HP Project and Portfolio Management Center

Software Version: 7.5

PPM Center Performance Best Practices Guide

Document Release Date: August 2008

Software Release Date: June 2008

Legal Notices

Warranty

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

The information contained herein is subject to change without notice.

Restricted Rights Legend

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

Copyright Notices

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

Intel®, Intel® Itanium®, Intel® Xeon™, and Pentium® are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

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

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

Microsoft Vista® is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.

Oracle® is a registered U.S. trademark of Oracle Corporation, Redwood City, California.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

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

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

To check for recent updates, or to verify that you are using the most recent edition of a document, go to: h20230.www2.hp.com/selfsolve/manuals.

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

Support

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

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

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

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

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

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

Contents

1 Introduction

This document provides best practices tips and suggestions for optimizing the performance for your PPM Center version 7.5 implementation. The recommendations and examples cited here are based on real-world experiences of HP PPM Center developers and customers alike.

While performance can be affected by numerous variables and conditions, this document focuses on several key areas that can significantly affect performance, including:

- PPM Server configuration and optimization
- Cluster configuration
- Services isolation

This document discusses performance best practices and it should be used with the *System Administration Guide and Reference*. The *System Administration Guide and Reference* contains the detailed information on how to tune and configure PPM Center, including details on how to configure your Oracle® database for use with PPM Center.

For hardware and software requirements, see the *System Requirements and Compatibility Matrix*.

2 Deployment Recommendations

Setting Up and Testing Your Deployment

Your deployment options and configuration choices can have a significant impact on the performance of PPM Center. HP recommends that you set up a test implementation to determine the best deployment for your implementation, following the recommendations in this chapter.

Choosing a PPM Center Configuration

PPM Center software is flexible and can be configured to meet the demands of your company. Your configuration choices can improve user load distribution, transaction capacity, and system performance. The degree of improvement depends on the number of PPM Servers in the cluster and the resources available to each.

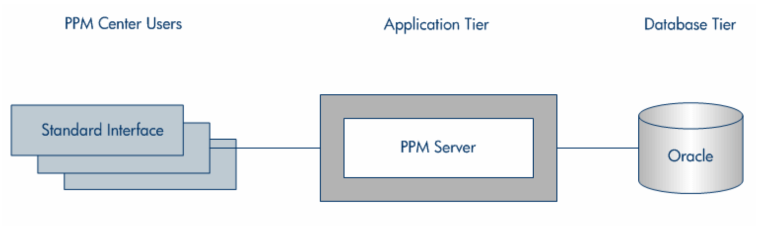
PPM Center configurations require:

- One or more clients
- One or more application servers
- A single Oracle relational database

PPM Center has the following main configurations:

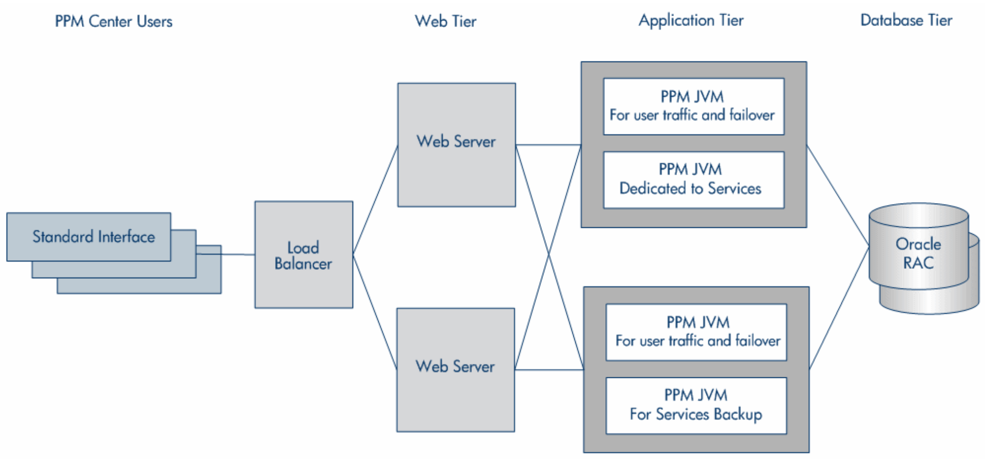
- **Single Server** is a configuration option with one database, one PPM Server, and any number of clients.

