PeopleSoft Red Paper Series

PeopleSoft Performance Monitor

By: Sekhar Korupolu, Bor-Ruey Fu, John Houghton, Peter Schwarz, Terry Martin

April 2013

Including:

- Setting Up Performance Monitor in a Production Environment
- Understanding Performance Data
- Performance Monitor Overhead
- Troubleshooting Performance Issues-Use Cases
- Common Questions
- Data Volume Management
- Known Issues

PeopleSoft®

PeopleSoft Performance Monitor

Restricted Rights

The information contained in this document is proprietary and confidential to Oracle USA, Inc.

Comments on this document can be submitted to redpaper@peoplesoft.com. We encourage you provide feedback on this Red Paper and will ensure that it is updated based on feedback received. When you send information to Oracle USA, Inc. you grant Oracle USA, Inc. a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of Oracle USA, Inc.

This document is subject to change without notice, and Oracle USA, Inc. does not warrant that the material contained in this document is error-free. If you find any problems with this document, please report them to Oracle USA, Inc. in writing.

This material has not been submitted to any formal Oracle USA, Inc. test and is published AS IS. It has not been the subject of rigorous review. Oracle USA, Inc. assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by Oracle USA, Inc. for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Information in this book was developed in conjunction with use of the product specified, and is limited in application to those specific hardware and software products and levels.

Oracle USA, Inc. may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

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

Table of Contents

Chapter 1 - Introduction

PeopleSoft Performance Monitor (PPM) is available in Enterprise PeopleTools 8.44 and higher. This Red Paper describes best practices to setup, configure and maintain a Performance Monitor System. It also illustrates the use of PPM by providing some examples.

The intended audience for this Red Paper is Performance Monitor administrators and analysts.

Structure of this Red Paper

This Red Paper provides the guidelines for setting up a Performance Monitor System in a production environment. It also provides an estimate of the overhead introduced by enabling the Performance Monitor. In addition, there is an FAQ section that addresses some commonly asked questions about Performance Monitor. Finally, it discusses the options available to manage the large volume of data generated by the Performance Monitor.

Keep in mind that PeopleSoft updates this document as needed so that it reflects the most current feedback we receive from the field. Therefore, the structures, headings, content, and length of this document are likely to vary with each posted version. To see if the document has been updated since you last downloaded it, compare the date of your version to the date of the version posted on Customer Connection.

Related Materials

We assume that our readers are experienced IT professionals, with a thorough understanding of PeopleSoft's Pure Internet Architecture (PIA). To take full advantage of the information covered in this document, we recommend that you have a solid grasp of PeopleSoft System Administration, Internet architecture, relational database concepts/SQL, and PeopleSoft Application Architecture.

This Red Paper is not intended to replace the PPM and PIA documentation delivered in the PeopleTools PeopleBooks. Before you read it, you need to familiarize yourself with the Performance Monitor and PeopleSoft Pure Internet Architecture related information in the PeopleTools PeopleBooks to ensure that you have a well-rounded knowledge of our technology.

Additionally, the PeopleSoft Server Administration and Installation course offered by Oracle University covers the PIA Architecture in detail and provides an overview of PeopleSoft Performance Monitor. This training is highly recommended for anyone seeking to maximize their use of PPM as a performance monitoring tool.

Support of Self Monitoring Configurations

PeopleSoft Performance Monitor can be used to monitor several systems simultaneously. When the PPM server monitors its own activity, this configuration is called a "self-monitoring" system. For a variety of reasons, Oracle supports only configurations where the monitoring system is a separate system from the monitored system(s). In other words, PPM should have its own separate database server, web servers, application servers, and process scheduler.

Installation Considerations

As you get started with PeopleSoft Performance Monitor, you need to consider your goals in using the product. While PPM is a very powerful tool, it can become quite complex and generate large volumes of redundant data unless it is configured and maintained properly.

To help you think through this process, you need to first consider some questions that will help you design the monitoring system and estimate the required storage for it. For example:

1. Which PeopleSoft environments will be monitored?
2. How many web server, app server, process scheduler domains exist in each monitored environment?
3. How much detail do you want to collect for a typical user's activity?
4. How much performance data history do you want to retain?

5. During normal operation, what Agent Filter Level do you think would be appropriate to use by default?
6. Do you want to use PPM to gather general system statistics, or simply be available for users to trace a particular transaction?
7. Are there any firewalls, load balancers, and proxy servers anywhere in any of the monitored environments?

You can also refer to the ***Enterprise PeopleTools 8.52 PeopleBooks >PeopleSoft Performance Monitor > Estimating Your Performance Database Size Overview*** for additional criteria to consider prior to setting up PPM.

Chapter 2 - Setting up Performance Monitor for a Production Environment

This chapter discusses the settings and setup options for configuring a PPM monitoring system for a production environment.

SETTING UP THE MONITORING SYSTEM

The first step to setting up PeopleSoft Performance Monitor is to build and configure a monitoring system. This is the system that will collect and report on the performance data collected from other systems. This section describes how to set up the monitoring system.

To install the Monitoring System, follow these instructions:

1. Create (or upgrade) a database for the monitoring system.

   An instance of a PeopleTools System database is adequate, but additional storage will need to be added to the device/data file that contains the PPM Tables. Refer to the PeopleTools Installation Guide Chapter 7A for details.

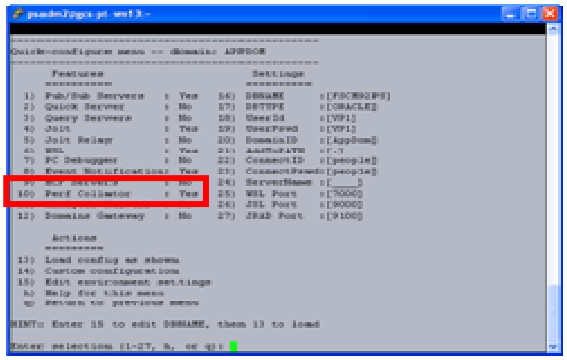
   The application DDL scripts are not needed since only PeopleTools tables are used for PPM. If you are building a database from scratch, one simple approach is to build the new database with the Database Configuration Wizard (DCW), and select Database Create Type = "PeopleTools System." By default, the PeopleTools System databases do not have any application tables.

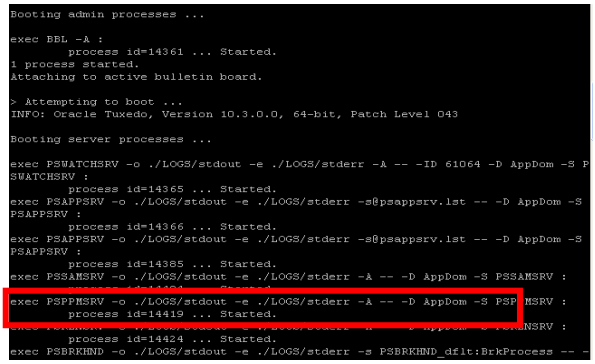
   Performance Monitor and the Environment Management Framework use a Global Unique Identifier (GUID) to uniquely identify a PeopleSoft system. If you choose to clone an existing PeopleSoft database as the monitoring system, make sure to set the GUID column in the PSOPTIONS table to a <space> before using it. The first time an application server accesses the database, it will populate the GUID. (Please refer to the note that cautions about directly manipulating PeopleTools metadata tables under the **Cloning Databases** topic on Page 30.)

2. Create an application server domain and enable Performance Monitor.

   a. Configure the application server domain to contain at least 2 PSAPPSRV processes.

   b. In the Quick Setup menu, turn on option "10. Perf Collator". Enabling the Performance Collator option tells Tuxedo to start a PSPPMSRV process in the Performance Monitor application server domain.





3. Deploy PIA to the web server.

   During that process, write down the following information for use in later steps:

   a. The web site name. The initial PIA setup automatically creates the default PeopleSoft site named *ps*. In subsequent PIA setups, change the site name from "ps" to a unique value. The site name is used in the URL used to access the system. In the following example *CRMPROD1* is the site name:

      `http://<machine_name>:<port_number>/CRMPROD1/signon.html.`

   b. The hostname

   c. The port number

   d. The web profile name

4. Create a process scheduler domain.

You need a process scheduler domain to run the PPM Archive, Reaper, and Look Up Application Engine programs.

5. Boot the web server, application server, and process scheduler domains.
6. Grant the User ID that will administer PPM access to the Performance Monitoring menu items.

   There are various ways to provide this access, e.g. PeopleSoft delivers the PTPT1200 permission list with this access. But, as is common with many delivered default configuration, the PPM Roles and Permission List are delivered as representative examples that need to be customized to meet each Customer's specific needs.

   It is not uncommon for Implementation Teams to adopt the delivered Security configuration. Security settings were will not apply universally to all Implementations, so we provide the extensive capabilities for Customers to configure Security according to their specific needs. PTPT1200 was originally produced to address the internal needs of PeopleTools Developers, and will not apply to most Customer Production Environments.

   Using the PeopleTools (PT) delivered Permission Lists in Production is also undesirable because PeopleTools may decide to change any PT Permission List and the Customer would potentially be impacted during the next upgrade. Also, there may be many Customers who currently use our delivered Security Configuration in such a way that WEBLIB_PPM access is properly isolated for them (very few people assigned the PeopleTools role), so they would be impacted if we made a change.

   Note: It is very important that the security role used to support PPM Monitoring menu items be tailored to meet specific customer requirements. The default role PTPT1200 is provided only as an example and may provide excessive privilege for typical users.

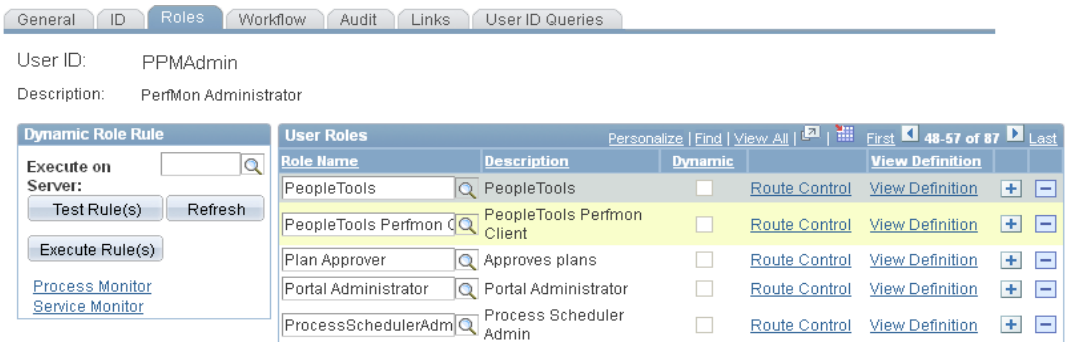
7. Refer to the following reference for more details configuring your system's security:

   `Enterprise PeopleTools 8.49 PeopleBook: Security Administration > Setting Up Permission Lists`

8. Setup PPMI user security.

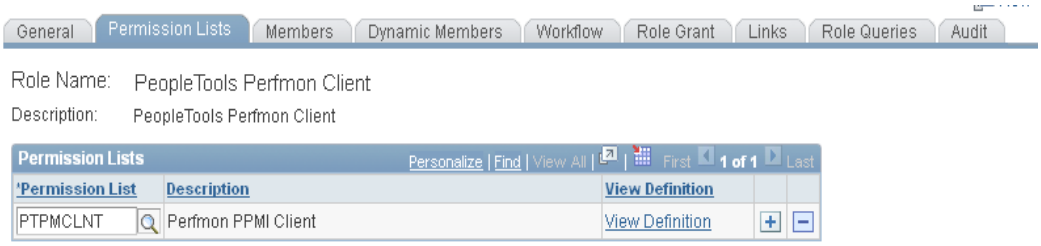
   The PPMI User ID is used by the monitoring system to register itself with the PPMI Monitor Servlet. Creating a new, dedicated user that is dedicated for this purpose is optional, but highly recommended. Only the security defined below is needed for this user.

   a. Navigate to **PeopleTools, Security, User Profiles, User Profiles**.
   b. Create a new user profile, e.g. PPMAdmin
   c. Assign the "PeopleTools Perfmon Client" role to the new user profile created above, to allow access to the PPMI servlet.



   d. Click the Save button to finish creating the new user profile.
   e. Confirm that this role contains the necessary permissions.

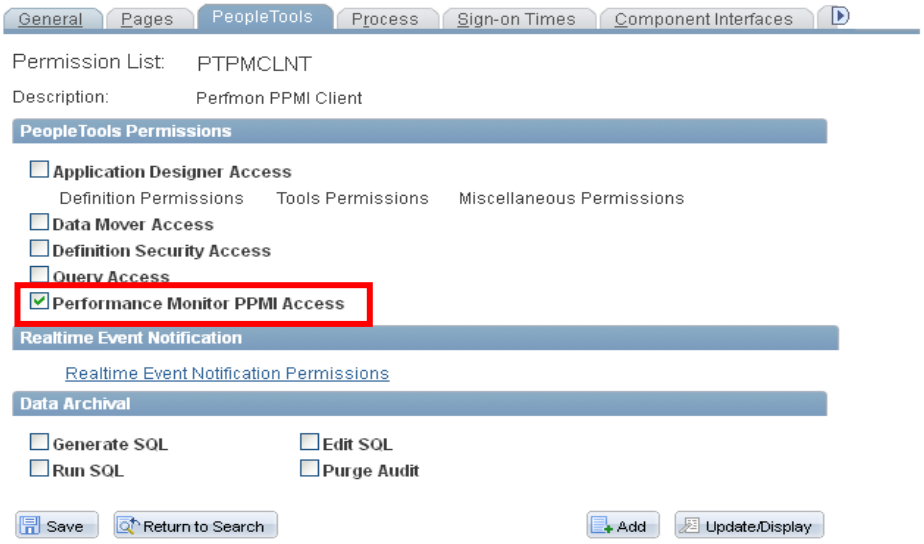
      Click on the View Definition hyperlink to open the Role Definition for the new user, e.g. PPMAdmin.



f. Click on the View Definition hyperlink to edit the Permission List for the "Perfmon PPMI Client" role.

g. Select the PeopleTools tab on the PTPMCLNT Permission List.

h. Make sure the PTPMCLNT Permission List has the Performance Monitor PPMI Access checkbox selected, e.g.:



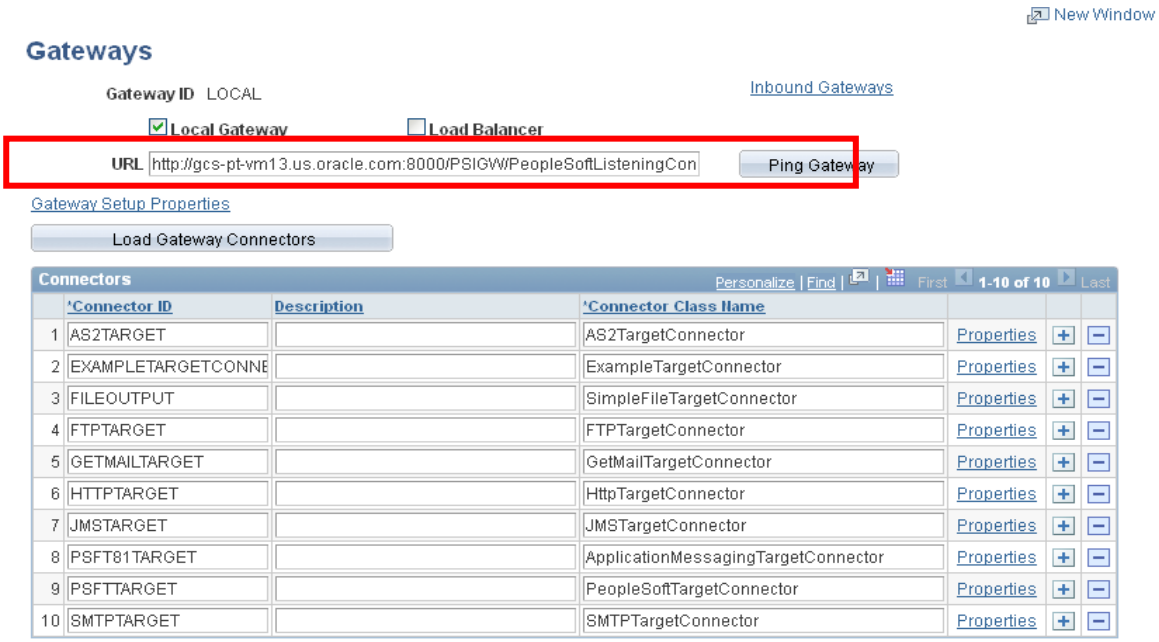
i. Save the permission list.

9. Enter the Integration Broker Gateway information. This is required for allowing dynamic changes to monitor Agent Filter Levels and parameters and immediate notification.

a. Navigate to **PeopleTools, Integration Broker, Configuration, Gateways** and enter the gateway URL, e.g.:

`http://<host>[:port]/PSIGW/PeopleSoftListeningConnector`

b. Enter the Gateway URL.



c. Click the Save button again.

This action will save the Gateway URL to the database.

d. Press Load Gateway Connectors

e. Click the Save button again.

Each time you change the URL, you need to perform these last three actions in that specific sequence.

10. Select the Agent Filter Level.

    On the monitoring system, confirm that the Agent Filter Level default is **04-Standard** or higher. If the Agent Filter Level is less than **04-Standard** then the System Performance page may display the message "stale agent data detected" for the monitored system.

    a. Navigate to **PeopleTools, Performance Monitor, Administration , Global Administration, System Defaults**

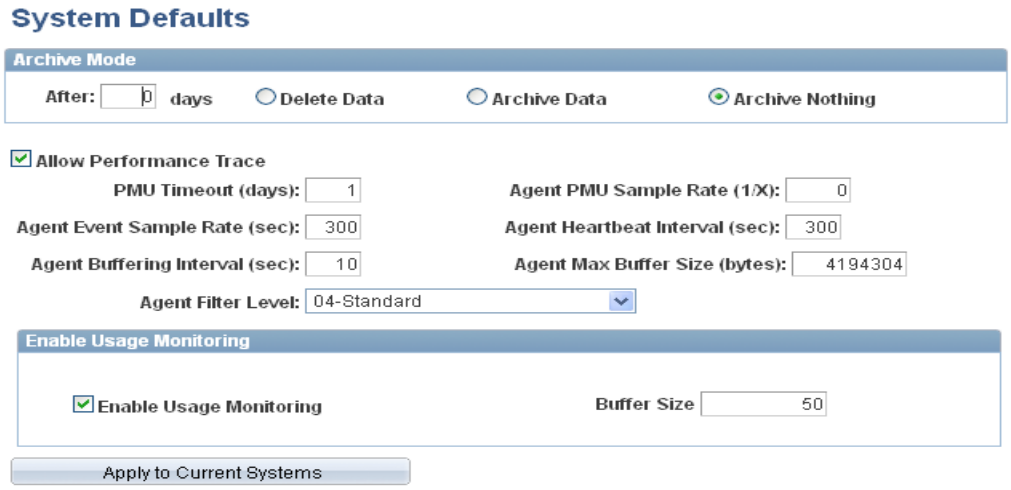
    b. Confirm that the Agent Filter Level default is set to a value of **04-Standard.**

    If it is not, update it accordingly.

    c. If you updated the Agent Filter Level default, click the Save button to ensure that the Agent Filter Level is saved to the database.

    Perform this step every time you change the Agent Filter Level and before you perform the next step. Forgetting to click the Save button at this point is a very common mistake made when configuring PPM - the result is that the new Agent Filter Level value is not updated.

    d. Click the "Apply to Current Systems" button to ensure that the Agent Filter Level default is applied to all monitored systems.



    e. To ensure that there was not a sequence error between steps c and d above, reopen the System Default page as explained in step a.

    f. Confirm that the Agent Filter Level is set to the value of **04-Standard**.

    If is not at this value, return to step c. above and repeat, making certain that you click the Save button before applying the settings to the current systems.

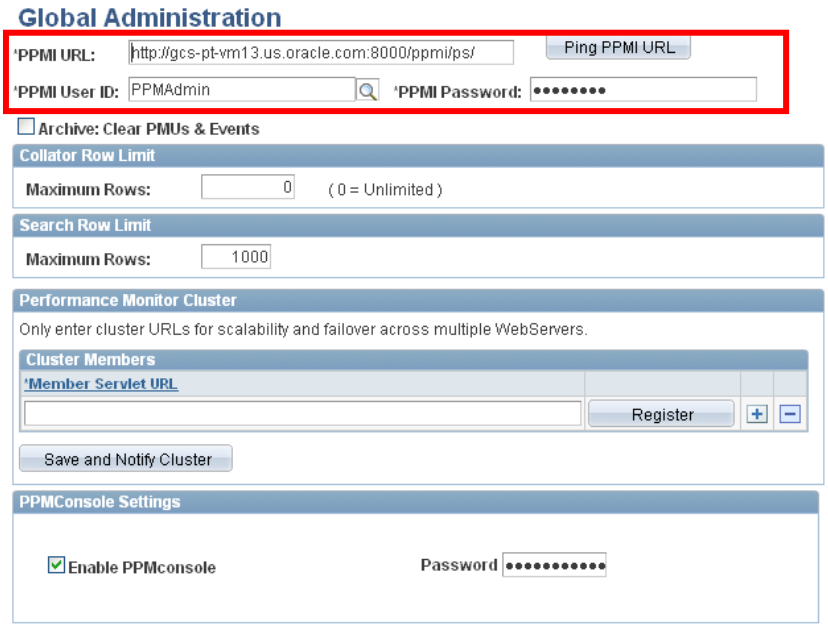
11. Enter the Global Administration URL and User ID information.

    a. Navigate to **PeopleTools, Performance Monitor, Administration, Global Administration**

    b. Enter the information specified earlier in the PPMI URL, e.g.

    `http://<host>[:port]/ppmi/<web site>/`

    Note: The syntax of the value is very important - the URL *must end* with a forward slash "/"):



Note: SSL is not supported in the communications between the PPMI Monitor Servlet and the PSPPMSRV Tuxedo server process. Do not attempt to use https as the protocol designator in the PPMI URL.

c. Enter the PPMI User ID and password created in the previous step.

d. Click the Save button.

e. Click the Ping PPMI URL button to confirm that you can locate the PPMI servlet.

12. Schedule the Reaper Application Engine program.

The Reaper program does the following:

- Deletes all rows in the current PMU table (PSPMTRANSCURR) that are flagged for deletion.
- Sets the status for expired PMUs to "timed out" in the current tables.
- Moves “timed out” PMUs to the history PMU table (PSPMTRANSHIST).

PeopleTools delivers a recurrence definition named **PerfMon Reaper Recurrence**, which is set to run every 15 minutes. This value is adequate for most installations.

To schedule the Reaper program for execution:

- Navigate to **PeopleTools, Performance Monitor, Administration, Schedule Reaper**
- Create a new run control, e.g. PPMReaper, or reuse one, as appropriate.

We recommend that this run control name be different from the other Archive processes.

