupport.sh` script from the Administration Console. For information, see "Gathering Information for HPE Software Support from the Administration Console" on page 296.

Running SQL Scripts with kSupport.sh

You can use the `kSupport.sh` script to run SQL scripts that the HPE Software Support provides for gathering database information for troubleshooting purposes. You can browse for SQL scripts to run, or manually paste SQL scripts in a text area to run. The server information collection summary page, provides information about the number of SQL commands that are prevented from running.

All of the SQL scripts you select to run are copied into the `<kSupport_Zip_ File>/ppmc/data` directory, and the SQL script content you entered in the text area is concatenated into a CSV-formatted file named `runsqls<Time_Stamp>.sql` in the `<kSupport_Zip_File>/ppmc/data` directory.

> **Caution:** If the resulting SQL file is larger than 100 MB, a java.lang.OutOfMemory error may be logged in the `<PPM_Home>/bin/support/ppmc/ksupport.log` file. To correct the problem, split the source SQL script into multiple files, and then run them again in batches.

You can run SQL scripts in silent mode, as follows:

```
sh ./kSupport.sh -silent [-runsql <SQL_File_Path>]
```

Listing Invalid Database Schema Objects and Database Indexes

The kSupport output `database.html` file lists invalid PPM schema objects and invalid database indexes. Running kSupport.sh adds the following two sections to the `<kSupport_zip_file>/ppmc/database.html` file:

- **Selecting invalid objects in DB** section. This lists the types and names of all invalid objects in the database. The list includes objects of the following types:
  - SEQUENCE
  - PROCEDURE
  - PACKAGE
  - PACKAGE BODY
  - TRIGGER
  - TABLE
  - INDEX
  - VIEW
  - FUNCTION
- **Selecting invalid indexes in DB** section. This section lists invalid database indexes. For example, some INDEX objects are of `UNUSABLE` status in `user_indexes` but are of `VALID` status in `all_objects`.

Bill of Materials Information

The ksupport tool performs a BOM check and places the findings in the Bill of Materials Information page, which includes the following three sections:

- **Files missing.** This section lists files that should exist, but do not.
- **Files unwanted.** This section lists files that are not shipped with PPM and should not exist.
- **Files changed.** This section lists files that have changed since they were shipped with PPM.

To access the Bill of Materials Information page, do one of the following:

- Navigate to and open the <*kSupport_zip_file*>/ppmc/index.html file, and then click **Bill of Materials Information** link.
- Navigate to and open the <*kSupport_zip_file*>/ppmc/md5filelist.html file.

Running Mode for kSupport.sh

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 -custom
```

To run in console mode:

```
sh ./kSupport.sh -console
```

To run in silent mode:

```
sh ./kSupport.sh -silent -customer <Company_Name> -sr <Service_Request_Number>
```

kTestSiteMinder.sh

Use the kTestSiteMinder.sh script to determine whether SiteMinder is correctly configured with your PPM 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 database. You can run the script either before or after you start to use 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 Time Management in PPM.

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 493). 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 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 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 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 77.

<PPM_Home>/pdf

The `pdf` subdirectory contains all documentation files for PPM (to view the documentation in PDF format in the `<PPM_Home>/pdf/manual/Content/PDFs` subdirectory, you need Adobe Reader).

You can also access product documentation:

- From **Help > Help Center** in either the PPM standard interface or the PPM Workbench interface
- The HPE Software Product Manuals Web site (https://softwaresupport.hpe.com)

> **Note:** PPM version 9.20 introduced a new framework for the PPM Documentation Library, which is contained in the <*PPM_Home*>/pdf/manual subdirectory. The new default PPM Documentation Library is not customizable.
>
> However, you can change back to the old "customizable" PPM Documentation Library. For more details, see the *Customizing the Standard Interface*.

<*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 Deployment Management packages and 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>/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 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 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 370.

<*PPM_Home*>/utilities

The `utilities` subdirectory contains scripts for granting the SYS-level privileges to the PPM database schema, and for creating the PPM users required for installing or upgrading PPM. It also contains diagnostic tools for troubleshooting PPM 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 251.

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.

Send documentation feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on Installation and Administration Guide (Project and Portfolio Management Center 9.40)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to your_IE_team_PDL@hpe.com.

We appreciate your feedback!