Figure 2-1. Single server configuration



- **Server Cluster** is a configuration option with a database, several PPM Servers, and any number of clients.

Figure 2-2. Server cluster configuration



With the server cluster configuration you have many configuration possibilities. The number of machines you use and where you place the various components is up to you and the needs of your company. There are no additional PPM Center licensing charges for using multiple PPM Servers or Web servers.

For guidance in selecting an option, see the *System Administration Guide and Reference*.

Balancing Network Traffic

When more than one PPM Server is supporting user traffic, the incoming PPM Center requests must be load balanced across the PPM Servers using one of the following:

- **Software load balancing** using the Tomcat connector running with a PPM Center compatible external Web server
- **Hardware load balancing** using a physical hardware load balancer

Deploying a Web Tier

Follow the recommendations in this section if you need to use an external Web server for one of the following reasons:

- Firewall requirements
- Single sign-on (SSO)
- Web-tier clustering

If redundancy is required at the Web tier, a hardware or software load balancer is required to distribute incoming traffic.

If SSO is not a requirement for your implementation, using hardware load balancers is recommended.

If your implementation takes advantage of an external Web server, HP requires that you use the following optimizations:

- Enable compression

  For example, on Apache servers, enable the `mod_deflate` module. On IBM IIS servers enable native compression. See "Enabling Dynamic Compression" in the *System Administration Guide and Reference*.

- Enable keep-alive

  Some Web servers disable keep-alive.

  If you are running Apache server and `mod_ssl` to provide HTTPS support you must make sure keep-alive is enabled. See "Enabling Dynamic Compression" in the *System Administration Guide and Reference*.

- Enable cache-control headers
- Disable cache-header interference

Deploying an Application Tier

On the application tier, the recommendations in the following sections can help improve performance.

Isolating Services

For medium-to-large PPM Center deployments, HP recommends that you devote one PPM Server on a single Java™ Virtual Machine (JVM) to processing PPM Center services.

By dedicating your services to one JVM, you should be able to:

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

For optimal PPM Center performance, for PPM Servers devoted to user traffic, use `server.conf` to turn off the PPM Center services.

Scheduling Services

There may be more services running than you need, some of the services could be running more often than needed, or some of the services may need to be run more frequently. For optimal PPM Center performance, for PPM Servers devoted to services, use `server.conf` to schedule the PPM Center services to run when they cause the least performance impact to the system.

Managing PPM Server Memory

A PPM Server runs in the context of a single JVM.

HP recommends using the server cluster configuration, with multiple JVMs, to provide for greater reliability and performance.

By using multiple PPM Servers in your configuration, you can:

- Support increased user load
- Improve capacity
- Maintain and provide acceptable performance

You can use multiple JVMs in addition to or as an alternative to adding extra machines to manage load balancing.

Using server clusters you can distribute client connections, based on Web traffic and server load, evenly among any number of PPM Servers in the cluster.

As always, you should test your solution to determine what is optimal in your environment.

Other important aspects of managing memory require reclaiming memory using garbage collection and properly sizing your memory heaps.

Garbage Collection

Garbage collection (GC) attempts to reclaim memory used by objects that will never be accessed or changed again by the application.

The garbage collector controls when objects are deleted from memory. Garbage collectors can add overhead that can affect performance.

Garbage collection is the process that releases memory from the Java heap and happens by the following methods.

Table 2-1. Garbage collection methods

Method	Characteristics
Scavenge	• Quick • Release application objects that have a short lifetime; for example, objects that are created and de-referenced within three minutes
Full	• Longer • Release all de-referenced application objects • Can impact application performance, because all application activity is stopped while they run

To modify and tune your garbage collection:

- Distribute users across PPM Servers
- Modify heap size by adding more memory

Generating Garbage Collection Performance Metrics

GC performance metrics can be generated by the JVM in which PPM Center runs. A common mechanism for generating GC information is to pass arguments into the JVM when an application is started. When monitoring GC data with specific tools, it is important to verify that the GC metrics generated match the needs of your organization.