Use **PerfMon Reaper Recurrence**, which by default runs every 15 minutes.

13. Schedule the Lookup Application Engine program.

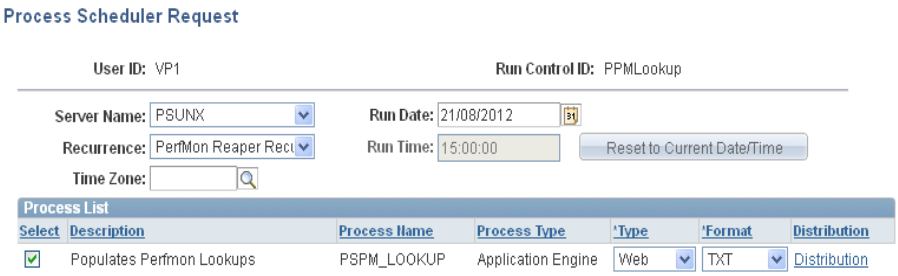
The PSPM_LOOKUP lookup program builds Performance Monitor's internal lookup tables for the User ID table (PSPMOPRDEFN), the components table (PSPMPNLGRPDEFN), and the performance trace names table (PSPMPERFTRACE).

a. Navigate to PeopleTools, Performance Monitor, Administration, Schedule Lookup.

b. Create a new run control, e.g. PPMLookup, or reuse one as appropriate. We recommend that this run control be different from the previous Archive processes.

c. Use **PerfMon Reaper Recurrence** which by default runs every 15 minutes.

If this recurrence is not available on your system, create a new one.



14. Schedule the Archive Application Engine program.

The archive program cleans up the current and history tables according to the archive options for each of the monitored systems. By checking the Run %UpdateStats at the end check box, the archive program will run %UpdateStats meta-SQL on both the history and archive tables after the archive program completes successfully.

a. Navigate to **PeopleTools, Performance Monitor, Administration, Schedule Archive**.

b. Create a new run control, e.g. PPMArchive, or reuse one as appropriate. We recommend that this run control be different from the previous Archive processes.

c. Check the Run %UpdateStats at the end check box and submit the run control.

d. Use **Daily Purge** recurrence, which runs daily at 1:00 am by default.

If this recurrence is not available on your system, create a new one.

15. Allow access to the PPMI Monitor Servlet diagnostic URL.

This URL is very helpful when diagnosing setup and configuration issues with PPM. For security reasons, you may choose to disable this access once the system is set up and functioning correctly. With PeopleTools 8.45 and higher, access to this interface is ***disabled*** by default.

a. Identify the current web profile being used. To identify which web profile is used for a web server domain, open the ***configuration.properties*** file under:

```
$PS_HOME/webserv/<web domain name>/applications/peoplesoft/PORTAL/WEB-INF/psftdocs/<web site>.
```

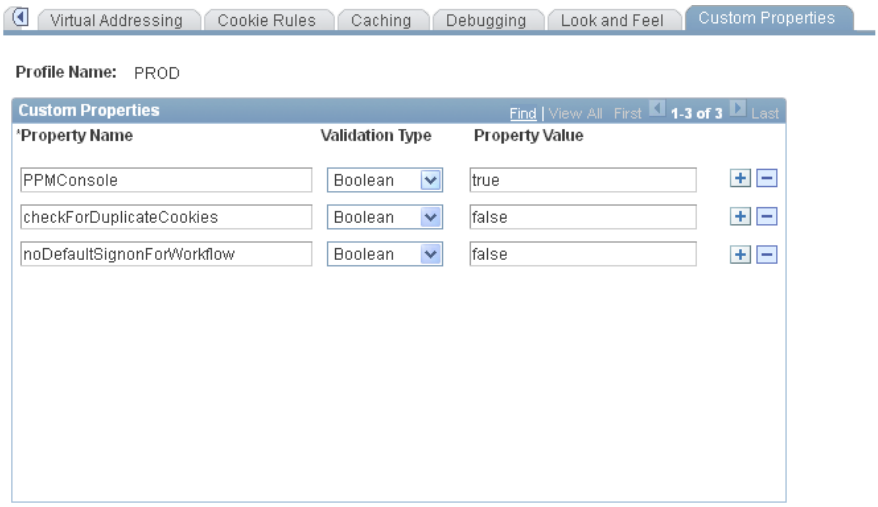
Look for the string "WebProfile" in this file to determine the current web profile.

b. Navigate to PeopleTools, Web Profile, Web Profile Configuration.

c. Open the web profile identified in step a.

d. Select the Custom Properties tab.

e. Add a Property Name "PPMConsole", type Boolean, Property Value of "true".



f. Save your changes.

g. Reboot the web server domains.

SETTING UP THE MONITORED SYSTEM

A PPM monitoring system can collect data from multiple monitored systems. This section focuses on the steps required to configure each of the monitored systems. Complete the following steps on each of the monitored systems.

1. Grant the User ID that will configure PPM access to the Performance Monitoring menu items.

   The easiest way to accomplish this is to link the User ID to the PeopleSoft delivered PTPT1200 permission list that authorizes the required access. Please refer to the following reference for more details on configuring your system's security:

   Enterprise PeopleTools 8.49 PeopleBook: Security Administration > Setting Up Permission Lists

2. Set up User Level Performance Trace privileges.

   Grant permission to access the WEBLIB_PPM WebLib to User IDs that will be initiating performance traces on demand . When the user logs on, the Performance Trace URL will be enabled in the default page header, near the log out URL.

   An authorized user can override the system default Agent Filter Level when requesting a Performance Trace. If the user session is timed out before stopping the trace, the performance trace is terminated and the Agent Filter Level reverts back to the System setting.

3. Confirm that the PPM agents are enabled.

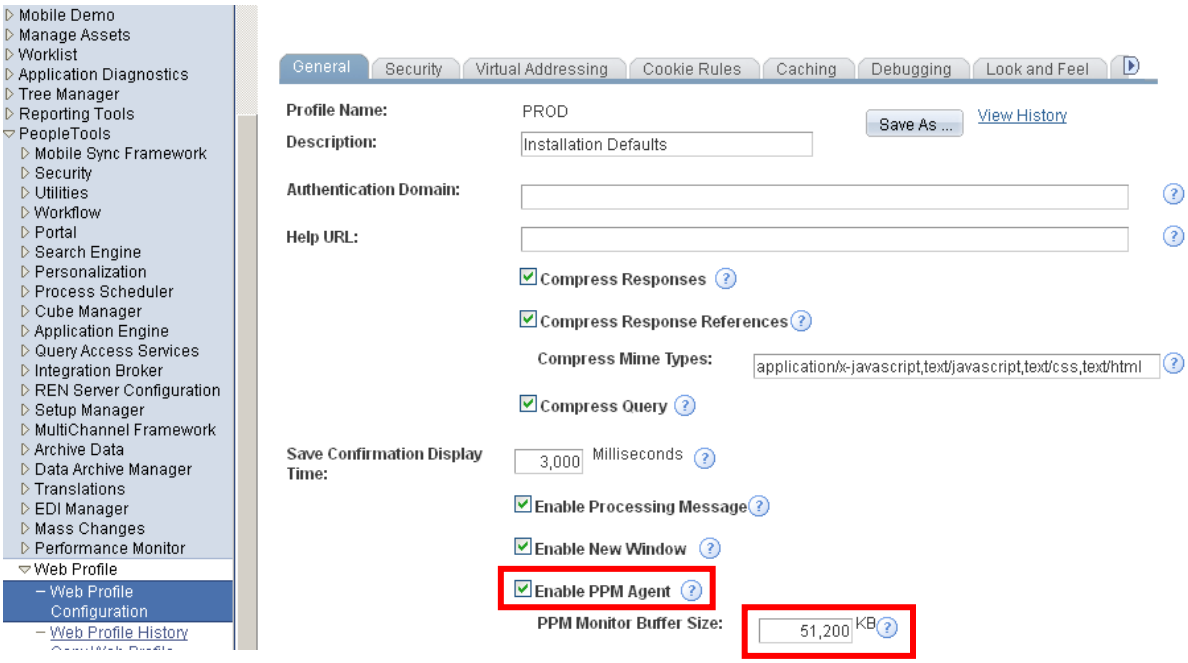
   In the monitored system, review the [PSTOOLS] section in the application server and process scheduler domain configuration files (i.e. psappsrv.cfg and psprcs.cfg).

Ensure that they are set to “EnablePPM Agent=1”.

```
[PSTOOLS]
General settings for PSTOOLS
=============================================================================
This will suppress SQL error dialogs. This option helps prevent
SQL injection attacks by only writing SQL errors to the log files
(and not displaying them to a user.)
Suppress SQL Error=1
Uncomment this to specify an alternate directory to search for Interface Drivers.
Business Interlink Driver Directory=
Set the value to 1 to Enable PPM Agents,0 to disable PPM Agents
EnablePPM Agent=1
```

If a change is made to the setting, restart the domain.

4. Confirm that the web profile is configured to enable PPM.
    a. Open the currently used web profile.
    b. Confirm that the Enable PPM Agent box is checked in the web profile
    c. Confirm the PPM Monitor Buffer Size is 51,200 KB (or at least greater than zero).



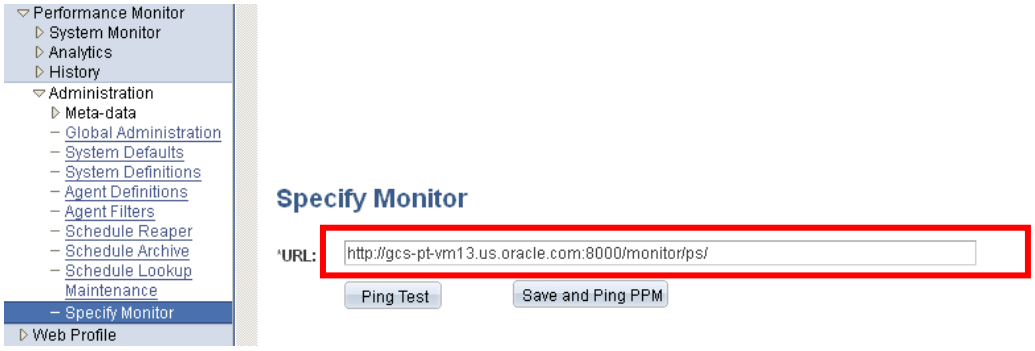
    d. If any changes are made to this page, restart the web server to ensure the web profile is reloaded, so the system will recognize the changed parameter settings.

5. Specify the Monitor URL.

    To enable PPM on a monitored system, populate the Specify Monitor URL with the monitoring system PPMI Monitor Servlet.

    a. Log into the monitored system.
    b. Navigate to **PeopleTools, Performance Monitor, Administration, Specify Monitor.**
    c. Enter the Specify Monitor URL information, making certain that it ends with a forward slash (“/”).

    ```
    http[s]://<host>[:port]/monitor/ps/
    ```



The communication between the Monitoring Agents and the PPMI Monitor Servlet can be secured using Basic Authentication and Digital Certificates. This configuration is described in the PeopleTools Security Red Paper available on My Oracle Support. Once the appropriate security is configured, use "https" as the protocol designator.

d. Click the Save button.

e. Click the Ping Test button to make sure your system can locate the PPMI Monitor Servlet.

This will confirm the communications path between the monitored system's PSAPPSRV process(es) and the PPMI Monitor Servlet on the monitoring system.

6. Reboot the web server, if necessary.

If you are enabling Performance Monitor for the first time, you do not need to reboot either the web server or application server domains of the monitored system(s). If the Monitor URL has ever been defined to a value other than **NONE** , then the next time you change it to another URL , i.e. pointing to a different monitoring system, or back to **NONE**, you must reboot the web server and application server domains of the monitored system.

7. Log into the monitored systems to trigger web server agent registration.

8. Wait for one agent heartbeat interval to allow full registration.

CONFIRMING PERFORMANCE MONITOR IS OPERATIONAL

This section lists the steps to follow to confirm that the PeopleSoft Performance Monitor systems are functioning correctly. Before proceeding with this section, ensure that all of the steps in the previous two sections have been carefully followed and no errors were produced during the setup process.

To verify that the common issues are resolved, follow the steps outlined below. If you are unable to resolve any of these issues, use MetaLink to generate a Service Request with Global Customer Support.

1. On the monitoring system, confirm that the Performance Collator is started. The PSPPMSRV process should be running in the monitoring system's application server domain. If it is not running, the PPMI Monitor Servlet running on the web server will not activate. To check the status of the PSPPMSRV process, do the following:

   a. Run the *psadmin* command for the monitoring application server domain.

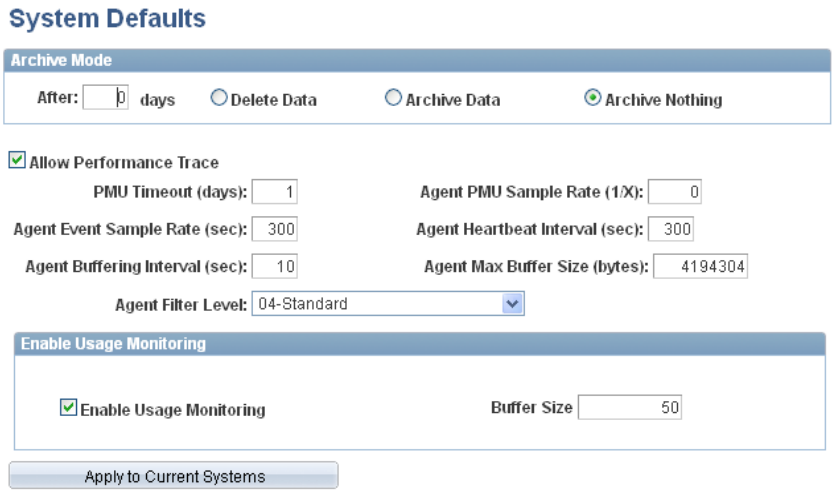
   b. Select *Application Server*, *Administer Domain* and choose the monitoring domain.

   c. Select *Domain status menu*, *Server status.*

   d. Verify that the PSPPMSRV process is running.

2. On the monitoring system, confirm that the Agent Filter Level default is set to the value **04-Standard** or higher. If the Agent Filter Level is less than **04-Standard** then the System Performance page may display the message "stale agent data detected" for the monitored system.

   a. Navigate to PeopleTools > Performance Monitor > Administration > Global Administration > System Defaults



b. Confirm that the Agent Filter Level is set to the value **04-Standard**.

c. Click the Save button to ensure that the Agent Filter Level is saved to the database.

   Perform this step every time you change the Agent Filter Level, *before* you perform the next step. Forgetting to click the Save button at this point is a very common mistake made when configuring PPM.

d. Click the Apply to Current Systems button to ensure that the Agent Filter Level is applied to all monitored systems.

e. To ensure that there was not a sequence error between steps c. and d. above, reopen the System Default page as described in step a.

f. Confirm that the Agent Filter Level is set to the value **04-Standard**.

   If not, repeat step c. above, making certain that you click the Save button before applying the settings to the current systems.

3. Review the monitoring system's application server logs to confirm that the collator PSPPMSRV process registered with the PPMI Monitor Servlet.

   In order for PPM to initialize properly, the PSPPMSRV process must contact the PPMI Monitor Servlet and complete the registration process. After this registration occurs, the PSPPMSRV process will send the desired Agent Filter Level to the PPMI Monitor Servlet, which will then broadcast that setting to each monitored system as it registers.

4. Open the current application server log file of the monitoring system which is typically located in:

   `$PS_HOME/appserv/<domain>/LOGS/APPSRV_mmdd.LOG:`

   When the PSPPMSRV registers with the PPMI Monitor Servlet, the following highlighted message is written to the application server log file:

```
PSPPMSRV.8709 (0) [10/29/12 06:08:56](0) PeopleTools Release 8.52.06 (Linux) starting. Tuxedo server is PPMGRP(91)/100
PSPPMSRV.8709 (0) [10/29/12 06:08:57](3) Worker Thread Starting
PSPPMSRV.8709 (0) [10/29/12 06:09:09](0) Collator registered with Monitoring Server
PSPPMSRV.8709 (0) [10/29/12 06:08:57](3) New PPMI URL. Starting Registration sequence
PSPPMSRV.8709 (0) [10/29/12 06:09:09](0) Collator registered with Monitoring Server
```

5. Review the monitored system's application server logs to confirm that the agents registered with the PPMI Monitor Servlet.

   When a PPM-enabled process is started, one of the first things it does it look to see if the PPM Monitor Server URL is populated with a value other than NONE. If the field is populated, then the agent will spawn a JVM and attempt to register with the PPMI Monitor Servlet.

6. Open each current application server log file of the monitored system, which is typically located in $PS_HOME/appserv/<domain>/LOGS/APPSRV_mmdd.LOG.

   When the agents register with the PPMI Monitor Servlet, the following highlighted message is written to the application server log file:


```
PSAPPSRV.8656 [10/29/12 06:09:55](2) (PerfMon Agent) Registered successfully ID:12
PSAPPSRV.8675 [10/29/12 06:09:56](2) (PerfMon Agent) Registered successfully ID:13
PSSAMSRV.8694 [10/29/12 06:09:56](2) (PerfMon Agent) Registered successfully ID:14
PSMONITORSRV.8768 [10/29/12 06:10:02](2) (PerfMon Agent) Registered successfully ID:15
```

7. Review the monitored system's web server logs to confirm that the agents registered with the PPMI Monitor Servlet. When a PPM-enabled web server is started, one of the first things it does is look to see if the PPMI Monitor Server URL is populated with a value other than **NONE**. If the field is populated, then the agent will attempt to register with the PPMI Monitor Servlet.
8. Open each current web server log file of the monitored system, which is typically located in:
   - WebLogic: $PS_HOME/webserv/<web server domain name>/logs/ PIA_weblogic.log
   - WebSphere: $ PS_HOME /webserv/<profile_name>/logs/server/<SERVER_NAME>/ WebSphere SystemOut.log
   - Oracle application server: $OAS_HOME/opmn/logs

   When the web server registers with the PPMI Monitor Servlet, the following highlighted message is written to the web server log file:

```
####<Nov 29, 2012 8:24:10 AM EDT> <Info> <ServletContext-> <gcs-pt-vm13.us.oracle.com> <PIA>
<[ACTIVE] ExecuteThread: '0' for queue: 'weblogic.kernel.Default (self-tuning)'> <<anonymous>> <>
<cc28c228418a5190:-2a5b2af:1390b56157a:-8000-000000000000002a> <1344515050363> <BEA-000000>

<PerfMon Agent Registered gcs-pt-vm13:8000:4430 type:WEBRESOURCE system:25fc6bc2-5ce8-11e1-8803-
fc2cef823c43 Assigned ID: 16 Filter: 4>

####<Nov 29, 2012 8:24:10 AM EDT> <Info> <ServletContext-> <gcs-pt-vm13.us.oracle.com> <PIA>
<Thread-15> <<anonymous>> <> <cc28c228418a5190:-2a5b2af:1390b56157a:-8000-0000000000000006>
<1344515050364>

<BEA-000000> <Registered successfully ID:16>
```

9. Review the monitored system's process scheduler server logs to confirm that the agents registered with the PPMI Monitor Servlet.

   When a PPM enabled process scheduler process is started, one of the first things it does is look to see if the PPMI Monitor Server URL is populated with a value other than **NONE**. If the field is populated, then the agent will attempt to register with the PPMI Monitor Servlet.
10. Review the monitored system's process scheduler logs to confirm that the process scheduler agents registered with the monitoring servlet.

    Normally you will see that at least one process will register, PSMONITOR. If the system is configured with a master scheduler, a second process will register, called PSMSTPRC.
11. Open each current process scheduler log file of the monitored system(s), which is typically located in:

    $PS_HOME /appserv/ prcs/<scheduler name>/LOGS/APPSRV_*mmdd*.LOG:

    This is what PSMONITOR looks like when it registers:

```
PSMONITORSRV.3888 [10/30/12 23:12:59](2) (PerfMon Agent) Registered Successfully ID:1
```

    If the master scheduler is enabled, this is what the PSMSTPRC process looks like when it registers.

```
PSMSTPRC.3796 [10/30/12 23:13:47](2) (PerfMon Agent) Registered successfully ID:8
```

12. [PT8.45 and higher]] On the monitoring system, confirm that the clients have registered properly with the PPMI Monitor Servlet Diagnostic URL. In this step, a "client" or "ppmiclient" is another name for a Monitored System.
    a. Enter the PPMI Monitor Servlet diagnostic URL as follows via a web browser.

       http://<host>:<port>/monitor/<site>/?cmd=ppmiclients

    b. The message "Monitor Console is disabled in this profile" indicates that the PPMConsole custom property was not properly set in the current web profile as indicated in a previous section.

       If changes are made to the web profile to correct this problem, be sure to restart the monitored system's web server so the changes will go into effect.

Clients -- CurrentTime: Thu Nov 22 21:20:15 EST 2012

Group	Id	Url	Queue Length	Estimated Queue Size	Item Processed	Estimated Bytes Processed	Max Size	Running Avg Size	LastComm	Limit
25fc6bc2-5ce8-11e1-8803-fc2cef823c43	3ca00000004	http://gcs-pt-vm13.us.oracle.com:43780	0	0	120536	119473284	2159312	78399	Nov 22, 2012 9:19:40 PM	52428800

1 Sessions, 1 clients

c. Review the output of the Show Clients page:

d. Each monitored system is listed in the Group column.

e. Confirm the value of the Limit field.

This value is the maximum size of the buffer per performance collator/PPMI client kept at the PPMI Monitor Servlet. This value is defined in the web profile as the PPM Monitor Buffer Size.

f. If you make a change in the web profile to address an issue with this step, restart each of the monitored systems' web servers so the changes will go into effect.

13. [PT8.45 and higher] On the monitoring system, confirm that the agents have registered properly with the PPMI Monitor Servlet Diagnostic URL. In this step, there should be an agent for each process on the monitored system that has registered with the PPMI Monitor Servlet.

a. Enter PPMI Monitor Servlet diagnostic URL as follows in a web browser.

`http://<host>:<port>/monitor/<site>/?cmd=agents`

b. The message "Monitor Console is disabled in this profile" indicates that the PPMConsole custom property was not properly set in the current web profile as indicated in a previous section.

If changes are made to the web profile to correct this problem, restart the monitored systems' web server so the changes will go in effect.

c. Review the output of the Show Agents page.

The Id column should correlate with each Agent ID that registered, as reviewed in the previous steps:

Agents -- CurrentTime: Thu Nov 22 21:29:48 EST 2012

System: 25fc6bc2-5ce8-11e1-8803-fc2cef823c43

Id	LastComm	Filter	Buf-Size	Send-Itvl	HeartBeat	Sample-Itvl	UserTrace	SamplingRate	SamplingFilter
17	Nov 22, 2012 9:27:53 PM	4	4194304	10000	300000	300000	true	0	0
16	Nov 22, 2012 9:28:58 PM	4	4194304	10000	300000	300000	true	0	0
19	Nov 22, 2012 9:29:37 PM	4	4194304	10000	300000	300000	true	0	0
20	Nov 22, 2012 9:28:16 PM	4	4194304	10000	300000	300000	true	0	0
25	Nov 22, 2012 9:27:01 PM	4	4194304	10000	300000	300000	true	0	0
26	Nov 22, 2012 9:27:03 PM	4	4194304	10000	300000	300000	true	0	0
11	Nov 22, 2012 9:25:20 PM	4	4194304	10000	300000	300000	true	0	0
12	Nov 22, 2012 9:29:37 PM	4	4194304	10000	300000	300000	true	0	0
13	Nov 22, 2012 9:25:39 PM	4	4194304	10000	300000	300000	true	0	0
14	Nov 22, 2012 9:27:19 PM	4	4194304	10000	300000	300000	true	0	0
15	Nov 22, 2012 9:25:58 PM	4	4194304	10000	300000	300000	true	0	0

d. If no systems are displayed, ensure that the monitoring system is set to an Agent Filter Level of **04-Standard**.

If it is not set to **04-Standard**, carefully perform the step "Configure Agent Filter Level" in the previous section "Setting Up the Monitoring System" If you make a change at this point, restart the monitored system's web server, application server, and process scheduler. Return to the beginning of this section and repeat the steps to confirm that the various components are working as expected.

e. Confirm that the agents for each monitored system appear on the list.

There should be a domain agent for each web server, each PPM-enabled application server process, and each PPM-enabled process scheduler process.

f. Verify that the Keep Agent buffering interval (Send-Itvl) for each monitored system is set to a reasonable value.

The default is 10,000 milliseconds (10 seconds), which should be adequate for most installations.

g. Confirm that for each registered system that the Agent Heartbeat Interval (HeartBeat) and Agent Event Sample Rate (Sample-Itvl) should also be set to a reasonable value.

The default for both is 300,000 milliseconds (300 seconds), which should be adequate for most installations.

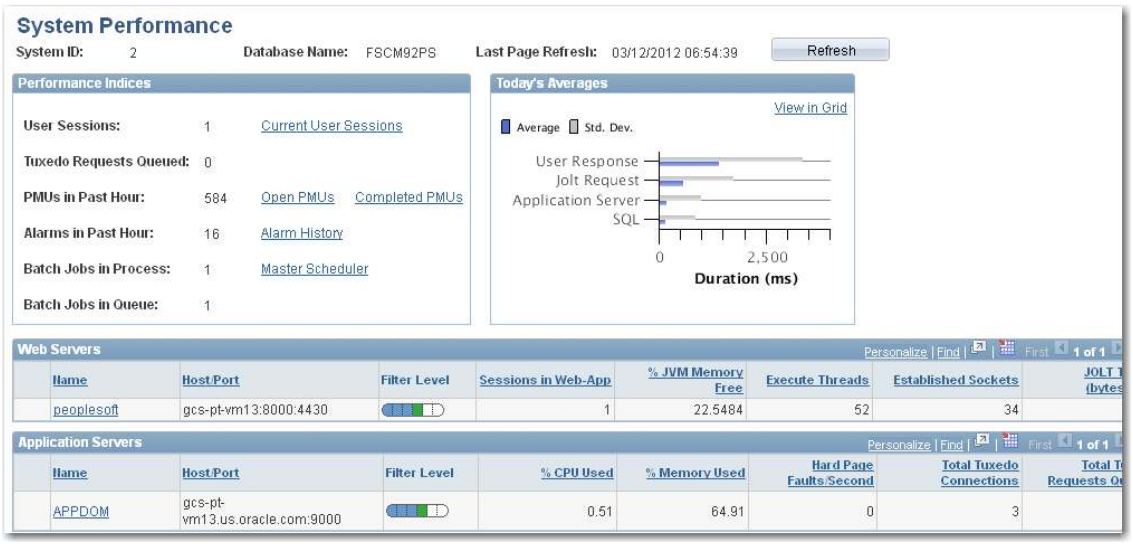
h. Verify that the Agent buffer size for each monitored system has a non-zero value.

If this value is 0, the agents will never be able to collect performance data. The default value is 4MB, which should be adequate for most installations.

14. On the monitoring system, ensure that the PMU data is being received.

a. Navigate to **PeopleTools, Performance Monitor, Administration, System Performance**

b. If there are multiple systems being monitored, select the desired system. The following page will be displayed:



Note: This is a Static PIA page, so the Refresh button must be clicked to ensure that the displayed data is current.

c. Next to each web server, application server and process scheduler the yellow triangle indicating “stale agent data” will disappear as data is received from the associated agents.

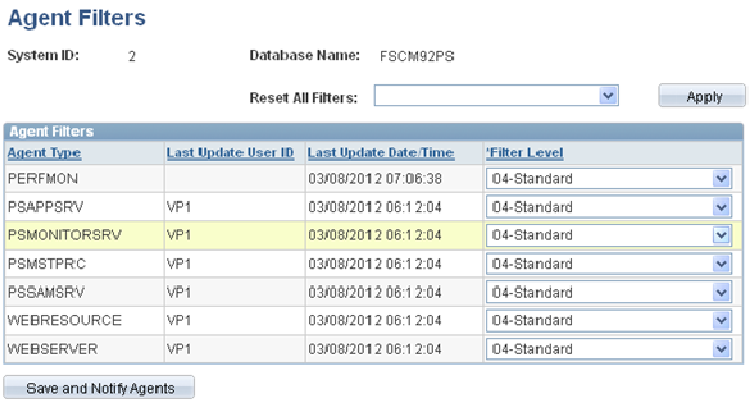
d. Once you have confirmed that the yellow triangle is not displayed next to a server, manipulate that monitored system to ensure that PMU data is being sent to the monitoring system.

e. Confirm that the values in the Performance Indices section indicate that the monitoring system is receiving PMU data from the monitored system.

15. Set the Agent Filter Level as desired. Once the PPM is configured and confirmed to be working properly, you can individually set the Agent Filter Levels on types of agents for a given monitored system.

Note that it is not typically necessary to override the System Agent Filtering Level.

a. Navigate to **PeopleTools, Performance Monitor, Administration, Agent Filters.**



b. Adjust the Agent Filter Level as desired.

c. Click the Save button.

d. Click the Save and Notify Agents button.

TROUBLESHOOTING PERFORMANCE MONITOR SETUP PROBLEMS

This section provides some additional details related to common configuration-related issues. If you are unable to resolve any of these issues, use MetaLink to enter a Service Request with Global Customer Support.

PerfMon Agent Error 500

Symptom: PerfMon Agent error 500 appears in the application server log of the monitored system.

```
PSAPPSRV.212 [12/02/12 08:55:00](2) (PerfMon Agent) Registration failed: Invalid response code from
server: 500 Site name is not valid. Check your url syntax and try again.
```

Recommendations: This message indicates that the agent is unable to connect to the PPMI Monitor Servlet. If you receive this error, try the following steps to resolve it.

- If this error occurs on a monitoring server for the PSPPMSRV process, check the monitoring system's PPMI URL on the Global Administration page. If you change the URL, you need to reboot the web server, application server, and process scheduler domains of the monitoring system. You may also need to restart the monitored systems to ensure they are correctly registered.
- If this error occurs on a monitored server PSAPPSRV process, confirm that the site name has been properly entered in the URL field on the Specify Monitor page. It is easy to confuse the "site" name with the web server's name. For example, in the following URL the site name is "ps".

  ```
  http://myhost.mydomain.com:8080/monitor/ps/
  ```
- If the web site was deployed with the name "psmon" this error will occur when agents attempt to connect to the incorrect URL. If you change the Monitor URL, reboot the web server, application server, and process scheduler domains of the monitored system.
- If this error occurs in the APPSRV_xxyy.log on a monitored server, confirm that the web site has been properly deployed on the monitoring server. It is possible that the web server is running, but the web site is not deployed.

Repeated PSPPMSRV Root Object Error

Symptom: A PSPPMSRV root object error appears in the application server log of the monitoring system. A sample of the error message looks like:

```
PSPPMSRV.3112 (0) [12/02/12 09:43:16](3) Root Object request failed. PPMI response <?xml version="1.0"
encoding="UTF-8" ?>
<Envelope xmlns="http://schemas.xmlsoaporg.org/soap/envelope/">
<Body>
<omiError>
<error number="10">
<message>Login Failure</message>
</error>
```



```
</omiError>
</Body>
</Envelope>
```

Recommendations: In the monitoring system, confirm that the following are correct:

1. The PPMI User ID and password are correctly defined on the PPM Global Administration page. Navigate to **PeopleTools, Performance Monitor, Administration, Global Administration** to verify the settings.
2. The PPMI User ID is attached to the "PeopleTools Perfmon Client" role. Navigate to **PeopleTools, Security, User Profiles, User Profiles**, and check the settings on the Roles tab to verify the attachment to the role.
3. The "PeopleTools Perfmon Client" role is associated with the "PTPMCLNT" permission list. Refer to the Permission Lists tab under **PeopleTools, Security, Permissions & Roles, Roles**.
4. The "PTPMCLNT" permission list grants access to the PPMI servlet. Refer to the PeopleTools tab under **PeopleTools, Security, Permissions & Roles, Permission Lists**. Make sure the Performance Monitor PPMI Access checkbox is selected. Save the changes you made. Performance Monitor will be activated the next time the PSPPMSRV process attempts to register with the monitor/PPMI servlets. There is no need to reboot the application server domain of the monitoring system.

Agent Registration Error

Symptom: PerfMon Agent error 404 appears in the application server log of the monitored system.

```
PSAPPSRV.3092 [12/02/12 09:09:30](2) (PerfMon Agent) Registration failed: Invalid response code from
server: 404 Monitor not activated.
```

Recommendation: The performance collator (PSPPMSRV) process has not been activated in the monitoring system, hence the PPMI Monitor Servlet has nowhere to send PMU data. Reconfigure the monitoring server's application server domain to ensure the Performance Collator option is set to yes, as this is the option that enables the PSPPMSRV process. Rebuild the domain and restart it.

Stale Agent Data Detected

Symptom: An  icon with the "Stale Agent Data Detected" message appears at the top of the System Performance page.

Recommendation: This symptom indicates that a displayed component has not reported data in within the sampling interval specified by the last Agent Event Sample Rate (Default value is 5 minutes.). It can be caused by many different issues.

If this symptom occurs, take the following steps as part of your troubleshooting strategy:

1. Confirm that the monitoring and monitored systems are set up correctly. The previous sections explain the detailed steps needed to properly configure PPM. Ensure that you have performed those setup steps correctly.
2. Examine the event data. It is possible that the event data is older than the specified Agent Event Sample Rate. This can happen when a domain is booting up and an agent has yet to report the first event data. If this is the case, after data is collected, refresh the page.
3. Check to see if one of the monitored domains is inactive, or has been shutdown. There is no need to change any of Performance Monitor's settings. Refer to the **Inactivate Unused Agents** topic in the **Tips On Administering The PeopleSoft Performance Monitor** section later in this chapter.
4. Review the monitor Agent Filter Level parameter setting on the monitoring system. It may be set so that no data is being sent to the monitoring server, e.g. **01-Standby** or **Off**. In PeopleTools 8.45 and higher, the System Default for the Agent Filter Level is **01-Standby** for any systems registered with the performance monitor. If this is the issue, set the Agent Filter Level to **04-Standard** and click the Save and Notify Agents button and wait for the next sampling interval to complete, which is determined by the Agent Event Sample Rate parameter value.
5. If a web server domain is the source of the "Stale Agent Data Detected" message, it could be the result of users not having logged into that domain yet. Note that a web domain monitor agent will only register itself and report event data to the performance monitor when the first user connects to the web server. To confirm that this is the source of the "Stale Agent Detected message, log into the monitored system through all available web server domains and then see if the message(s) goes away.
6. Determine if the Collator row limit has been reached for the PSPPMSRV tracking tables. See the following topic for further information on how to detect and resolve that issue.

Collator limit is reached

Symptom: PSPPMSRV agent reports the following message in the application server log of the monitoring system.

```
PSPPMSRV.1524 (0) [12/02/12 13:48:13](2) Row Limit reached. Not inserting data (209,43).
```

Additionally, you may see following message in the same log file due to side effects.

```
PSAPPSRV.1222 [12/02/12 13:53:00](2) (PerfMon Agent) Communication to Monitor Servlet failed.
Either the Performance Monitor URL is invalid or the Performance Monitor Webserver is unavailable.
Exception: java.net.ConnectException: Connection refused: connect
```

Recommendation: The PSPPMSRV process stops inserting data into the database immediately when the total row count of the three tables, PSPMTRANSHIST, PSPMTRANSCURR, PSPMEVENTHIST, exceeds the maximum value. This will also cause the "Stale Agent Data Detected" message to appear in the system performance page, even when everything else is setup correctly.

In the monitoring system,

1. Navigate to **PeopleTools, Performance Monitor, Administration, Global Administration**.
2. Set Maximum Rows in the Collator Row Limit box to 0 (unlimited) or a higher number than the current value.

Click the Save button after changing the Maximum Rows value. The PSPPMSRV process will pick up the new value automatically, you do not need to reboot the application server or web server of the monitoring system. .

The "Stale Agent Data Detected" message should disappear from the System Performance page after one interval of the Agent event sampling rate.

Note: When the collator row limit is reached, no more performance data will be inserted into the database by the Performance Collator. However, the monitor agents continue collecting and sending performance data to the PPMI Monitor Servlet. The PPMI Monitor Servlet continues publishing SOAP messages. Therefore, there will be no interruption to a 3rd party system integrated with Performance Monitor.

No Performance Data Is Collected

Symptom: No data is returned on the User Requests or Completed PMU pages.

Recommendation: This symptom can be caused by many different issues. Start your troubleshooting by confirming that the monitoring and monitored systems are set up correctly. The previous sections explain the many steps needed to properly configure PPM. Common explanations for this problem include**:**

1. Incorrect permission list or security settings.
2. Incorrect URL parameter values.
3. The performance collator option for the application server domain is not turned on.
4. The Agent Filter Level is set to 01-Standby.
5. The Collator Row Limit has been reached.
6. Incorrect values have been entered for the agent buffer and collator buffer parameters.
7. The Agent PMU Sample Rate (1/X) it set too high.
8. The Agent Event Sample Rate is set to 0.

Incorrect Column Values in the PSPMTABLEMAP table

Symptom: No performance data is recorded by the performance monitor and the following message appears in the application server log file.

```
PSPPMSRV.1255 [12/02/12 11:13:10](1) No row found in PSPMTABLEMAP (209,5)
```

Recommendation: The values in this table direct PPM where to store event and PMU data. The values should be "PSPMTRANSHIST" for the PM_TRANS_TBL_NAME field, and "PSPMEVENTHIST" for the PM_EVENT_TBL_NAME field. If these are incorrect, this error will be generated.

Note: Do not point the fields to any other database tables. If changes are made, none of the performance monitor charts and reports will be populated

These tables are populated in the **PerfmonPurgeAll.dms** data mover script.

Also, it is important to understand that when the Schedule Archive Application Engine program is running, these fields are temporarily changed to “PSPMTRANSHISTCL” and “PSPMEVENTHISTCL”, respectively. This avoids table locking when the Schedule Archive Application Engine program runs.

SCALING A MONITORING SYSTEM

The PeopleSoft Performance Monitor is designed to monitor one or more PeopleSoft systems. However, adding additional monitored systems increases the potential for a large volume of event and PMU data to be generated. Therefore, it is helpful to understand how to scale the monitoring system to handle the increased monitoring load.

Architectural Issues

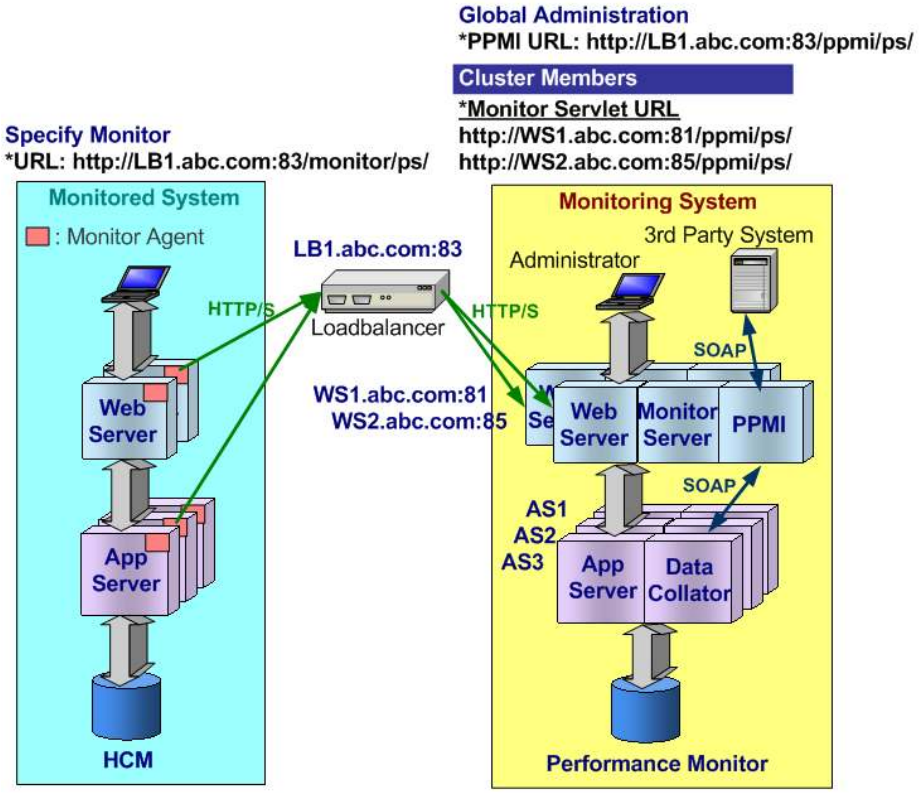
From an architectural standpoint, there are two critical communication paths that need to be optimized to handle increased volumes of event and PMU data:

1. Communication between the monitored systems’ Agents and the PPMI Monitor Servlets. Throughput can be increased by adding additional PPMI Monitor Servlets into the architecture.
2. Communication between the monitoring system’s PPMI Monitor Servlet and the PSPPMSRV processes. Since the PSPPMSRV is responsible for storing Event and PMU data in the database, additional parallel processes can improve throughput.

It is possible to setup a PPMI Monitor Servlet Cluster to distribute the load from the Monitoring Agents. For details on this configuration refer to:

PeopleBooks > Performance Monitor > Administering the Performance Monitor System > Setting up Monitor Clusters

The following diagram shows a monitoring server configuration that has been scaled with multiple web and application servers. It shows the various URLs that need to be used to ensure proper operation.



Note: If using load balancing, use the load balancer’s machine name in place of the host machine name in Specify Monitor URL and PPMI URL of the monitored systems. Use the web server machine name in place of the host machine name in Cluster Member URL.

Application Server Configuration

Like other Tuxedo processes, more instances of PSPPMSRV can be started, to handle the load from the PPMI monitor servlet. Change the following lines in the psappsrv.cfg, and reconfigure the Tuxedo domain.

```
[PSPPMSRV]
;===========================================================================
; Settings for PSPPMSRV
;===========================================================================

;---------------------------------------------------------------------------
; UBBGEN settings
; Since Performance Collators do not advertise services Max and Min
; Instances must always be equal
Min Instances=3
Max Instances=3
```

Though PSPPMSRV instances are created as a Tuxedo Services, they communicate with PPMI Servlet using HTTP, not JOLT. Therefore, traditional Tuxedo utilities used to gauge the load on a Tuxedo process will not produce accurate results. Also, there is no PSPPMSRV spawning, so the Min and Max Instances must be set to the same value.

Avoiding Message Duplication

Each of the PPMI monitor servlets will make sure only one copy of a SOAP message will be sent to a group of PSPPMSRV processes so that there is no duplicate data. If a 3rd party system is integrated and subscribes to the SOAP message, then it will be treated as a group different from the monitoring system. Again, only one copy of a SOAP message will be sent.

In the diagram on the previous page, AS1, AS2, and AS3 are different application server domains. Regardless of the number of PSPPMSRV processes in the AS1, AS2, and AS3 domains, the PPMI servlets ensure that only one copy of performance data is sent to those PSPPMSRV processes.

The PPMI Monitor Servlet applies a Cache Array Routing Protocol (CARP) algorithm to balance the load among the group of available performance collator (PSPPMSRV) processes. The same CARP logic applies to a 3rd party system with multiple processes listening to the SOAP messages.

TIPS ON ADMINISTERING THE PEOPLESOFT PERFORMANCE MONITOR

The PeopleSoft Performance Monitor is a complex tool that provides the performance statistics required to adjust a system configuration to best meet customer needs. This section is a collection of helpful recommendations for PPM system administrators.

Use an Alternate Monitoring System

You can direct all agents to send performance data from one performance monitoring system to another by changing the Monitor URL. The procedure is as follows:

1. Log into the monitored system(s).
2. Navigate to PeopleTools, Performance Monitor, Administration, Specify Monitor.
3. Change the URL to point to the new monitoring system and click the Save button.
4. **Reboot** the monitored system to make sure all of the agents pick up the new URL.

Note that once an agent is active, it does not check if the monitor URL has been changed. Therefore, it is important to reboot the servers so that the system recognizes the change to the Monitor URL.

Temporarily Disable the Monitoring of a System

To temporarily disable the monitoring of a system, perform the following steps:

1. Log into the monitoring system.
2. Navigate to **PeopleTools, Performance Monitor, Administration, Agent Filters**.
3. Select the monitored system to disable.
4. Select **01-Standby** in Reset All Filters.

5. Click the Apply button, then the Save and Notify button.

For this to work properly, the Integration Broker Gateway needs to be configured correctly. See the discussion in the previous section, **Setting up the Monitoring System**, **Step 8.**

Stop the Monitoring of a System

To stop the monitoring of a system, complete the following steps:

1. Log into the monitored system you want to stop monitoring.
2. Navigate to **PeopleTools, Performance Monitor, Administration, Specify Monitor**.
3. Enter NONE in the URL field.
4. Click the Save button.
5. Recycle the servers of the monitored system.

Delete Data of an Old Monitored System from the Performance Monitor System

To delete performance date from an old monitored system, complete the following steps:

1. Turn off monitoring of the system whose performance data you want to delete.

   If you did not turn off monitoring of the system (including recycling the servers), any active agent will register the system and reappear.
2. Delete Data
   a. Log into the monitoring system.
   b. Navigate to **PeopleTools, Performance Monitor, Administration, System Definitions.**
   c. Select the monitored system whose performance data you want to delete.
   d. Set the Archive Mode parameter to **Delete System**.
   e. Run the Schedule Archive Application Engine program to delete the system registration, agent registration, and performance data from the performance monitor tables of the monitoring system.