Review your `gc.log` files to determine if and how garbage collection may be affecting your PPM Center implementation.

To enable verbose GC logging when starting PPM Center:

1. Open `kStart.sh` using your favorite text editor.
2. Locate the following `SYSTEM_PROPS` lines in `kStart.sh`:

   ```
   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 restart the PPM Server.

A file with garbage collection metrics should now exist in the *<PPM_Home>* directory. In this document, *<PPM_Home>* represents the path where PPM Center is installed.

Heap Sizing

Heap sizing, garbage collection, memory usage, and memory management are interrelated factors in the performance of PPM Center.

The goal of tuning heap usage is to reduce the number of full garbage collections that occur in any given time period. In most cases, the gains associated with eliminating full garbage collection cannot justify the effort needed to eliminate them.

To control the rate of full garbage collection, maintain a static amount of memory capacity on the heap for a given workload. Ensure that the amount of memory you allocate to the heap is sufficient to avoid a series of closely-spaced full garbage collections. This requires regular monitoring of memory usage on the JVM. As necessary, you should distribute users across additional PPM Servers or increase the amount allocated to heap.

The amount of memory allocated to the heap by default for a PPM Server is defined in application configuration files.

PPM Center provides components that you can use to configure your Java process memory settings. HP recommends starting with the following settings.

Component	Setting
Heap	1 GB or more
Perm space	256 MB

These settings can be increased or decreased based on your needs. Because memory settings are dependent on the application server platform and available physical memory on the machine, you will need to test the settings in your system to see what provides the best performance for you.

For example, if you have a single PPM Center application on a Microsoft Windows® server that has at least 2 GB of free RAM available for PPM Center, then use the following settings in your `server.conf` file:

```
SERVER_MAX_HEAP_SIZE=1280m
SERVER_INIT_HEAP_SIZE=1280m
SERVER_MAX_PERM_SIZE=256m
```

Deploying a Database Tier

The *System Administration Guide and Reference* contains the detailed information on how to tune and configure PPM Center, including details on how to configure your Oracle database for use with PPM Center.

Improving Portlet Performance

Your PPM Center performance can be significantly improved by tuning your portlets.

The load time of portlets on the PPM Dashboard page impacts everyone.

HP has the following recommendations for your portlets:

- Add mandatory filters for portlets to avoid queries that have poor performance.
- Limit the number of portlets on any given page.
  - Three to five is recommended.
  - Login times can be improved if there are fewer portlets.
  - Pages should load in ten seconds or less.

    Unless it is a very infrequently used page and users understand the expected load time.
- Ensure that any custom portlets you develop can return information in eight seconds or less (with the use of portlet filters) and the result set is less than 100 rows.
- Set the PPM Dashboard auto-refresh frequency to 60 minutes. 30 minutes should be the most frequently that the PPM Dashboard is auto-refreshed.
- Test the SQL and response time with large sets of data.

- Performance test all the important and complex portlets.

  Include testing the execution plans for any custom portlets with different filter field combinations.

- Avoid portlets that use several calls to SQL in a row.

  Every call to SQL requires a new database connection. Consider combining calls into a database function or procedure to limit the number of necessary connections. Connections established by the application and stored in the connection pool are not used.

- Leverage materialized views for complex SQL queries.

  Materialized views can cause overhead on the database if used improperly. Other areas to watch are:

  - How long it takes to refresh the portlet?
  - What is the refresh interval?
  - Does the requirement really call for a materialized view?

  Consult your DBA on the trade-offs and requirements of materialized views.

- `PORTLET_MAX_ROWS_RETURNED` is a system-wide parameter that sets the default maximum number of rows. The default value is 200. If it needs to be increased, understand the performance impact and increase it incrementally. HP recommends using increments of 50.

If your portlet does not meet the suggested recommendation, develop a custom report.

Improving Workflow Performance

If your implementation will be using the workflow functionality provided with PPM Center, then you should avoid the following while configuring your system and defining entities within PPM Center.

Avoid	Description
Tight loops	Tight loops of execution steps could be executed faster than the backend processing. For example, if the loop depends on a backend update, the backend update may not have been committed before the loop is executed again.
Transitions back to the workflow step itself resulting in infinite loops	For example, the “Failure” results are often used as a transition to loop back to the execution itself. Such a loop can cause indefinite processing, thereby slowing down the entire system.
Unnecessary execution steps	Combine execution steps or use database functions and procedures to accomplish extra steps. Unless step-by-step commits are needed, it is best to combine execution steps into one database function or procedure. Too many execution steps can be confusing and affect workflow performance.
Creating Timeout transitions	HP recommends that timeouts be used sparingly. Timeouts should be targeted for hours or days and should never be shorter than 15 minutes.

If custom functions or procedures are needed, use autonomous transactions, or try using built-in special commands in an execution step.

Pre-Production Performance Testing

HP recommends performing thorough testing of your system before providing it to your user community. Optimally, the test system should be a copy of your production system. This test system is essential to being able to properly troubleshoot functional and performance related issues.

Test cases should exercise the:

- Tasks performed most often
- Portlets and filters used most frequently

Load testing of the application is the most important pre-production testing that you can do. If you have the resources and bandwidth, load testing is the best way to predict the performance of your system. Your load testing should simulate:

- Multiple users accessing the program's services concurrently