Purge All Monitored Data From the Performance Monitoring System

To purge all monitored data from the monitoring system, perform the following procedure:

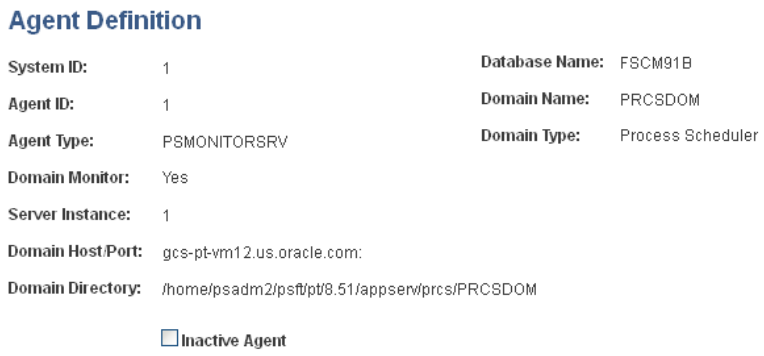
1. Shut down the performance monitoring system.
2. Log into Data Mover with a valid User ID.
3. Run the data mover script **PerfmonPurgeAll.dms** delivered in PeopleTools 8.45 and later <PS_HOME>/scripts. The System ID and Agent ID sequences will restart from 1.

You do not need to shut down all monitored systems, only the monitoring system. While the monitoring system is down, all of the active agents will stop sending data since the PPMI Monitor Servlet is not available. As soon as the purge completes and the monitoring system is booted, all active agents and systems will re-register and obtain a new ID from the PPMI Monitor Servlet.

Inactivate Unused Agents

Sometimes domains or servers definitions may no longer be valid due to configuration changes. Their agents will appear as Stale Agents on the System Performance page. You can stop these agents from appearing.

1. In the monitoring system, navigate to **PeopleTools, Performance Monitor, Administration, Agent Definitions**.
2. Locate the correct Agent Definition by the System ID, and Agent ID, Agent Type, and Domain Monitor flag.
3. Check the Inactive Agent box.
4. Click the Save button.



Removing Timed Out PMUs

The PMUs generated by the transactions are stored in the current PMU table, PSPMTRANSCURR. When PSPPMSRV gets notified that an open PMU has finished by receiving a stop, it flags the corresponding PMU start and update rows in the current PSMTRANSCURR table for deletion. When the Reaper Application Engine program executes, it deletes all the rows in the current PMU table (PSPMTRANSCURR) that are flagged for deletion, and inserts a row for the completed PMU in the PSPMTRANSHIST table.

However, some transactions may run for a long time and exceed the timeout value allowed, which results in "timed out PMUs." When the Reaper program executes, it sets the status for expired PMUs to timed out in the current tables, and then moves the timed out PMUs to the history PMU table, PSPMTRANSHIST.

Reduce PMU Sampling

You may not want to monitor every request submitted to the system. To reduce the amount of performance data collected, set the Agent PMU Sample Rate on the System Definitions page to a positive integer "N." As a result only one out of every N server requests generates PMUs. Once a request is being sampled, all PMUs are generated at the Agent Filter Level set for each agent type involved in processing the request.

The following PMU types are always monitored regardless of the Agent PMU Sampling Rate setting.

- PMU 108-user sign off
- PMU 109-user log on
- PMU 116-user being redirected to other sites

Use Sample Queries to Extract Information

Performance Monitor comes with the following three sample queries. Each query has two versions. One version queries the PPM history tables, and the other queries the PPM archive tables. Note that the date/time format is "MM/DD/yyyy hh[:mm:ss(A/P)M]" or "MM/DD/yyyy hh[:mm:ss]" if using a 24-hour clock, where month (MM) and date (DD) are numeric.

- Component Cache Misses: This query returns all application server requests for a specific system that had to retrieve metadata from the database as opposed to the cache. It also shows the file cache and memory cache for comparison.
- Timed Out SQL Statements: This query returns information from the PMU (history or archive) table while joining information stored in the event table. It retrieves all PMU 400's (Tuxedo Service PCode and SQL) that were running SQL statements when an Event 500 (JOLT Service Exception) was received. It is assumed that this exception occurred because of a timeout, but it could also have been due to an application server outage or a JOLT error.
- Server Process Starting Count This query returns the number of times a server process boots in a web server, application server, or process scheduler domain within a specified period.

Export and Import Performance Data

Since PeopleTools 8.45 a data mover script, **PerfmonDataExport.dms**, is delivered that allows you to export performance data from the monitoring system database for a specific System ID, a specified time range, or a performance trace.

You need to edit the script so that it is appropriate for your environment. This script can also be used on monitoring systems running PeopleTools 8.44 as well.

Since PeopleTools 8.45 an associated data mover script, **PerfmonDataImport.dms**, that allows you to import performance data extracted by the **PerfmonDataExport.dms** script. This script can also be used on monitoring systems running PeopleTools 8.44 as well.

Note: Do not run PerfmonDataImport.dms on a "live" system. The script uses the REPLACE * command. As a result, current data may be lost.

Understand when a Reboot of the Web and Application Server After Configuration Changes is Required

Here are some guidelines regarding when you should restart various system's components after making changes:

- If you change the PPMI URL, the entire monitoring system must be rebooted, including the web server, application server, and process scheduler domains.
- If you change the Monitor URL on a monitored system, the entire monitored system must be rebooted so that all of the agents can pick up the new URL address. This is important even if the value is set to NONE.
- If you switch the Agent Filter Level from an active value (**01-Standard**, **02-Error**, **03-Warning, 04-Standard, 05-Verbose**, **06-Debug**) to **OFF**, the entire monitored system must be rebooted.
- If the system is setup correctly, a reboot should not be required for other changes to other agent configuration parameters. Clicking the Save and Notify button notifies the PPMI Monitor Servlet of the configuration changes.

Understand the Result of Using Save and Notify

When you press this button, an application message is sent by the PeopleSoft Integration Broker Gateway to the PPMI Monitor Servlet notifying it of the PPM configuration changes. The PPMI Monitor Servlet, in turn, passes these changes along to the affected agents during subsequent conversations. The changes are stored in the database for persistence as well.

Note: When you update the Agent Filter Level, or any other PPM Parameter, be sure to click the Save button to ensure that the change is saved to the database. If you forget to click the Save button prior to clicking the Save and Notify button, your PPM parameter changes may not be updated when you click the Save and Notify button.

Since the PPMI Monitor Servlet waits for the agents to initiate communication, there may be a delay in the publishing of changes to the monitored systems. The maximum delay is the Agent Heartbeat Interval in: **PeopleTools, Performance Monitor, Administration, System Definitions**.

Make Necessary Changes when Cloning Databases

In the process of building a new database, whether it is for general application or PPM use, you must ensure that the information used to identify the databases is unique. If you clone a database, make sure to:

1. Ensure that there are no PIA server components, e.g. web servers, application servers, process scheduler, etc., running for the PPM database.
2. Using a command line SQL editor, set monitor URLs to NONE for the rows with URL_ID = "PPM_PPMI" and "PPM_Monitor" in PSURLDEFN table.
3. Using a command line SQL editor, Set GUID = '<space>' in PSOPTIONS table. This should be done ***before any*** application server domain and web server domain is connected with the database instance. This value will be repopulated when the first domain is started for the database.
4. Delete any and all cache files from any PIA server components that attach to this database.
5. Restart the system as normal.

Note: Generally speaking, direct manipulation of the PeopleTools metadata tables is not a GSC-supported activity. In the above case, no on-line tools are provided to make these changes, so this procedure is provided here. It is not a general purpose process and is very specific to these objects. This procedure will not work in all cases where the metadata is directly manipulated.

Disable Agent Event Sampling

It is possible to eliminate Event sampling by setting the Agent Event Sampling Rate to 0. This parameter is expressed in milliseconds, and normally defaults to 300,000, or 5 minutes. As a result, the agents will not report server events.

To disable agents reporting events:

a. Log into the monitoring system.

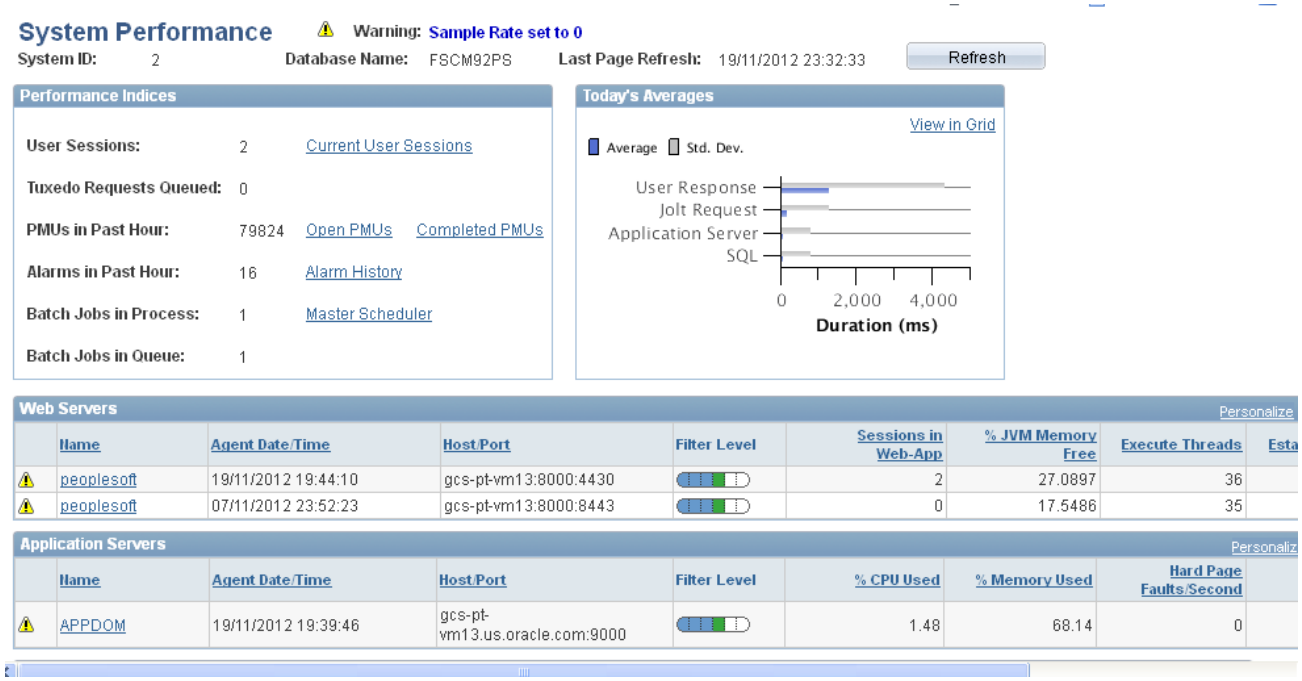
b. Navigate to **PeopleTools, Performance Monitor, Administration, System Definitions**.

c. Set the Agent Event Sampling Rate to 0.

d. Click the Apply to Current Systems button.

This setting applies to all agents in the monitored system. The "Stale Agent Data Detected" message will appear in the System Performance page because the agents are no longer reporting event data. Agents will continue to collect and send PMU data if the Agent Filter Level is set to **04-Standard** or higher.

The following screenshot is what you will see on the System Performance page. The warning message indicates that the Agent Event Sampling Rate is set to 0. The agents no longer report server events but will continue monitoring user requests and reporting PMUs.



Perform PPM Table Maintenance

In the monitoring system, if you select Archive Data or Archive Nothing as the archive mode for any of the monitored systems, make sure the archive tables (PSPMTRANSARCH, PSPMEVENTSARCH) or history tables (PSPMTRANSHIST, PSPMEVENTHIST) are manually cleaned up on a regular basis in the monitored systems.

The Performance Monitor Archive Data Application Engine program moves any performance data older than the Retention History (based on the value of the After N Days parameter specified on the System Definitions page) from the history tables to the archive tables. If the parameter is set to **Archive Nothing**, it means that no performance data will be moved from the history to the archive files when the PPM Archive Application Engine program is run. Neither archive mode purges any performance data; so the PPM tables in the database of the monitoring system will continue to grow unless manually maintained.

Purge Data as Necessary

Since PPM can generate large volumes of data it is critical to ensure that the maintenance Application Engine programs are properly configured and are run at appropriate intervals.

If the amount of data that has been collected in the monitoring system has grown "out of control" or if you want to remove all of the transactional data from the PPM, there are two approaches that you can use depending on your particular situation:

1. DataMover Script: A DataMover script, **PerfmonPurgeAll.dms,** is provided that will delete the data and set some other required variables.
2. Custom Script: Since DataMover does not currently support the TRUNCATE command due to the lack of consistency in support for the function across platforms, the only way to remove data using DataMover is to issue an SQL DELETE. DELETE is a logged transaction and as such can be very inefficient against large data sets common to PPM. If this is the case, then use that script as a template and rewrite it using TRUNCATE commands and issue that script with your preferred SQL tool.

SPECIAL CONSIDERATIONS

Load Balancer

For advice regarding integrating a Load Balancer into a monitoring system architecture, refer to the Performance Monitor PeopleBook, Chapter 4:

Administering the PeopleSoft Performance Monitor > Setting up Monitor Clusters

and Chapter 9:

Understanding PeopleSoft Performance Monitor Security Considerations.

Firewalls

You can have a firewall between monitored and monitoring systems, but the monitoring system web server's listening port must be open to allow agent communications.

We don't recommend a firewall between the web server and application server in the PPM monitoring system. If you choose to install one, it is critical to understand that the conversation between the PSPPMSRV and the PPMI Monitoring Servlet can be on any port allocated by the operating system. To reduce setup effort, a static port is not used for this conversation. As a result of PPM's need to access ports dynamically and with little predictability, setting up a firewall between the web server and application server on the monitored system is very difficult to configure and is therefore highly discouraged. For more information, refer to

Performance Monitor PeopleBook > Chapter 9 Understanding PeopleSoft Performance Monitor Security Considerations.

Upgrading Monitored and Monitoring Systems

The best strategy for taking advantage of PPM's available functionality is to use the latest version of Enterprise PeopleTools in the monitoring system. For any monitored system, you can mix and match the versions of Enterprise PeopleTools as long as they are at a minimum of 8.44.

When upgrading the monitoring system, you can either migrate the current instance of the database or use a fresh instance of the upgraded version of the PTSYS database. Using the latter approach means that you lose all the performance data and have to repeat the monitoring system installation steps again. You will also lose the registered System ID and Agent IDs.

You can export definition information from the current monitoring system by running the Export Definitional Elements section in the export data mover script, **PerfmonDataExport.dms.** Be sure to replace the initial System ID value of **9999** with a meaningful System ID. Then, run the import data mover script**, PerfmonDataImport.dms**. Finally, update the values of PM_AGENT_SEQUENCE, and PM_SYSTEM_SEQUENCE in the PSPMMONITORGBL table to the highest Agent Ids and System IDs, respectively.

Partially Enabling Performance Monitor

When Performance Monitor is enabled, all agents across all of the web server and application server domains are activated. To disable the agents in specific application server domains or process scheduler domains, follow these steps:

1. Shutdown the monitored server domain(s) that you want to disable.
2. Set EnablePPM Agent = 0 in the corresponding configuration files, psappsrv.cfg and psprcs.cfg.
3. Start PSADMIN. Select the corresponding application server domain. Select “Configure this domain”, then “Load configuration as shown”.
4. Boot the application server or process scheduler domain(s).

To disable the agents in a specific web server domain, follow these steps:

1. Create a separate web profile for use by the web server domain whose agent you want to disable.
2. In that new Web Profile, navigate to the General tab and ensure that the Enable PPM Agent checkbox is unchecked.
3. Deploy or redeploy PIA, or edit the configuration.properties file for that domain to reflect the new web profile.
4. Restart the web server as necessary.

Using HTTPS

For information on how to implement HTTPS in the PPM infrastructure, refer to the Performance Monitor PeopleBook, Chapter 4

> *Administering the PeopleSoft Performance Monitor > Setting up Monitor Clusters* and Chapter 9 *Understanding PeopleSoft Performance Monitor Security Considerations*

PeopleSoft Ping (PSPing)

PeopleSoft Ping is a diagnostic utility that allows you to troubleshoot possible infrastructure/connectivity issues. If you enter the ping URLs for the monitored systems in the URL catalog in the monitoring system, you can launch the ping test from the monitoring system instead of having to sign on to the monitored system. The following requirements must be in place:

- Single Sign on must be configured between the monitored and monitoring systems. Refer to the Security Administration PeopleBook.
- The Current User ID must be a valid User ID in both the monitored and monitoring systems.
- The Current User ID must have permission to access the PSPing page in the monitored system. For example, the PeopleSoft User Role (via the PTPT1000 permission list) allows a user to access these PSPing options, PTPERF_TEST, PSPING_CHART, PSPING_DELETE, and PINGOPTIONS.

Any system that you intend to ping must have an entry in the URL catalog for the PSPing page. The URL must contain PTPERF_TEST in order for the URL to appear in the ping lists associated with the PeopleSoft Performance Monitor. For example,

```
http://server_name/psp/<site>[_newwin]/EMPLOYEE/<PT_LOCAL>/c/UTILITIES.PTPERF_TEST.GBL
```

Replace PT_LOCAL, if a different portal node is used.

If the string “*_newwin*” is issued, a new browser window pops up when the Execute button is clicked. This second window allows you to continue using both the previous window for Performance Monitor and the new window to run the PSPing process.

Chapter 3 – Understanding Performance Data

The PeopleSoft Performance Monitor is built on the assumption that transactions progress through several states where performance data about the transactions can be captured. This section discusses the two types of performance data.

TYPES OF PERFORMANCE DATA

The two principle types of performance data reported are PMUs and Events.

- A *Performance Measurement Unit*, or PMU, is a unit of measure of activity that typically has a duration, can be defined with start and stop times, and can be related to other activities hierarchically. Examples of PMUs include Jolt transactions and SQL executions.
- *Events* are notifications containing performance metrics of a component of the PPM architecture. Events differ from PMUs in that they cannot be hierarchically organized and are "point in time" in nature i.e. they do not have durations. Examples of Events include CPU usage or JVM memory allocation.

Performance Measurement Unit (PMU)

A PMU is a unit of measure that reflects the execution of a section of code in the PeopleSoft infrastructure. The system "starts" and "stops" a PMU at specific code locations, and may update a PMU anytime between the start and stop times. PeopleTools has defined a fixed set of PMU types, and has inserted instrumentation for each PMU type at specific code locations; e.g. when an SQL Execute occurs in a PSAPPSRV process, or when a JOLT request is issued by the web server.

Each PMU reports:

- The Agent ID (which server instance processed the request)
- The User ID (who submitted the request)
- Agent start time (time when the PMU initiated)
- Monitor start time (time when the PPMI Monitor Servlet received the request)
- Duration (elapsed time of the PMU), instance IDs (for the PMU itself, its immediate parent, and the top of the PMU tree)
- Metrics (such as the number of SQL fetches or buffer size used in a JOLT response)

PPM uses high resolution Operating System calls to collect timing information. All supported platforms can be measured to 1 microsecond accuracy except for Windows which is at 10 microsecond accuracy.

Open PMUs are those that are currently processing, i.e. PMUs for which a "stop" has not yet been received. Completed PMUs are those for which a "stop" has been received.

PMUs can also assume a parent-child relationship. Child-PMUs start within a parent-PMU, and are linked to their parents by the parent Instance ID. You can view PMUs within a tree structure that reveals the hierarchy of parent-child PMUs and indicates the processing times at each level.

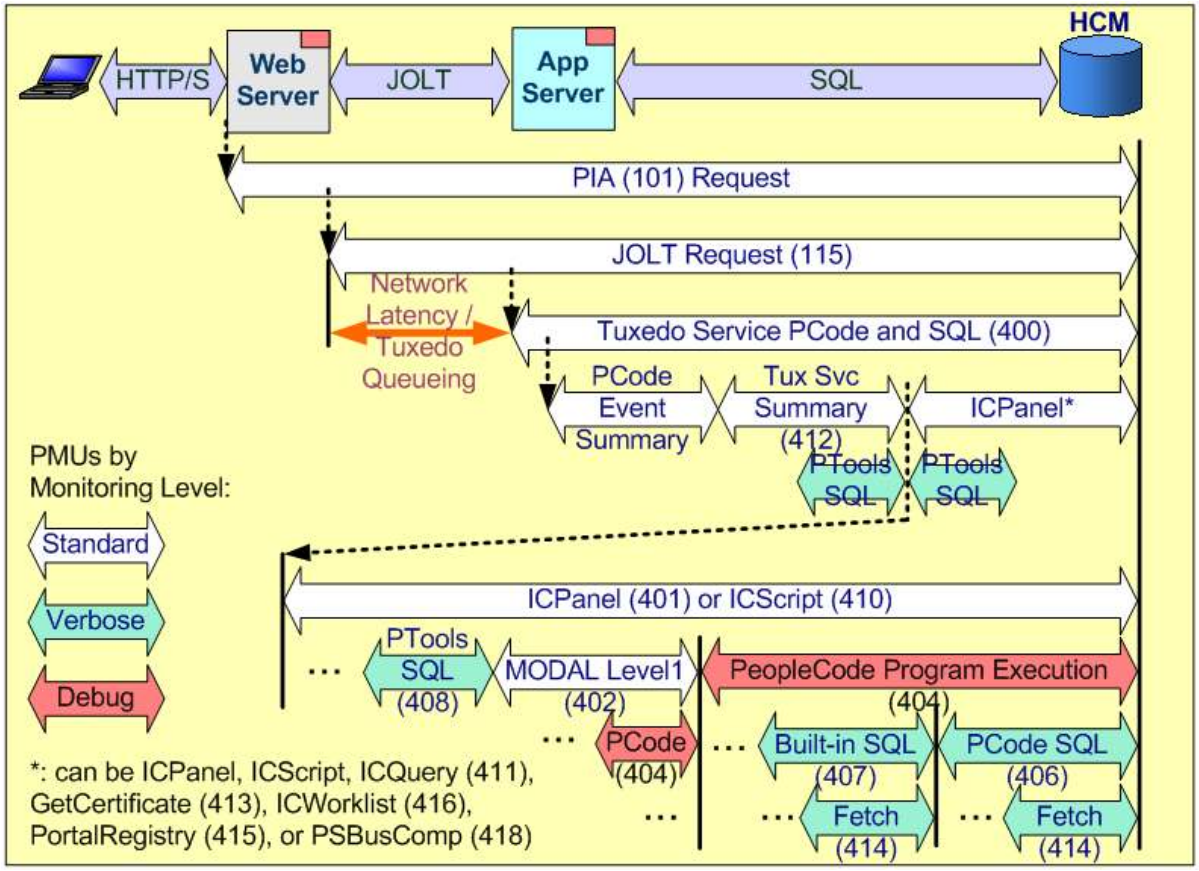
PMUs are defined in the PSPMTRANSDEFN table. Notice that each row contains a PMU definition set ID. PeopleTools uses the tableset of PMU types that are identified by Set ID 1. Each PMU can have up to three Context fields, seven Metric fields , and one Additional Description field. Context values are common to the entire user request or a specific server tier within PIA. Context field can store up to 254 characters. Metric values are the measurement taken by the monitor agents during the lifetime of a PMU. Metrics 1-6 are numeric. Metric 7 can store up to 128 alphanumeric characters. The Additional Description field has a Long Character type.

Customers cannot create new PMUs or Events, only PeopleSoft can instrument the PeopleTools binary code to expand the set of PMUs or Events. Simply adding new PMU and Event definitions to Performance Monitor metadata does not enable instrumentation.

The following diagram is crucial to understanding when PMU's are triggered and at what Monitoring Level they are collected. The color coding reflect the minimum monitoring level required to capture a PMU, e.g.:

- A monitoring level of "Standard" is required to capture a report of PIA request performance. This is captured in PMU type 101.

- A monitoring level of "Verbose" is required to capture to track SQL performance. It will also capture everything indicated with the "Standard" level.
- A monitoring level of "Debug" is required to capture PeopleCode program execution statistics. Other logging levels do not capture this information. . It will also capture everything indicated with the "Standard" and "Debug" level.



Some points of clarification to note about this diagram:

- The difference in the duration time of PMU 115 and its corresponding PMU 400 is the total time spent on JOLT network latency, waiting time in the Tuxedo queue, serialization and deserialization of the full request/reply at the web server, and the serialization and deserialization of the User ID information on the application server.
- The difference in the duration time of PMU 100 (Portal request) or 101 (PIA request) and its corresponding PMU(s) 115 is the total time spent on the web server for everything excluding the serialization and deserialization of the full JOLT request/reply.
- A Portal request (PMU 100) may contain one or more JOLT requests (PMU 115) and may sometimes have GetContent (PMU 117) and PIA Request from Portal (PMU 106) PMUs in between them. A Portal request is processed by the portal servlet (psp). PIA requests and PIA requests from Portal are processed by the PIA servlet (psc). The following two PMU trees demonstrate the difference between a PIA request and a PIA Portal request.

PMU Tree Example: PIA Request (PMU 101)

PMU Tree Example: PIA Portal Request (PMU 100)

Event Data

Events are notifications containing performance metrics that are different from PMUs in that they are not hierarchical and they do not have durations. Events relate to resource usage, such as CPU usage or memory allocation. PeopleTools has defined a set of Event types, and each type of Event is reported at a specific location in the instrumented code.

The current version only instruments **03-Warning**, **04-Standard**, and **06-Debug** Events. All events are defined in the PSPMEVENTDEFN table. You can review the various events online by navigating to

PeopleTools, Performance Monitor, Administration, Meta-data, Event Definitions.

Notice that each row contains an event definition set ID. PeopleTools uses set ID 1. The delivered events include:

Warning Events

Some events indicate warning condition. These include:

- Event 355 reports when a query process has been killed by the PSMONITORSRV process by a user's action on the monitored system.
- Event 356 [PeopleTools 8.46 and after] records any occurrences of server recycle and shutdown. Navigate to **PeopleTools, Performance Monitor, System Monitor, Recycled Processes Diagnosis** to get a list of recycled server events. Follow the Event History and Completed PMUs links to find out the user requests associated with the event.
- Event 500 reports JOLT service exceptions, such as Tuxedo service timeouts, or failures during a JOLT call.
- Event 801 reports PPMI Monitor Servlet buffer overruns. If you are seeing this event, make sure the monitor Agent Filter Level is not set to **05-Verbose** or **06-Debug,** because large volumes of monitoring data will be generated if the agents are set to either of the two Agent Filter Levels. If that is not the case and it happens during normal user load and monitoring, then other possibilities to consider are:

1. There are too many monitor agents.
2. The PPMI Monitor Servlet is being throttled by infrastructure limitations.
3. The performance collators are not fast enough.

You might want to experiment with increasing the PPM Monitor Buffer Size field value in the Web Profile, or increasing the number of instances of the performance collator (PSPPMSRV) processes.

Below is an example showing the PPMI Monitor Servlet buffer level in the Show Clients page from the PPMI Monitor Servlet diagnostic URL.

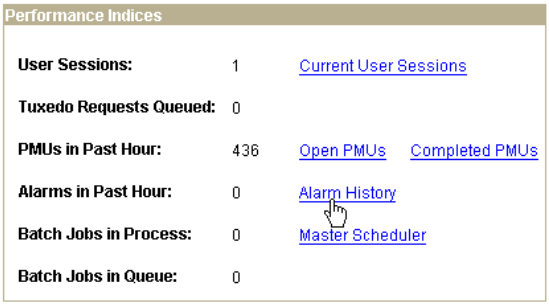
Clients -- CurrentTime: Thu Nov 22 21:20:15 EST 2012

Group	Id	Url	Queue Length	Estimated Queue Size	Item Processed	Estimated Bytes Processed	Max Size	Running Avg Size	LastComm	Limit
25fc6bc2-5ce8-11e1-8803-fc2cef823c43	3ca00000004	http://gcs-pt-vm13.us.oracle.com:43780	0	0	120536	119473284	2159312	78399	Nov 22, 2012 9:19:40 PM	52428800

1 Sessions, 1 clients

- Event 803 reports any data lost error due to PPMI client.
- Event 900 reports any agent buffer overruns. Make sure that the monitor Agent Filter Level is not set to **05-Verbose** or **06-Debug**. Large volumes of monitoring data can be generated if the agents are at either the **05-Verbose** or **06-Debug** Agent Filter Level. If this is not the case, you might want to experiment with increasing the Agent Max Buffer Size parameter in the System Definitions page of the monitored system.

  The System Performance page displays any Alarms reported in the last hour. Simply click on the Alarm History link to bring up the list of events.



PERFORMANCE DATA ATTRIBUTES

There are two significant attributes to performance data, Context and Metrics.

1. Context - Each PMU contains three context value fields. Context values are common to the entire user request or a specific server tier within PIA. Performance Monitor uses context to “flatten” a PMU tree and simplifies the database schema. For example, you do not need to navigate up from a PeopleCode SQL Execution PMU to an ICPanel PMU to see which PeopleSoft Component generated a particular SQL statement. All of the defined context types are stored in the PSPMCONTEXTDEFN table. You can review them online by navigating to

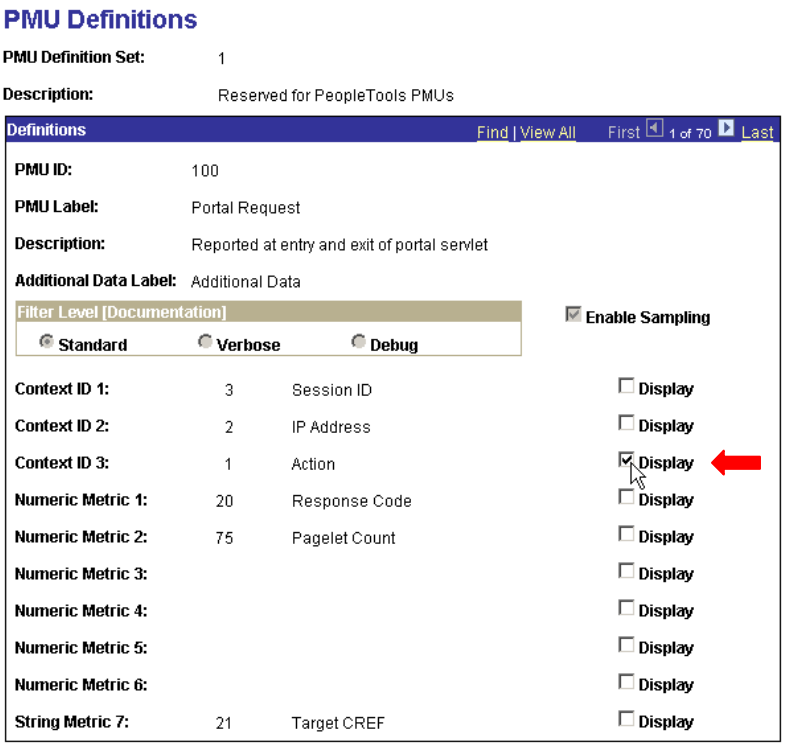
   PeopleTools, Performance Monitor, Administration, Meta-data, Context Definitions.
2. Metrics - Metrics are statistics in the form of either a value or content measured by the Performance Monitor agents. All the metrics types are stored in the PSPMMETRICDEFN table. You can review the metric types online by navigating to

   PeopleTools, Performance Monitor, Administration, Meta-data, Metric Definitions, Filter Level.

Tailoring the PMU Detail Page Display

You can tailor whether a context or metric value associated with a PMU appears in the detailed PMU page. Navigate to **PeopleTools, Performance Monitor, Administration, Meta-Data, PMU Definitions**, find the PMU and check or uncheck the Display checkbox.

Note: If you make changes to these display options, record the default display option. Performance Monitor cannot automatically reset back to factory defaults.



ENABLING TRACING

The Performance Monitor enables you to monitor and record performance information for activity on PeopleSoft systems. Our discussion to this point has focused on using PPM to monitor a system's overall performance on a global basis. Using PPM in this manner is appropriate when you are trying to analyze and troubleshoot overall system performance, but it does put some stress on the monitored systems.

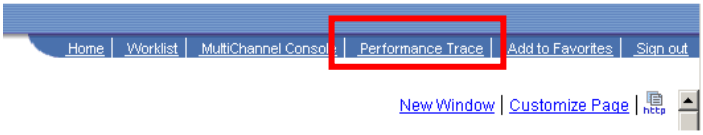
Another PPM option, called Performance Trace, is available for troubleshooting performance issues related to specific business processes, or specific users. This option puts less stress on the monitored system as a whole, and is relatively easy to activate or deactivate as required.

Performance Trace

Performance Trace is a very powerful and practical tool for troubleshooting performance issues in a production environment, because it allows for collection of very detailed information to help determine the causes of performance problems without impacting other users on the system.

End users who have a permission list containing WEBLIB_PPM web libraries can access the Performance Trace menu in the portal header. The menu allows a user to instruct all monitor agents to track his/her interactions with the PeopleSoft system(s), and override the Agent Filter Level during the user session. It is a powerful tool because of its ability to pinpoint specific steps within a business process that are causing performance issues.

Note: Sometimes the Performance Trace menu does not appear even after the required permission list is added for a user. This can be resolved in one of three ways: (1) Force a browser refresh. With Internet Explorer, while the PeopleSoft web page is active, hold down the Control key and click the refresh toolbar icon. (2) Delete the browser cache and refresh the page. (3) Sign out. Close the Browser. Reopen the browser and sign in.



After clicking the Performance Trace menu, a window pops up.

The user can then name the Performance Trace and set the Agent Filter Level for his/her user session. Clicking the Start Trace button will activate the trace. The user should then switch back to their active web page and conduct business processing as usual. Clicking the Stop Trace button ends the trace. It is all right to reuse the same performance trace name for future traces.

The new Agent Filter Level only applies to the particular user in the active user session with the monitored system. If the user session times out, the Agent Filter Level resets itself back to that of the monitored system. If a user closes the Performance Trace Console window before stopping the trace, the performance trace will continue until the user signs out or the session times out.

Performance Trace also works well when a user is authenticated across multiple PeopleSoft systems which are all monitored. For example, if the user executes a transaction requiring integration across systems behind the scenes, the monitor agents will carry the trace and filter level across the systems as well. This means, for example, that you can follow a View Paycheck request from the Portal to an HCM system.

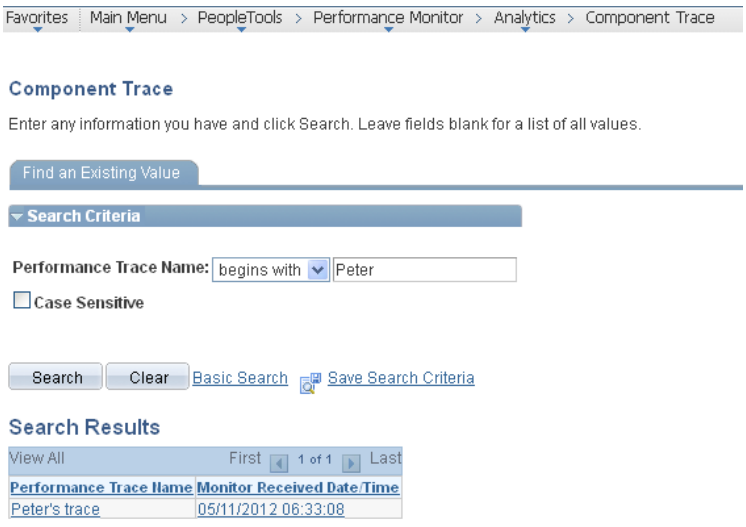
There are several ways to review the results of a Performance Trace. The most common method is through the User Requests page. To access it, navigate to

PeopleTools, Performance Monitor, Analytics, User Requests.

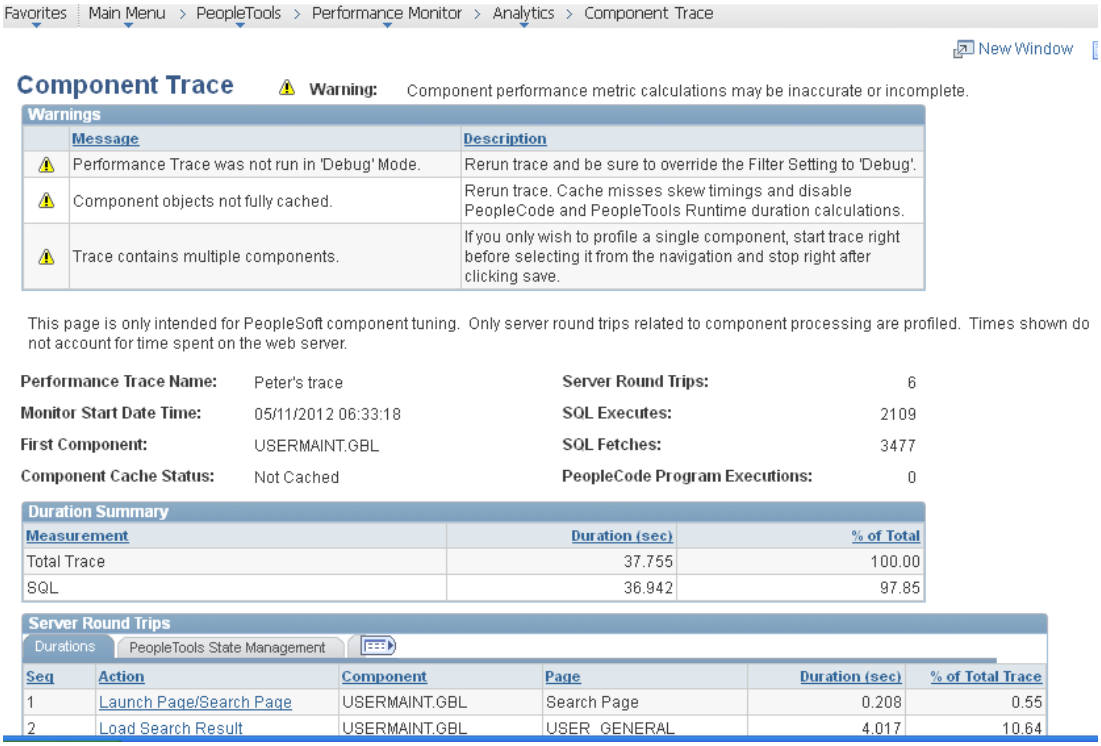
Select the Performance Trace Name from the dropdown list if the Lookup Application Engine program has been run. If not, enter the full case-sensitive text.

For detailed analysis, use the Component Trace option by navigating to

PeopleTools, Performance Monitor, Analytics, Component Trace.



Click on the Trace name to get further details displayed.

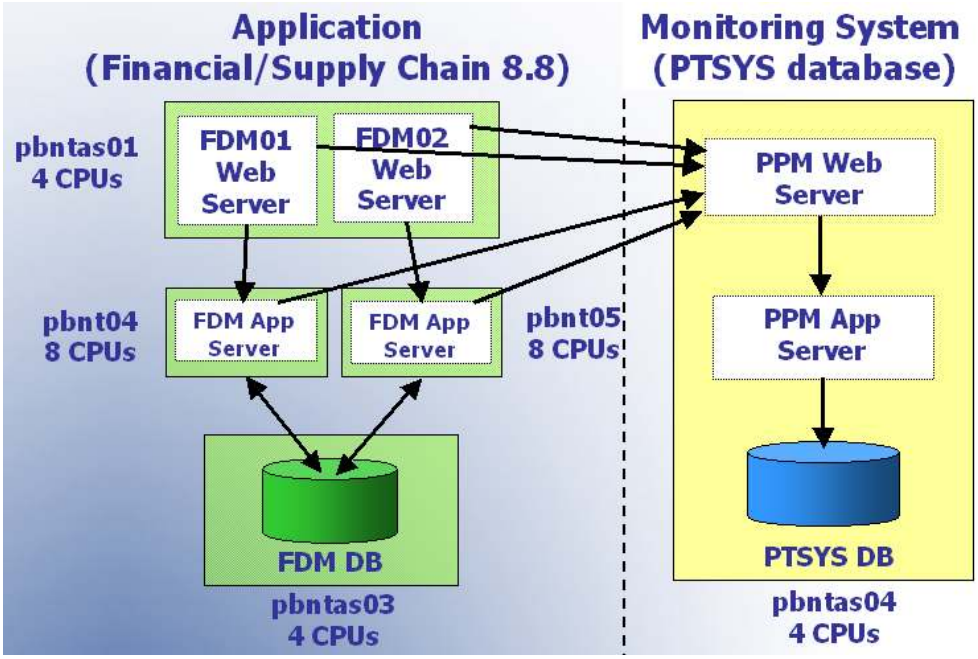


Chapter 4 – Performance Monitor Overhead

Enabling Performance Monitor requires additional CPU and memory resources for the Monitoring Agents to gather performance data. Agents send the data to the PPMI Monitor Servlet using HTTP, therefore network bandwidth is also required.

PERFORMANCE MONITOR BENCHMARKS

Below is the benchmark information based on Financial/Supply Chain 8.8 application running on PeopleTools 8.45.



The hardware specifications are:

Machine Name	pbntas01	pbnt04/05	pbnt03	pbntas04
Description	Financial/Supply Chain – web server (2 instances)	Financial/Supply Chain – App Servers	Financial/Supply Chain Database Server	PTSYS Performance Monitor Environment (web, application, and database server)
OS	Windows 2000 SP3	Windows 2000 SP4	Windows 2000 SP3	Windows 2000 SP3
Model	HP Proliant DL360 G3	HP Proliant DL560 G1	HP Proliant DL360 G3	HP Proliant DL360 G3
No. of CPUs	4	8	4	4
CPU Speed	2.80 GHz	2.50 GHz	2.80 GHz	2.80 GHz
Memory	2.4 GB	3.7 GB	2.4 GB	2.4 GB
Storage	60 GB	20 GB	60 GB	60 GB

The benchmark simulates a scenario with a mix of transactions: heavy, medium, small and Kiosk usage. The benchmark is conducted with 1000 concurrent user sessions. Pacing is 6 minutes between user sessions. Each user exercises 10 sessions during the benchmark. The duration of the test takes 60-75 minutes in addition to an allowance of 15 minutes ramp up time. The distribution of transactions among the users are:

Transaction Name	No. of Users
Order Management – Order Entry	200
Customer Add	300
Vendor Enquiry	200
Edit User Profile Defaults	300

Each of the FCM application server domains is configured with 10 instances of application server (PSAPPSRV). The PPM application server domain has 3 instances of the PSPPMSRV process. No data loss was recorded. We observe similar benchmark results when the application server recycle count increases from the default value to 10,000.

CPU OVERHEAD

The benchmark result of CPU utilization.

Server	No PPM	PPM on Standby	PPM on Standard
application server (monitored system)	38.20%	38.25%	39.75%
web server (monitored system)	7.48%	7.62%	8.01%
PPM Server (monitoring system)	NA	0.54%	3.44%

Notes:

No PPM – Performance Monitor agents are disabled in the web server and application server domains in the monitored system. Refer to PeopleSoft Performance Monitor PeopleBook, Chapter 4 – Administering the Performance Monitor, Section – Disabling PeopleSoft Performance Monitor Agents.

PPM on Standby – Performance Monitor agents in the web server and application server are set to the **01-Standby** Agent Filter Level in the monitored system. All agents are active but not collecting or sending performance data. The only activity by an agent is contacting the monitoring server at a fixed Agent Heartbeat Interval.

PPM on Standard – Performance Monitor agents are set to the **04-Standard** Agent Filter Level in the monitored system.

For the monitored system (application server and web server rows in the table above),

- Performance Monitor on **Standby** mode has negligible resource impact.
- Between No PPM and PPM on **Standard**, the impact on the web server and application server is less than 0.5% and 2% increase in CPU utilization, respectively.

For the performance monitoring PPMI server,

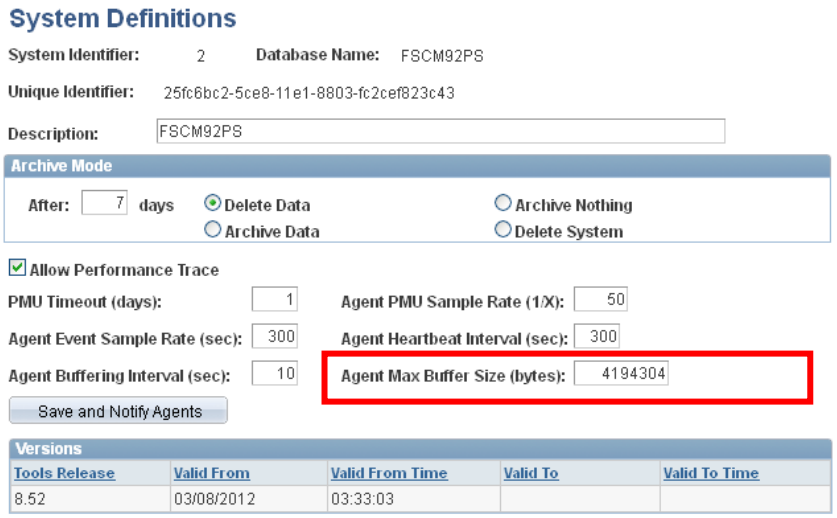
- The 3.44% CPU utilization in the PPM on Standard column includes web server, application server, process scheduler and the database server for the monitoring system.

- Application server has two PSAPPSRV processes, and two performance monitor data collator processes (PSPPMSRV) for inserting performance data.
- The benchmark reflects resource utilization on receiving and inserting performance data.
- It does not simulate users using performance monitor from the browser.
- It does not schedule Performance Data Archive and Lookup Maintenance.
- It assumes the Reaper program is scheduled to run every 15 minutes.

The benchmark does not use SSL (secured socket layer). Using SSL to secure the communication between Agents and Monitoring Servlet might increase the overhead. You may consider using hardware SSL accelerators to offload the CPU overhead.