- Processing large volumes of data input at one time, such as might occur with time-sheet entry on the last day of a pay period
- Running of multiple application functions, depending on the areas you will be implementing
- Retrieval of large sets of data

  Because data sets vary across most production deployments in terms of both entity volumes and characteristics, the performance cannot always be predicted.

- The hardware and configuration you want to use for production

Testing under low and even single-user load should also provide indications of the application performance for key operations. Single-user pre-production performance testing can have minimal affect on time lines and may be incorporated into the user acceptance test cycle or other pre-production tasks.

Critical use cases are also an important part of pre-production testing. HP recommends that you test those actions that are executed frequently and must perform acceptably for users of the system to successfully complete their work. For example, if implementing HP Time Management, it is critical that your users be able to log their time sheets.

During your testing phase, notes should be captured on areas of poor performance no matter how trivial they may seem. It is always worthwhile to investigate performance concerns. Depending on the concern, you may choose to test the issue again in a staged version of your production environment or in a separate single-user load system.

3 Monitoring Ongoing PPM Center Performance

Post-Deployment Performance Monitoring

HP recommends using your system utilities to monitor the following for PPM Center:

- CPU
- Memory, especially memory usage in the PPM Servers
- I/O (network and disk)

Monitoring physical system resource metrics over time can provide an administrator with a view of available system capacity.

When estimating the capacity of the PPM Server JVM to support the necessary workload, you should monitor metrics for memory usage.

As PPM Center workload characteristics change over time, it is important to monitor memory usage on the PPM Servers to identify when basic memory demand increases. When full garbage collection activity approaches a threshold that may compromise application performance, the deployment configuration should be modified to support the new demands.

When monitoring memory usage you should evaluate data that spans a week or more.

Periods of peak workload tend to have the greatest memory demands on the heap. Because every system is unique, it is important that you monitor and trend memory usage for your system.

PPM Center Factors to Monitor and Trend

HP strongly recommends that you manage performance regularly and proactively. Proactively monitoring performance can help you identify areas of poor performance before they become issues for your user community.

At a minimum you should monitor and trend the following, with the overall goal of understanding usage patterns and what your concurrent load is during peak hours.

Table 3-1. Items to monitor and methods to use

Items to monitor and trend	Suggested method to use
Monitor heap memory usage	JVM verbose garbage collection
Workload and response times	Web log analysis
Trend SQL characteristics (buffers, CPU)	Database reports
System usage	Operating system or enterprise monitoring software

For example, you may have 8,000 users registered in PPM Center, but only 600–800 are on the system concurrently at any given time. By monitoring the patterns of your user community, you can better tune the system to meet their needs.

Database Tier Monitoring

For your PPM Center database tier, make sure to monitor the following:

- Baseline CPU usage
- Logical I/O per second, that can indicate if you have runaway queries
- Background services, using the Redo Log per second as an indicator

The recommended database setup is detailed in the *System Administration Guide and Reference.*

Application Tier Monitoring

For your PPM Center application tier, make sure to monitor the following:

- System metrics, including:
  - CPU
  - Page rate and system virtual memory
  - Disk I/O rate
- Heap size

Monitoring and Analysis Tool Recommendations

There are many tools and methods available to you to monitor the performance of PPM Center.

HP recommends several tools that you can use for monitoring and analyzing the behavior of your systems. The following can help you with PPM Center:

- HP Diagnostics Probe

  Diagnostics Probe ships with PPM Center and helps monitor transient data.

- HP SiteScope

  Provides system resource profiling and allows you to configure alerts.