MEMORY OVERHEAD

Each Performance Monitor Agent stores the collected data in internal buffers. The maximum buffer size is specified in the System Definitions page, **PeopleTools, Performance Monitor, Administration, System Definitions**.



The maximum memory overhead on a machine that hosts an application server Tuxedo domain can be calculated as:

$$Max\ Buffer\ Size\ specified\ in\ the\ System\ Definition \times (Number\ of\ \ Appservers + 1)\ \text{bytes}$$

The above formula takes into account the presence of an additional server, PSMONITORSRV. The PUBSUB and other servers do not contain any overhead because they are not currently monitored. Application server processes load an Agent that is written in Java and hence share the agent buffer space with other JVM usage like Charting and PeopleCode Java Objects. Therefore, in addition to the change in the System Definition, it may be necessary to increase the JVM's maximum memory allocation in psappsrv.cfg.

The maximum memory overhead on a machine that hosts a web server can be calculated as:

$$Max\ Buffer\ Size\ specified\ in\ the\ System\ Definition \times (Number\ of\ \ Sites + 1)\ \text{bytes}$$

The web server domain contains a domain wide Agent to gather machine resource consumption data.

The maximum memory overhead on a machine that hosts a process scheduler server can be calculated as:

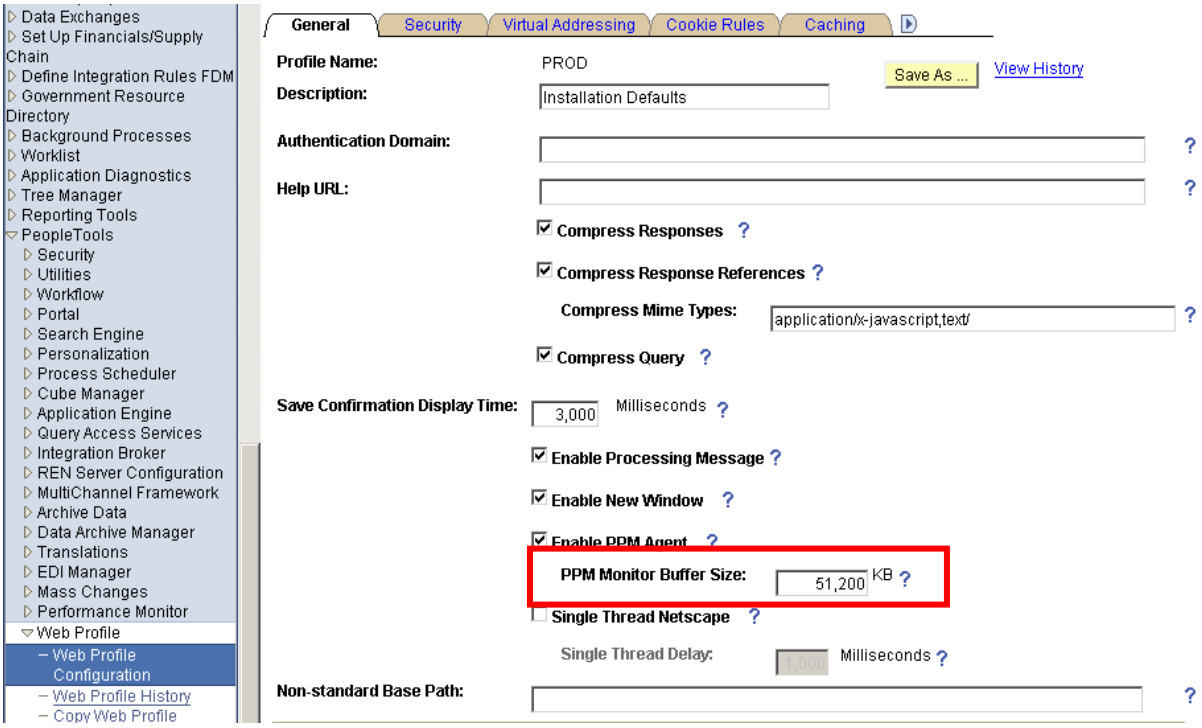
$$Max\ Buffer\ Size\ specified\ in\ the\ System\ Definition \times (Master\ Scheduler + 1)\ \text{bytes}$$

The process scheduler domain also hosts the PSMONITORSRV process. The Performance Monitor Agent is active in the Master Scheduler only. The individual process schedulers are not currently monitored.

Note: When the agent buffer overrun occurs, PPM logs a warning (Event 900) for the agent. Some data may be lost at this time. Collection of performance data will resume as soon as the used buffer falls below the limit.

Apart from the Agent buffers, the Monitoring Servlet also allocates a buffer to store the data sent by the Agents. Depending on the number of registered Agents and the amount of data collected, it may be necessary to increase the size of the buffer. The default value for the size of the buffer in the Monitoring Servlet is 50 MB (for PeopleTools 8.45 and higher) and 40 MB (for PeopleTools 8.44). This value can be changed in the Web Profile configuration pages **PeopleTools, Web Profile, Web Profile Configuration.**

Note: When the PPMI Monitor Servlet buffer associated with a PPMI client (a performance collator or a 3rd party client) is overrun, a warning (Event 801) is logged. Some data may be lost at this time. Publication of performance data to the PPMI client will resume as soon as the used buffer falls below the limit.



DATABASE OVERHEAD

During the test of 1,000 concurrent users, the number of rows of event and PMU data collected were 2,000 and 740,000, respectively. The table below shows the results for an IBM DB2 database, based on the 1000 concurrent users benchmark specified above.

	Before	After	Total
User Space (pages)	53,848	378,384	324,536
Index Space (pages)	46,696	238,104	191,408
Total Pages	-	-	515,944
In size (in MB)	-	-	2,015

NETWORK OVERHEAD

The table below is the benchmark result on network bandwidth based on 1000 concurrent users. Notice the significant increase in network bandwidth when all of the agents are monitoring at **05-Verbose** Agent Filter Level.

PPM Agent Filter Level	Bandwidth due to PPM	% of 10 Mbps Network	% of 100 Mbps Network
Standard	720 K bits / second	7.2	0.72
Verbose	4800 K bits / second	48	4.8

Performance Monitor Agents consume network bandwidth while sending performance data to the PPMI Monitor Servlet using HTTP. The approximate bandwidth used by the Agents can be calculated as:

$$\frac{Number\ of\ \ Agents \times Max\ Buffer\ Size\ specified\ in\ the\ System\ Definition \times 60}{Buffer\ Interval\ specified\ in\ the\ System\ Definition}\ \text{bytes/min}$$

Additional bandwidth is required for the communication between the PPMI Servlet and the PSPPMSRV process.

The communication between Agents and the PPMI Monitor Servlet is carried out using HTTP/1.1 using the **Keep Alive** option. If the **Keep Alive** option is turned off on the web server then the Agent disconnects and re-establishes the connection every time it has to send performance data.

While using SSL for securing the communication between Agent and Monitoring Server, key exchange is the most expensive operation. The Agent performs SSL Session caching so that the re-connect operation is less expensive.

Chapter 5 –Troubleshooting Example Performance Issues

This chapter discusses common use cases of utilizing Performance Monitor to troubleshoot performance issues.

SLOW SIGN ON

Common reasons for poor performance during sign on include:

- Loading multiple pagelets.
- Slow User authentication.
- Network latency between the browser client and web server.

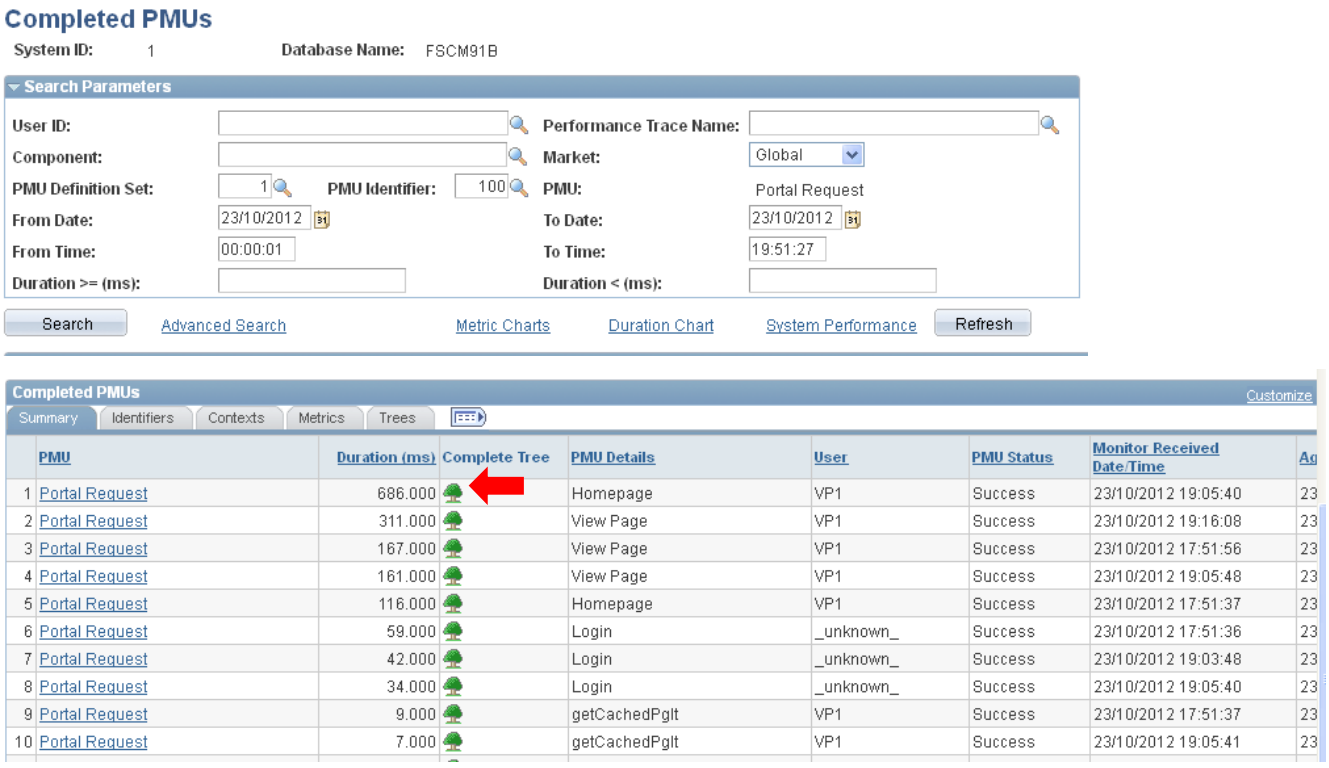
Each of these three are explained below.

Multiple Pagelets

Navigate to **PeopleTools, Performance Monitor, History, Completed PMUs,** and search for slow Portal Requests. Enter 100 in the PMU Identifier field and click the Search button. [Optional] Enter a duration threshold in the Duration >= (ms) field as search criteria. For example, a value of 10,000 will return any Portal request taking longer than 10 seconds.

Drill down to the Complete Tree with long running Portal requests where PMU Details = Homepage. Each of the GetContent PMUs reveals the pagelet load time.

Click on the GetContent PMU link to research which pagelet has been accessed.

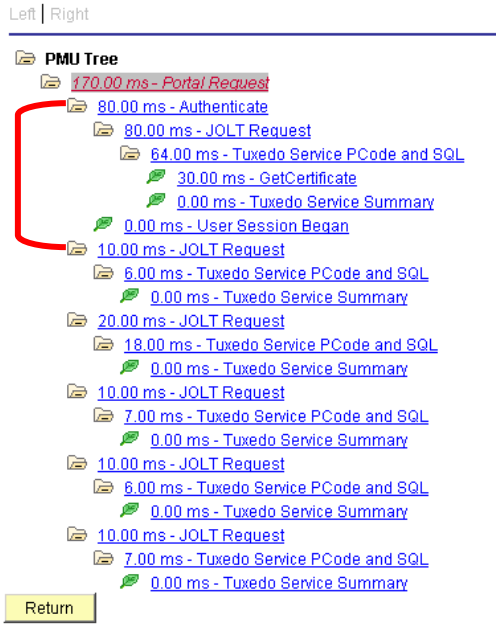


Slow User Authentication

Navigate to **PeopleTools, Performance Monitor, History, Completed PMUs,** and search for slow Portal Requests. Enter 101 in the PMU Identifier field and click the Search button. Drill down to the Complete Tree with long running Portal requests where PMU Detail = Login.

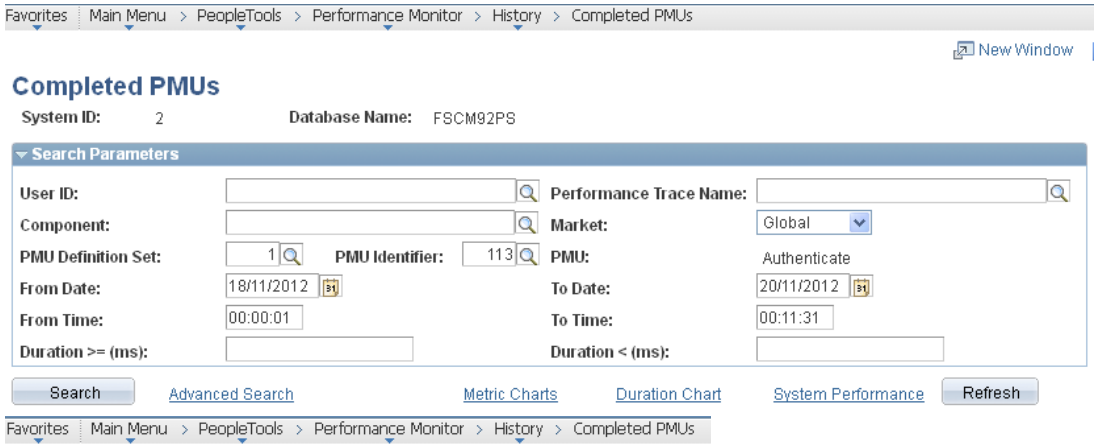
81	Portal Request	170.000	Login	_unknown_	Success
82	Portal Request	171.000	Login	_unknown_	Success
83	Portal Request	80.000	Login	_unknown_	Success
84	Portal Request	90.000	Login	_unknown_	Success
85	Portal Request	70.000	Login	_unknown_	Success
86	Portal Request	70.000	Login	_unknown_	Success
87	Portal Request	0.000	Login	_unknown_	Success
88	Portal Request	80.000	Login	_unknown_	Success

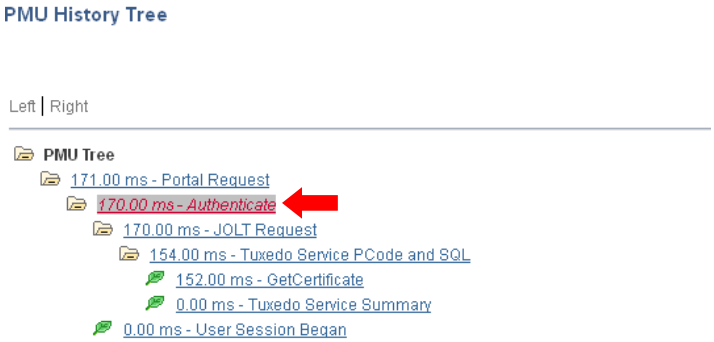
Inspect the time spent in Authenticate and GetCertificate.



Alternatively, you can enter PMU Identifier = 113 (Authenticate) or 413 (GetCertificate). Then drill down to the Completed Tree.

If LDAP (Lightweight Directory Access Protocol) is used, you can sample LDAP timing information (PMUs 422 and 423) by setting the application server Agent Filter Level to **05-Verbose**. Both PMUs 422 (LDAP bind) and 423 (LDAP search) should appear under PMU 413. Click on the first Portal Request (170.000)





Network Latency

Network latency between the user's browser client and the web server is captured and sampled during user sign on. To view latency statistics, navigate to

> *PeopleTools, Performance Monitor, System Monitor, Current User Sessions, or, PeopleTools, Performance Monitor, History, User Session History*

Click on the Details tab. The latency is displayed in milliseconds.

Alternatively, you can ask the end user experiencing the latency to run PeopleSoft Ping from their browser client. To activate PS Ping, navigate to

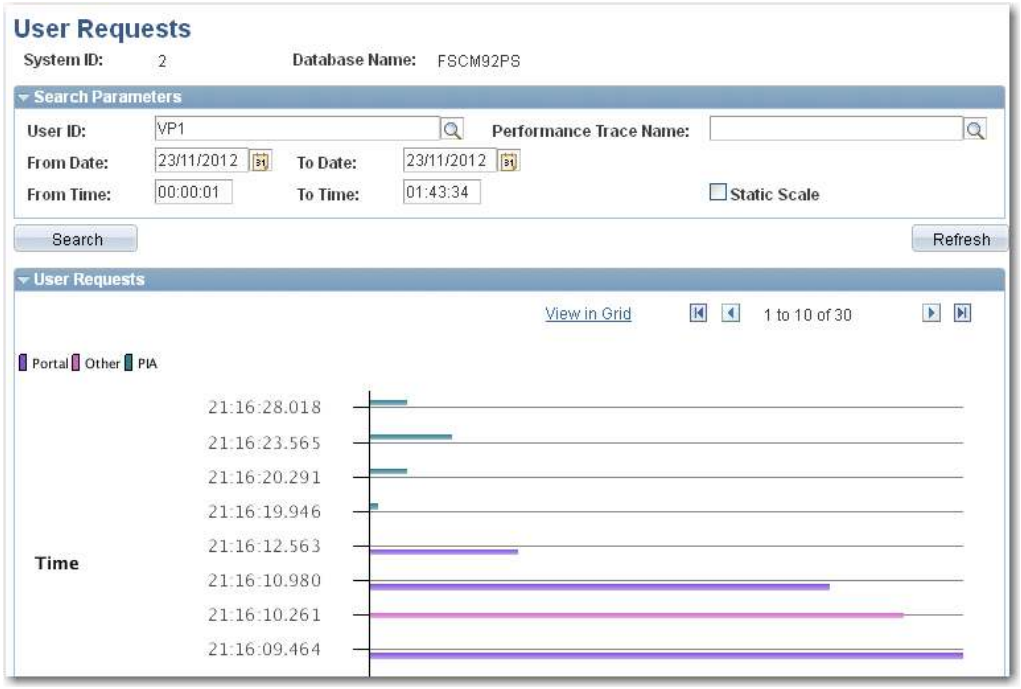
> *PeopleTools, Utilities, PeopleSoft Ping.*

LONG RUNNING USER AND PORTAL REQUESTS

When users complain about poor online performance, you can use Performance Monitor to verify that you have performance problems and diagnose their causes. There are two ways for searching for long running user requests, as explained below.

Known User ID, Time, and Monitored System

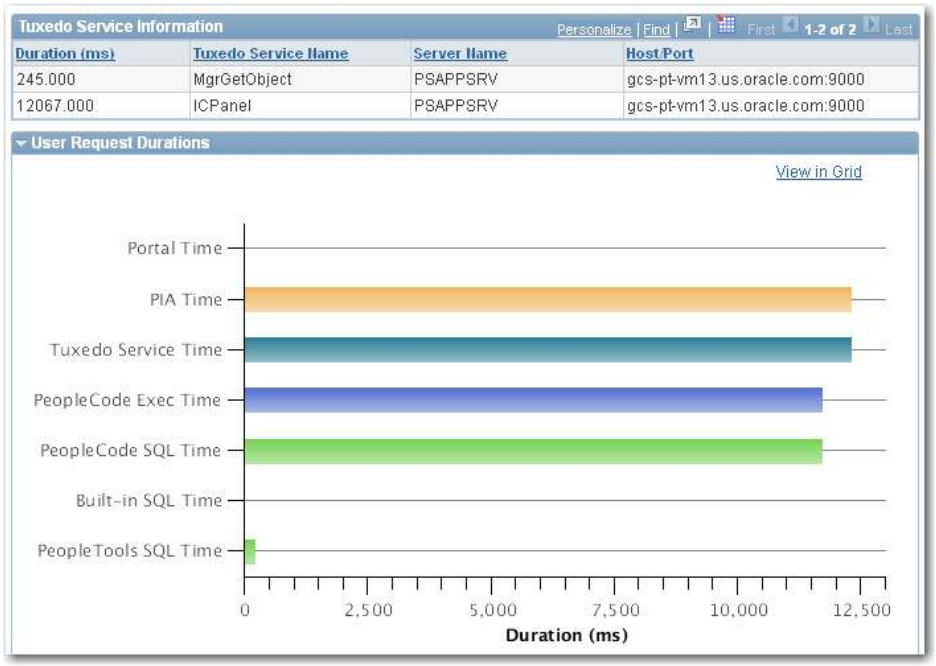
1. Navigate to **PeopleTools, Performance Monitor, Analytics, User Requests**. Select the monitored system. Enter the User ID and the approximate From/To Date and Time. Optionally, check the Static Scale checkbox.
2. Click the Search button.
3. Then navigate through the data and find the request(s) with a long duration.



4. Click on the bar to display the request details. Alternatively, you can click on the View in Grid link and sort the result by duration.

5. The User Request Details page provides a summary and breakdown of where the system spent the time. In the application server Totals group box, you can analyze how the timings are distributed.

6. Click on the IC Panel link to get the details on where the time has spent.



Here are some suggestions when interpreting this information:

High SQL Total Times suggest database tuning opportunities. If the PeopleCode SQL time shows a long duration time in the User Request Durations diagram, you should investigate in detail the SQL statements in the Component (look for the Name in the Component PMU group box). Built in SQL time and PeopleTools SQL time result from queries issued from PeopleTools binaries and can only be tuned by PeopleSoft.

High Serialization or Deserialization Times can be due to the amount of the information and/or complexity of the html page passed between the web server and the application server. If this is a customized Component, you might want to investigate if the Component contains too many items in the Component buffer. You can check the Component Buffer Size in the Component PMUs group box. Refer to the Chapter ***Referencing Data in the Component Buffer*** in the ***PeopleCode Developer's Guide PeopleBook*** for more details.

High Total Cache Misses, Cache File Hits, or Cache Memory hits – This time records waits caused by components were requested but the metadata was not found in the file or memory cache of the process.