- HP Business Availability Center (BAC)

  Provides remote performance monitoring capability and allows for end-to-end transaction monitoring.

To monitor database performance there are many tools available. HP suggests full understanding and analysis of your company's particular needs when selecting a database performance monitoring tool.

Capacity Planning

When considering present and future workload and system capacity needs, consider the following:

- Existing and forecasted workload characteristics for at least a three-year time span.
- Distribution of transactions by type and over time.
- Different workload classes affect sizing.
- Session duration and characteristics differ significantly among classes.

Make sure to leave room in all areas of your PPM Center configuration for growth and usage beyond current expectation. If possible, create a system that should meet the needs of your company one to three years from now.

The memory capacity made available on the heap for a set of PPM Center application users can be controlled by increasing the:

- Amount of memory allocated to the heap
- Number of JVMs in the cluster supporting the user workload

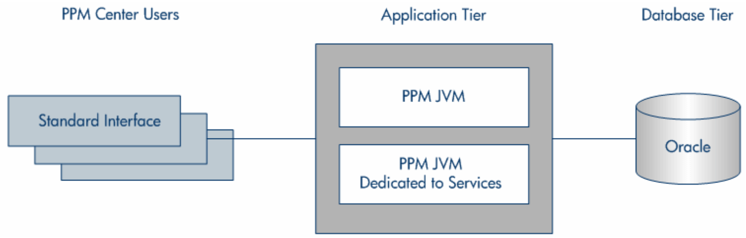
Because of per-process memory allocation limits on most operating systems there is a limit to which the size of the heap can be increased. The preferred way to grow PPM Center capacity on the application tier is to create additional PPM Servers within a cluster.

4 Solution Configurations

Solution Configuration 1

This configuration focuses on minimum investment at all levels. There is minimal application clustering, low hardware cost, and no load balancing.

Figure 4-1. Solution configuration 1



- Two physical servers
  - One for the application tier
  - One for the database tier
- No external Web server
- One JVM is dedicated to handling user traffic
- One JVM is dedicated to handling PPM Center services
- 6–8 GB RAM for the application tier is recommended
- 4 CPUs are recommended for the application tier

For small implementations with an estimate of under 500 users or for implementations where database and system administration support is limited, this may be the ideal configuration.

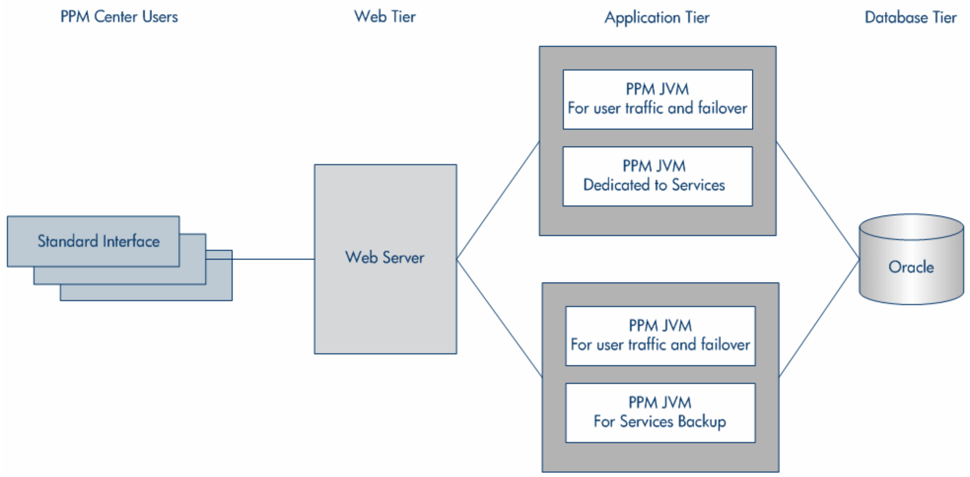
This solution is ideal where your project has the following goals:

- Minimum hardware cost
- Minimum configuration and administration
- User traffic is separated from services run by PPM Center
- Failover capability for user traffic is not a primary concern at the server level
- Single sign-on (SSO) support is not required

Solution Configuration 2

Builds on *Solution Configuration 1* by adding an external Web server and an additional JVM dedicated to user traffic. This configuration represents a moderate investment. There is minimal application clustering, low hardware cost, and no load balancing.

Figure 4-2. Solution configuration 2



- Four physical servers
  - One for the Web tier
  - Two for the application tier
  - One for the database tier
- Two JVMs are dedicated to handling user traffic and provide failover
- One JVM is dedicated to handling PPM Center services
- 8–12 GB RAM for the application tier is recommended
- 4–8 CPUs are recommended for the application tier

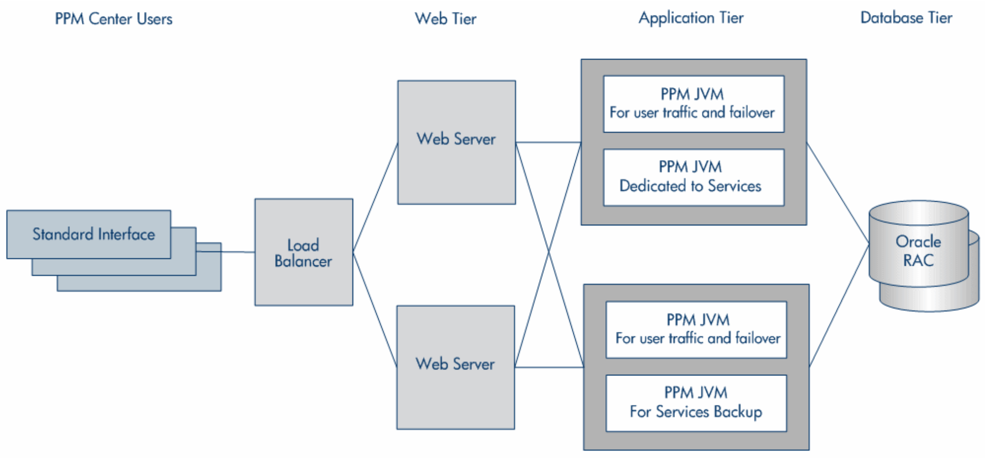
This solution is ideal where your project has the following goals:

- Single sign-on support
- Application-level failover and load balancing

Solution Configuration 3

This configuration represents a mission-critical server cluster configuration with clustered application, Web, and database tiers.

Figure 4-3. Solution configuration 3


PPM Center Users
Web Tier
Application Tier
Database Tier
Standard Interface
Load Balancer
Web Server
Web Server
PPM JVM For user traffic and failover
PPM JVM Dedicated to Services
PPM JVM For user traffic and failover
PPM JVM For Services Backup
Oracle RAC

- Six physical servers:
  - Two for the Web tier
  - Two for the application tier
  - Two for the database tier
- External Web servers sit on separate hosts
- User traffic is distributed across physical hosts and JVMs
- Two JVMs are dedicated to handling PPM Center services

- Application JVMs configured in a cluster to support service failover
- Oracle RAC implemented to provide database failover support
- 8–12 GB RAM for the application tier is recommended
- 4–8 CPUs are recommended for the application tier

This solution is ideal where your project has the following goals:

- Continuous application availability with significantly minimized system outages
- Single sign-on support
- Capacity to support future growth

Index