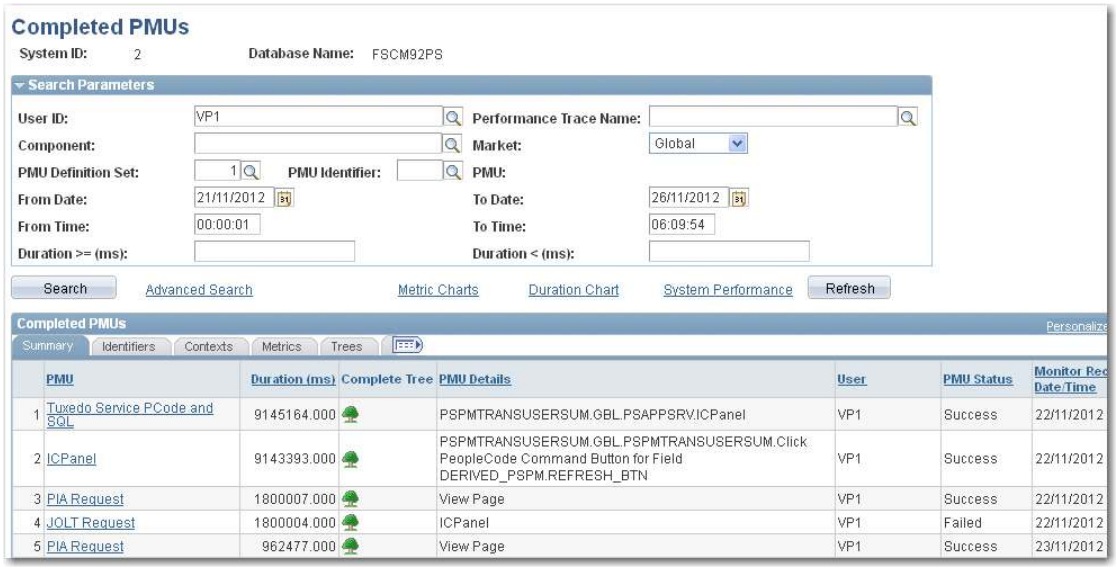
- **Cache Misses** occur when the particular instance of an application server process does not have the metadata required to process a user request in cache. The counter indicates the number of PeopleTools objects metadata that were retrieved from the database. You might see high PeopleTools SQL time as a result of this. Typically, cache misses happen when the application server domain is fresh and users run their transactions for the first time within that domain.
- **Cache File Hits** occur when the application server retrieves metadata from the application server cache file (under the <PS_HOME>/appserv/<domain>/CACHE directory) instead of going to the database. This should take less time than accessing the database if the file system is close to the application server box. You see Cache File Hits as user transactions being processed after a reboot of the application server domain. Cache Memory Hits occur when the application server retrieves the metadata from its memory, instead of the cache file or the database.

Note. *Persistent* high cache misses or file hits indicate possible cache problems that should be investigated.

Searching for Long Running Transactions

1. Navigate to

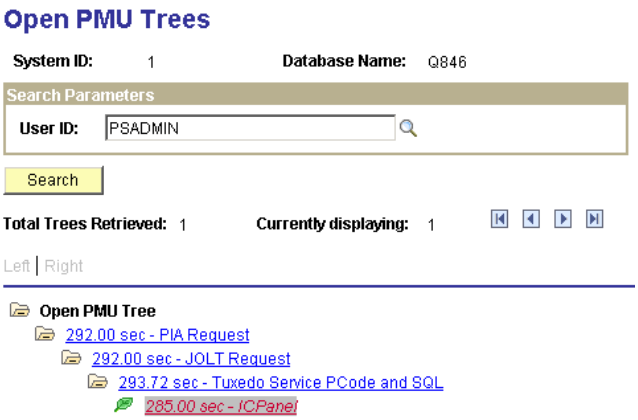
   PeopleTools, Performance Monitor, History, Completed PMUs.
2. Select the monitored system. Enter PMU Identifier = 100 (Portal request) or 101 (PIA requests), and a value in the Duration >= (ms): field. Additionally, the Advance Search page allows you to select a particular web server domain, Agent ID, etc.
3. Click on the Duration Chart and Metric Chart links to present the results in a chart.



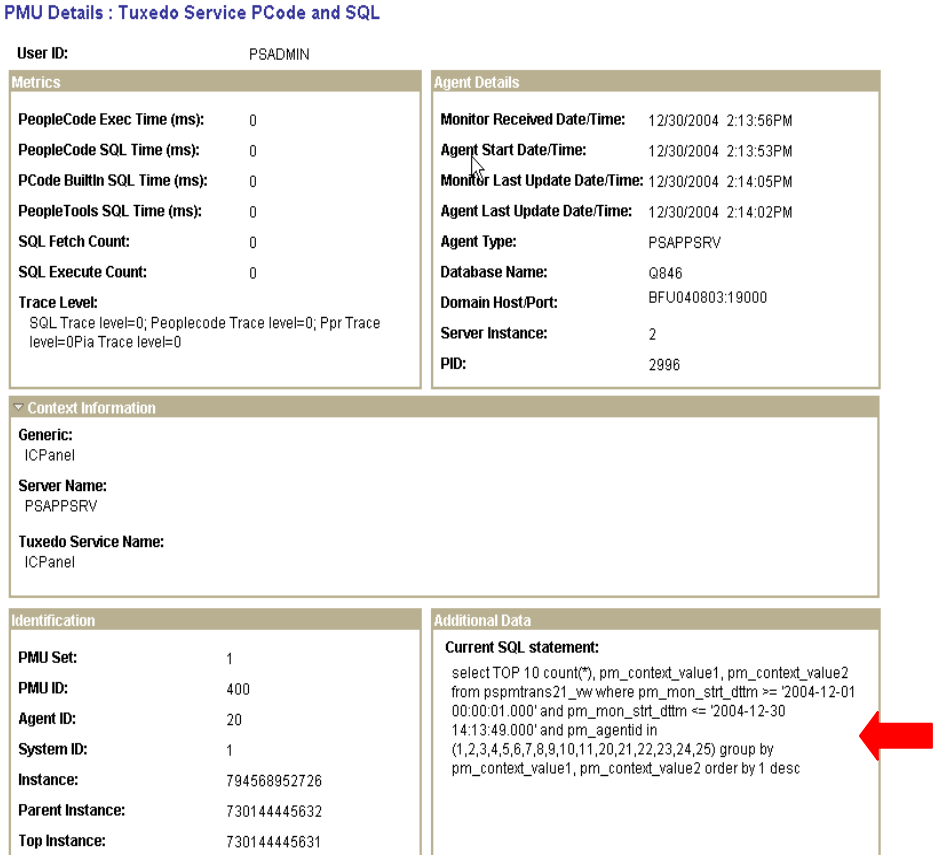
POORLY PERFORMING, ACTIVE TRANSACTIONS

When a user reports that a transaction has been running for a long time and it is still running, Performance Monitor might be able to catch the relevant performance data using the **04-Standard** Agent Filter level. Navigate to **PeopleTools, Performance Monitor, System Performance, Open PMU Trees** (if you know the User ID associated with the transaction and which monitored system is executing it) or Open PMUs (if you don't know the User ID). The PMU tree shows the elapsed time at each level. If the PMU occurs before 12AM today, then a date-time stamp is displayed instead.

Note: Some of the duration values in the middle of the PMU tree may be longer than others. This is because some PMUs could arrive at the PPMI Monitor Servlet earlier than other PMUs or the CPU clock might not be synchronized between servers. You only see the discrepancies during the life of the PMUs in the Open PMU Trees page. When the PMUs are completed, Performance Monitor computes the duration based on the agent timestamp and synchronizes the values across all tiers of servers.



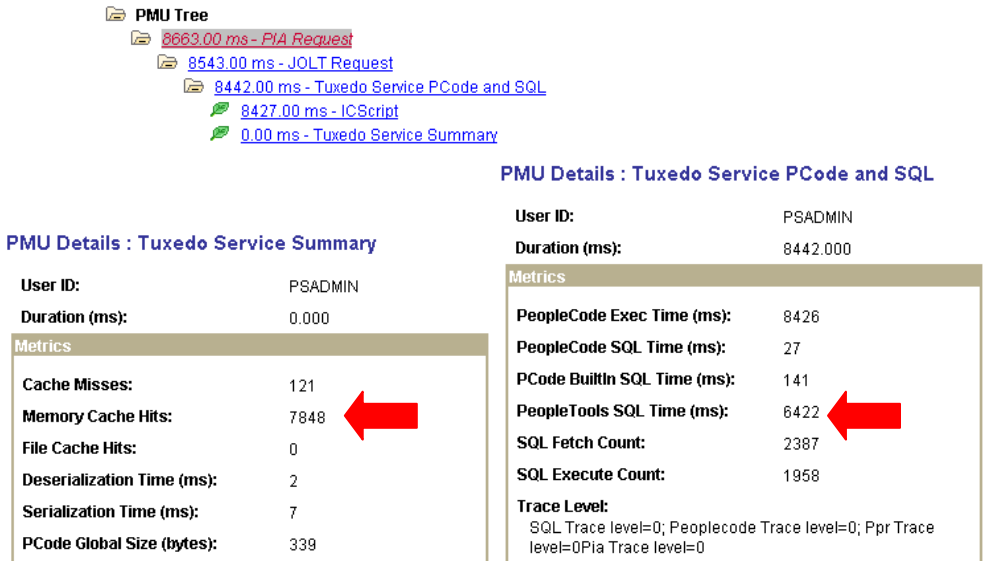
Click on the Tuxedo Service PCode and SQL link. The details page shows the Current SQL Statement.



Note: Normally, you need to use the **05-Verbose** Agent Filter Level to capture all of the SQL statements. In the Open PMU Trees, Performance Monitor captures current running SQL statements even if the Agent Filter Level is set to **04-Standard**.

Determining If a Delay is Due to Caching

Unless all user transactions are executed at least once on every instance of the application server processes, none of the application server processes cache the same PeopleTools object metadata. This means that when the first user submits a transaction that is new to an application server, the user will experience a delay due to the initial caching of the transaction metadata. Follow the Long Running User and Portal Requests section above to verify that the delay is due to caching. Another method is to examine the caching metrics in PMU 412 (Tuxedo Service Summary). A none-zero value in Cache Misses may attribute to the delay. Here is an example where caching takes up the majority of the total SQL time reflected in PeopleTools SQL Time.



Navigate to:

PeopleTools > Performance Monitor > System Performance > Open PMU Trees – PMU Details

Alternatively, navigate to:

PeopleTools > Performance Monitor > History, Completed PMUs,

and Search with the PMU Identifier = 412. Then review Caching statistics in the Metric Charts or in the grid. Optionally, select or enter the Component name that you want to investigate.

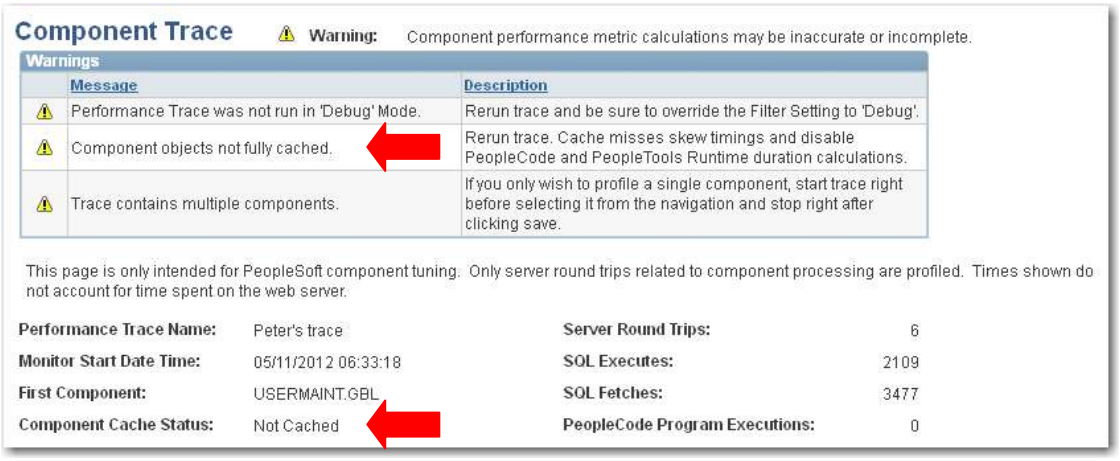
CAPTURING A COMPONENT TRACE

If you are using the Component Trace page to analyze a Performance Trace, make sure to run it with a minimum of the **05-Verbose** Agent Filter Level. The SQL Summary section contains the result only if a Performance Trace is captured with **05-Verbose** level (or higher). The PeopleCode Summary section requires a Performance Trace captured at the **06-Debug** Agent Filter Level.

The Component Trace page is most effective if all the transactions are coming from a single Component. If the data comes from more than one Component, the Component caching statistics will be misleading.

Check the Component Cache Status before coming to any conclusions. It is easier to analyze the data if the status is Cached.

The Duration Summary section displays how the time spent in the system is distributed between SQL, pack/unpack (i.e., serialization/deserialization), PeopleCode runtime, and PeopleTools infrastructure. Investigate further if you see that a large percentage of time is being spent on SQL or PeopleCode.



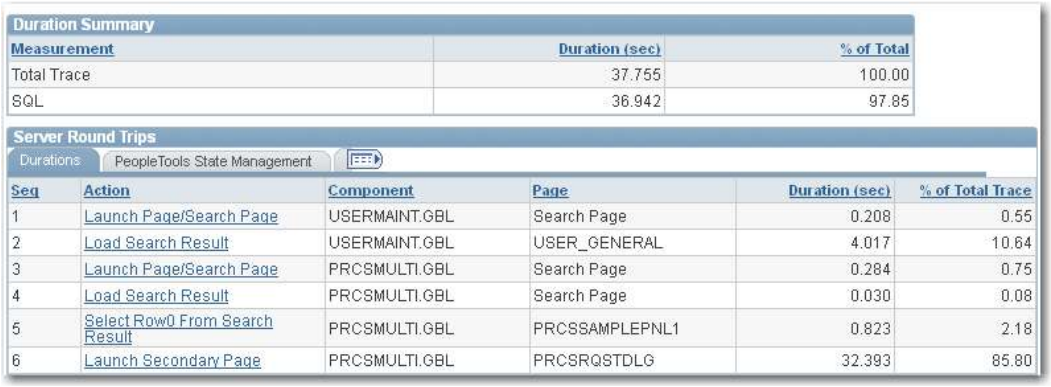
Component Trace ⚠ **Warning:** Component performance metric calculations may be inaccurate or incomplete.

Warnings

	Message	Description
⚠	Performance Trace was not run in 'Debug' Mode.	Rerun trace and be sure to override the Filter Setting to 'Debug'.
⚠	Component objects not fully cached.	Rerun trace. Cache misses skew timings and disable PeopleCode and PeopleTools Runtime duration calculations.
⚠	Trace contains multiple components.	If you only wish to profile a single component, start trace right before selecting it from the navigation and stop right after clicking save.

This page is only intended for PeopleSoft component tuning. Only server round trips related to component processing are profiled. Times shown do not account for time spent on the web server.

Performance Trace Name:	Peter's trace	**Server Round Trips:**	6
Monitor Start Date Time:	05/11/2012 06:33:18	**SQL Executes:**	2109
First Component:	USERMAINT.GBL	**SQL Fetches:**	3477
Component Cache Status:	Not Cached	**PeopleCode Program Executions:**	0



Duration Summary

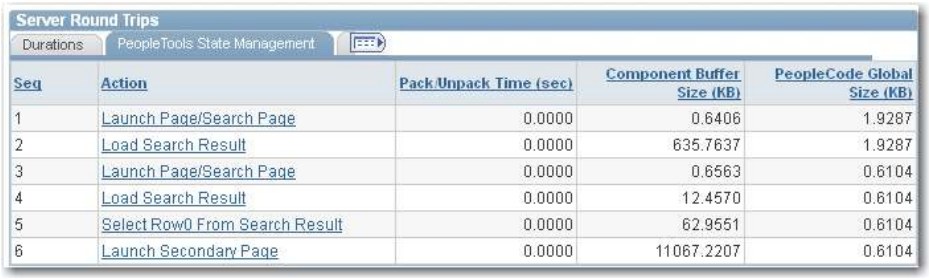
Measurement	Duration (sec)	% of Total
Total Trace	37.755	100.00
SQL	36.942	97.85

Server Round Trips

Durations | PeopleTools State Management

Seq	Action	Component	Page	Duration (sec)	% of Total Trace
1	Launch Page/Search Page	USERMAINT.GBL	Search Page	0.208	0.55
2	Load Search Result	USERMAINT.GBL	USER_GENERAL	4.017	10.64
3	Launch Page/Search Page	PRCSMULTI.GBL	Search Page	0.284	0.75
4	Load Search Result	PRCSMULTI.GBL	Search Page	0.030	0.08
5	Select Row0 From Search Result	PRCSMULTI.GBL	PRCSSAMPLEPNL1	0.823	2.18
6	Launch Secondary Page	PRCSMULTI.GBL	PRCSRQSTDLG	32.393	85.80

Click on the PeopleTools State Management tab to display the serialization time, deserialization time, Component Buffer size and the size of PeopleCode global.

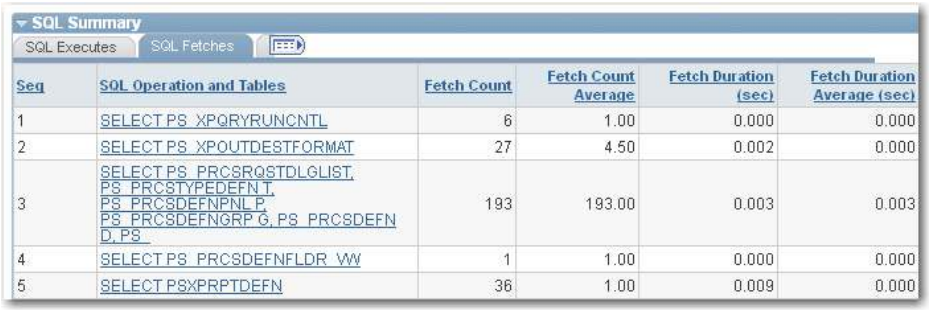


Server Round Trips

Durations | PeopleTools State Management

Seq	Action	Pack/Unpack Time (sec)	Component Buffer Size (KB)	PeopleCode Global Size (KB)
1	Launch Page/Search Page	0.0000	0.6406	1.9287
2	Load Search Result	0.0000	635.7637	1.9287
3	Launch Page/Search Page	0.0000	0.6563	0.6104
4	Load Search Result	0.0000	12.4570	0.6104
5	Select Row0 From Search Result	0.0000	62.9551	0.6104
6	Launch Secondary Page	0.0000	11067.2207	0.6104

The server round trips is the total number of occurrences during the trace. The sequence of the trips and their durations appear in the Server Round Trips section. Click on the Action link to display a summary of all SQL statements during the server trip.



SQL Summary

SQL Executes | SQL Fetches

Seq	SQL Operation and Tables	Fetch Count	Fetch Count Average	Fetch Duration (sec)	Fetch Duration Average (sec)
1	SELECT PS_XPQRYRUNCNTL	6	1.00	0.000	0.000
2	SELECT PS_XPOUTDESTFORMAT	27	4.50	0.002	0.000
3	SELECT PS_PRCSRQSTDLGLIST, PS_PRCSTYPEDEFN T, PS_PRCSDEFNPNL P, PS_PRCSDEFNGRP G, PS_PRCSDEFN D, PS_	193	193.00	0.003	0.003
4	SELECT PS_PRCSDEFNFLDR_VW	1	1.00	0.000	0.000
5	SELECT PSXPRPTDEFN	36	1.00	0.009	0.000

LONG RUNNING SQL

There are three PMUs that report SQL statistics. All three PMUs will have their performance data monitored at the **05-Verbose** Agent Filter Level.

- 406-PeopleCode SQL Execute
- 407-PeopleCode Built-in SQL Execute
- 408-PeopleTools SQL Execute

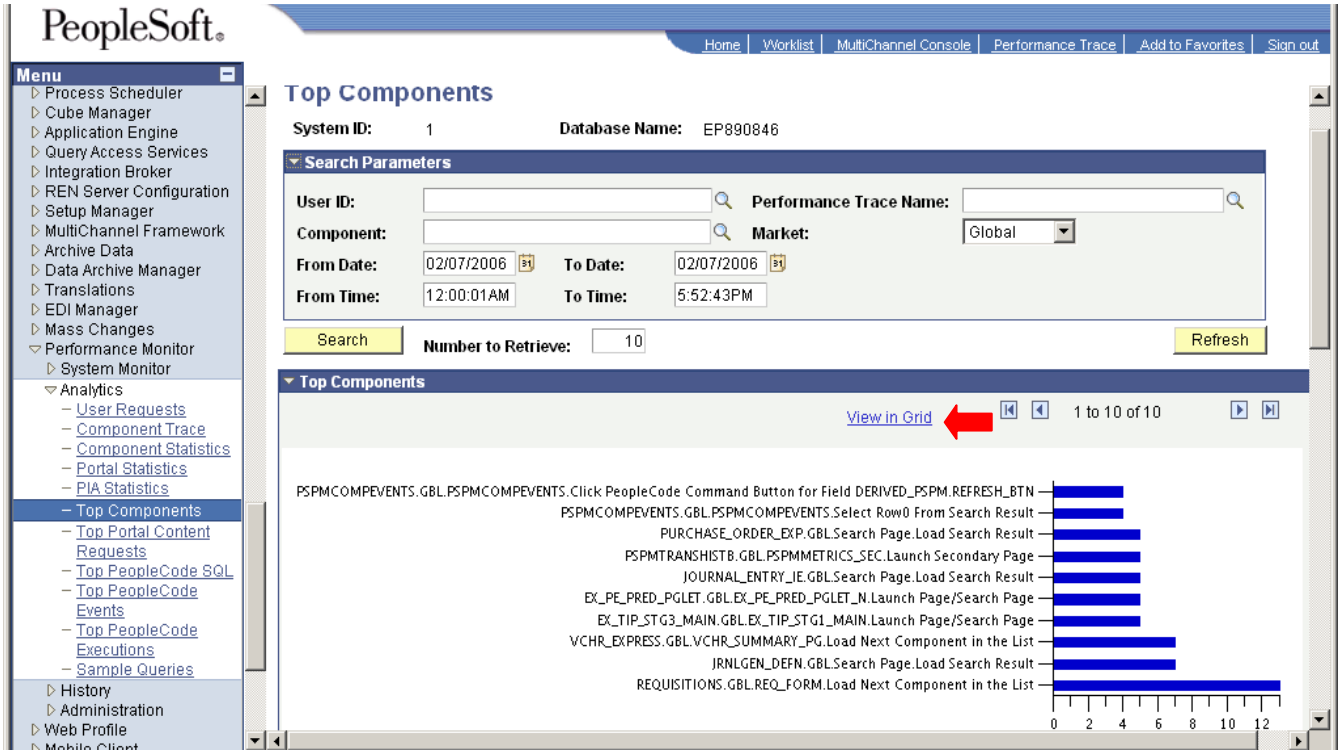
Navigate to **PeopleTools, Performance Monitor, History, Completed PMUs**. Select the monitored system. Enter PMU identifier = 406, and a value in the Duration >= (ms): field. Check to see if the SQL statement uses a cursor, and if it has been compiled.

TOP COMPONENT USAGE

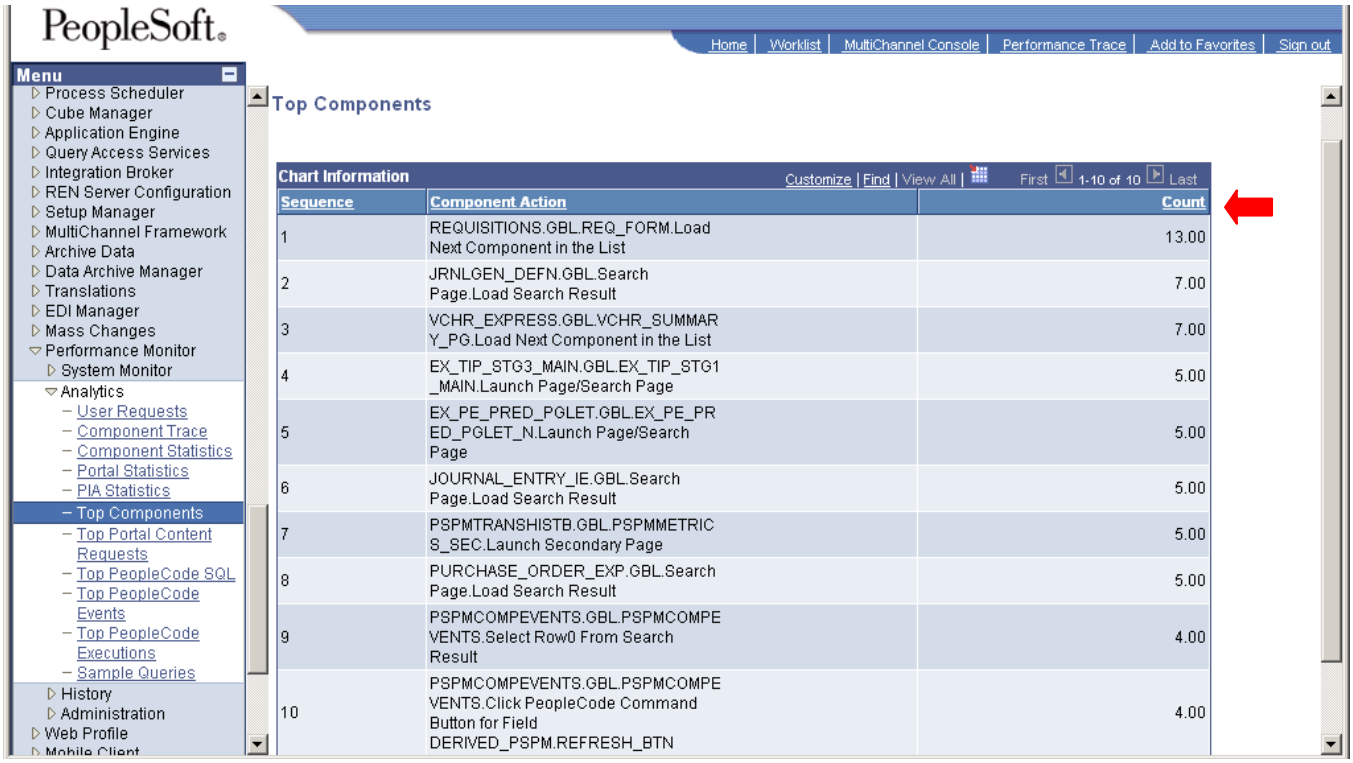
Navigate to **PeopleTools, Performance Monitor, Analytics, Top Components**. Select the monitored system. Enter a Date / Time range and the number of Top Components you want to see. There are three sections / charts on this page:

- A display of Top Components by the number of times visited (based on count).
- A display of Top Components by cumulative time spent in the Component (based on cumulative durations).
- A display of Top Components by average time spent in the Component (based on duration averages).

This is the main screen for getting the Top Components displayed depending on your selection.

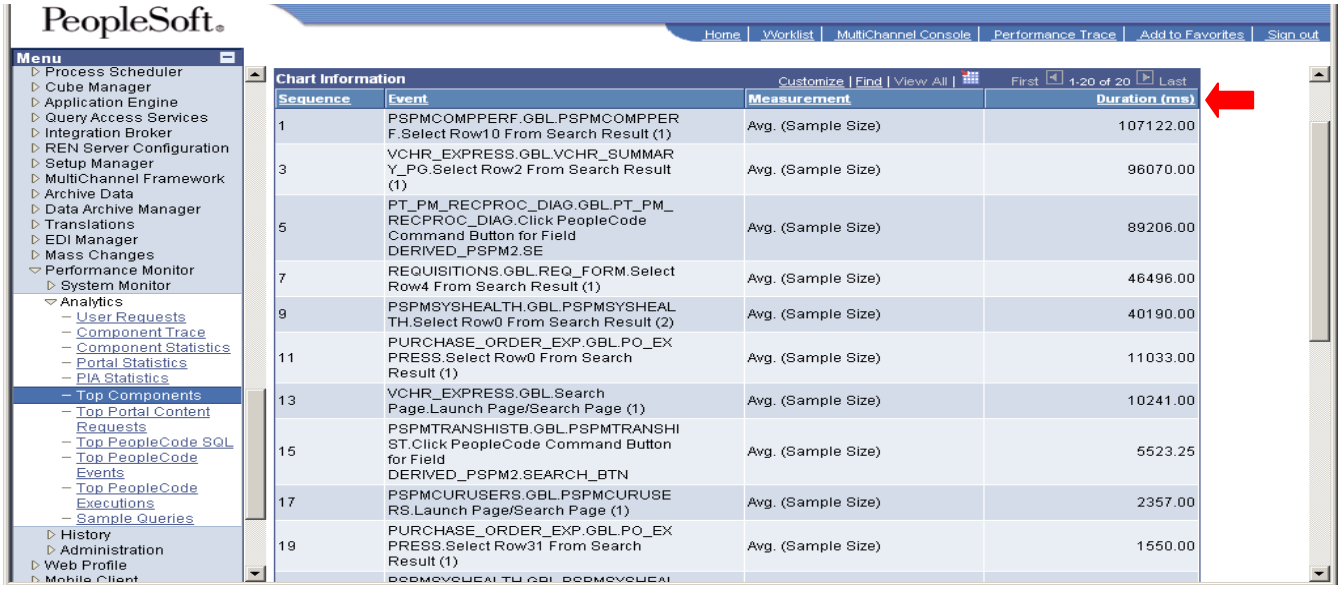


Click on the View in Grid link to display the actual data, and sort it by clicking on the column headers

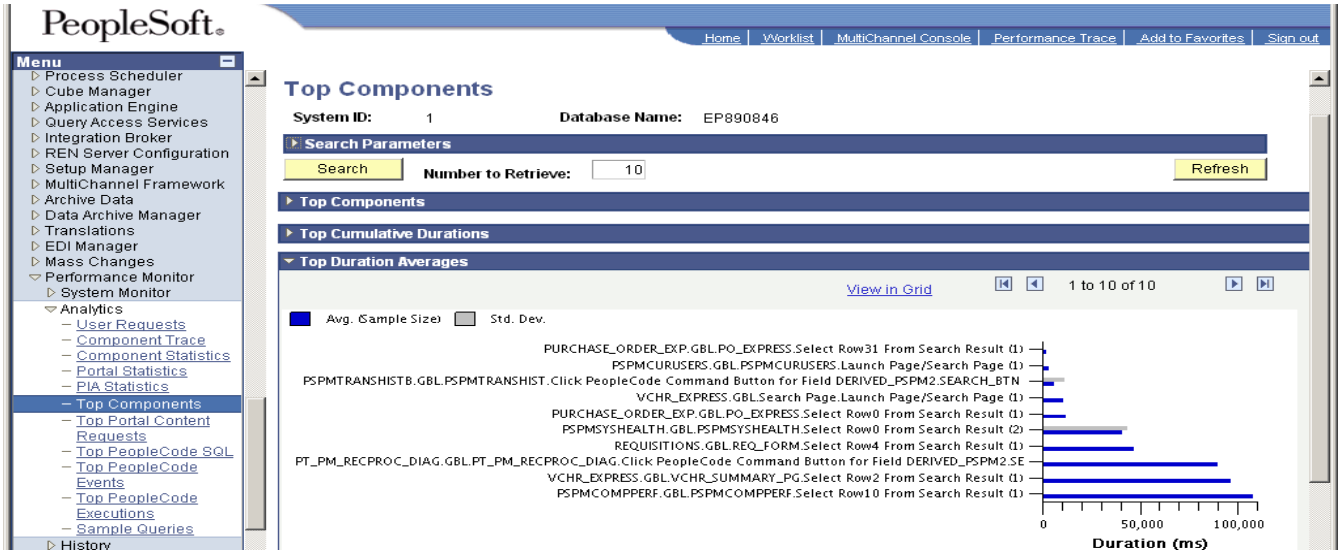


By expanding the section 'Top Cumulative Durations' you will get the Top Components sorted by durations (ms)

Click on the View in Grid link to display the actual data, and sort it by clicking on the column headers



By expanding the section 'Top Durations Averages' you will see the average time for loading each of the components (ms)



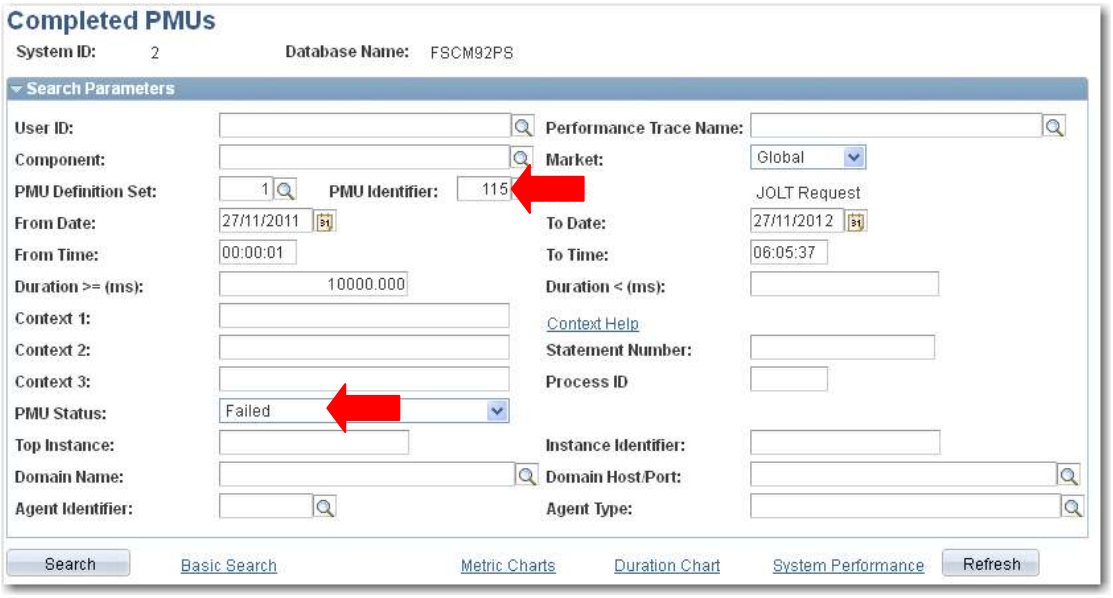
JOLT ERROR AND TIMEOUT

Web browsers and integrated systems don't send requests directly to the application server. Instead, they send HTTP/S requests to the PeopleSoft servlets running on the web server. The web server translates the HTTP/S request into a Jolt request that is sent to a specified Jolt port. Then the application server, running on Tuxedo, submits the appropriate SQL to the database.

If there is a communication problem (request are taking too long) you will see various Jolt errors.

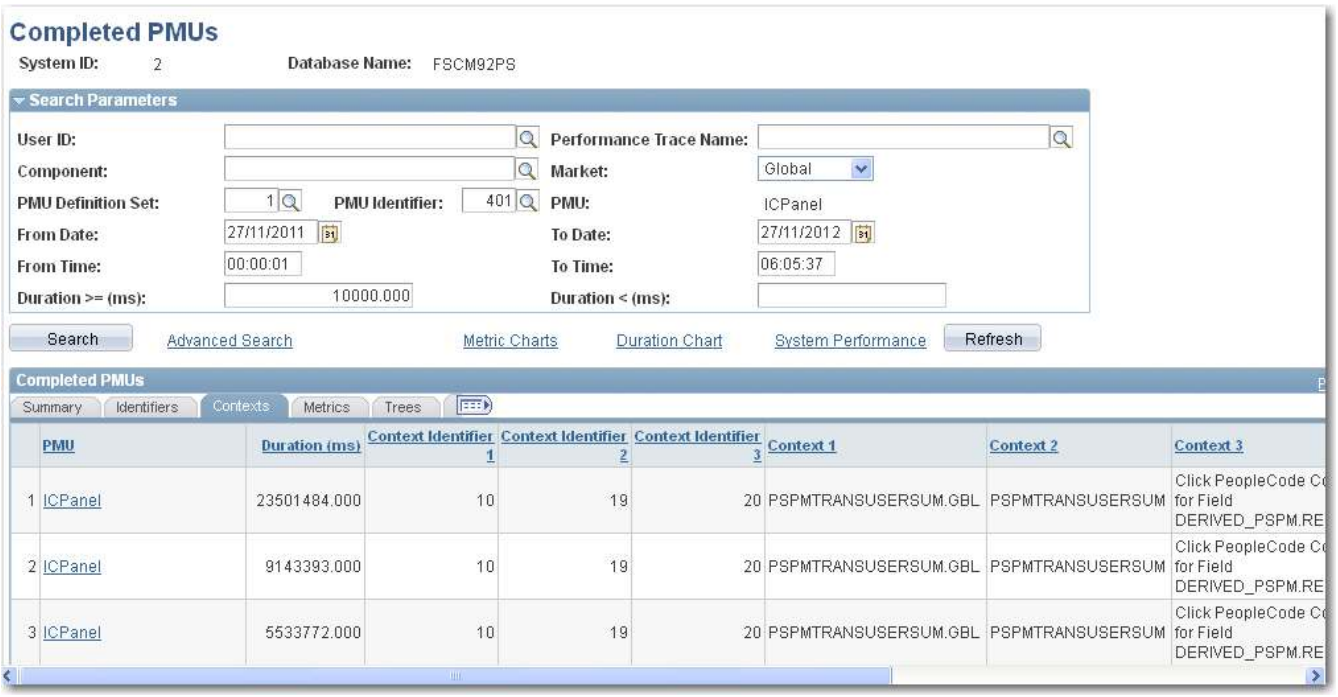
The difference in the duration time of PMU 115 (JOLT Request) and its corresponding PMU 400 (Tuxedo Service PCode and SQL) is the total time spent on JOLT network latency, waiting time in the Tuxedo queue, serialization and deserialization of the full request/reply at the web server, and the serialization and deserialization of the User ID information on the application server

Navigate to **PeopleTools, Performance Monitor, History, Completed PMUs**. Select the monitored system. Enter PMU identifier = 115 (JOLT Request) and PMU Status = Failed or Timeout on the Advanced Search Page.



SPECIFIC COMPONENT NAME / USER ACTIONS

The User action is reported in the Context 3 field for PMUs 401, 402, 403, and 416, and in the Context 2 field for PMU 411. The Component name is reported in the Context 1 field for PMUs 401, 402, 403, and 416. These values are shown in the Contexts tab on the Completed PMUs page.



EXTERNAL SOURCE CALLING COMPONENT INTERFACE

Here is an example of tracking Web Services performance, such as SOAP (Simple Object Access Protocol) or WSDL (Web Services Description Language) interactions between systems.

Note that the PMU details show that a SOAP web service request has been sent to the Integration Broker and is, in turn, making a call to a component interface executing business logic. PMU 418, which is an External Component Interface Call into the application server, captures the performance statistics. PMU 418 is monitored at the **05-Verbose** Agent Filter Level.

Open PMUs — Customize | Find | View All | First 1-29 of 29 Last

Summary | Identifiers | Contexts | Metrics

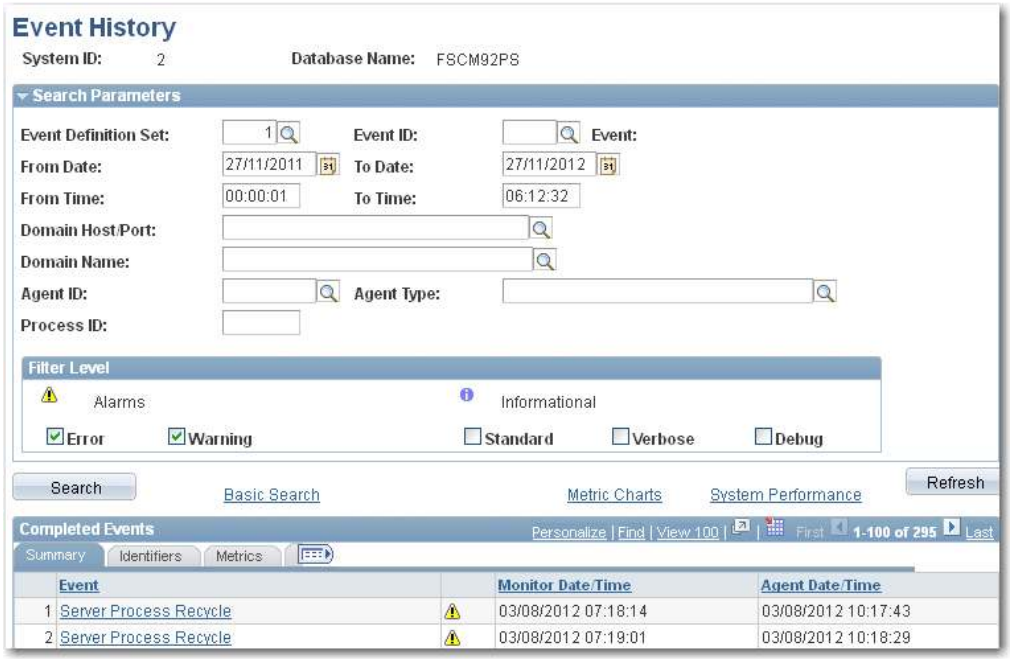
	PMU	Duration (sec)	PMU Details	User ID	Action	Monitor Received Date/Time	Agent Start Date/Time
1	Tuxedo Service PCode and SQL	5931.733	ICScript.ICScript	_unknown_	Update	06/21/2004 10:18:45AM	06/21/2004 9:46:17AM
2	ICScript	5914.000	http://psft-web.combe.com:10380/psc/fs88tst/EMPLOYEE/PT_LOCAL/s/WEBLIB_SOAPTOCI.SOAPTOCI.FieldFormula.IScript_SOAPToCI	_unknown_	Start	06/21/2004 10:19:03AM	06/21/2004 9:46:17AM
3	Tuxedo Service PCode and SQL	5455.847	ICScript.ICScript	_unknown_	Update	06/21/2004 10:26:41AM	06/21/2004 10:26:40AM
4	ICScript	5446.000	http://psft-web.combe.com:10380/psc/fs88tst/EMPLOYEE/PT_LOCAL/s/WEBLIB_SOAPTOCI.SOAPTOCI.FieldFormula.IScript_SOAPToCI	_unknown_	Start	06/21/2004 10:26:51AM	06/21/2004 10:26:40AM
5	Tuxedo Service PCode and SQL	4849.517	ICScript.ICScript	_unknown_	Update	06/21/2004 10:36:47AM	06/21/2004 10:36:44AM
6	ICScript	4849.000	http://psft-web.combe.com:10380/psc/fs88tst/EMPLOYEE/PT_LOCAL/s/WEBLIB_SOAPTOCI.SOAPTOCI.FieldFormula.IScript_SOAPToCI	_unknown_	Start	06/21/2004 10:36:48AM	06/21/2004 10:36:44AM
7	Tuxedo Service PCode and SQL	3831.483	ICScript.ICScript	_unknown_	Update	06/21/2004 10:53:45AM	06/21/2004 10:53:41AM

Done

ALARMS AND WARNINGS

Events 355, 356, 500, 801, 802, 803 and 900 are monitored at the **03-Warning** Agent Filter Level. Navigate to **PeopleTools, Performance Monitor, History, Event History**. Click the Advanced Search link and check the Warning checkbox.

Here is an example of a JOLT service failure and the subsequent loss of transactional data.



Event History

System ID: 2 Database Name: FSCM92PS

Search Parameters

Event Definition Set: 1 Event ID: Event:

From Date: 27/11/2011 To Date: 27/11/2012

From Time: 00:00:01 To Time: 06:12:32

Domain Host/Port:

Domain Name:

Agent ID: Agent Type:

Process ID:

Filter Level

Alarms Informational

Error Warning Standard Verbose Debug

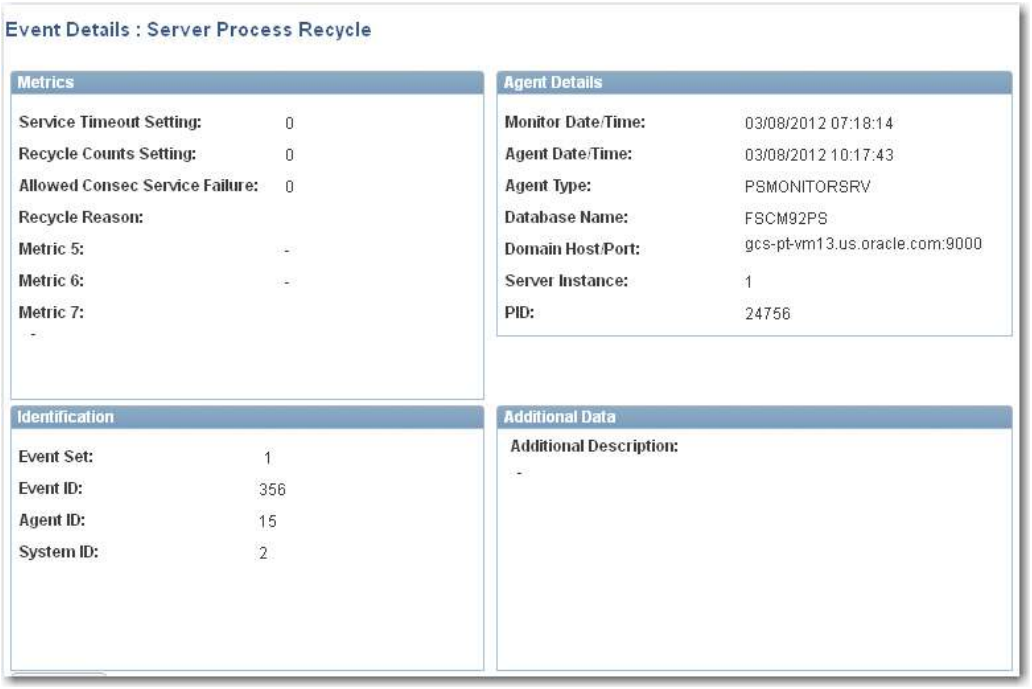
Search Basic Search Metric Charts System Performance Refresh

Completed Events — Personalize | Find | View 100 | First 1-100 of 295 Last

Summary | Identifiers | Metrics

	Event		Monitor Date/Time	Agent Date/Time
1	Server Process Recycle	⚠	03/08/2012 07:18:14	03/08/2012 10:17:43
2	Server Process Recycle	⚠	03/08/2012 07:19:01	03/08/2012 10:18:29

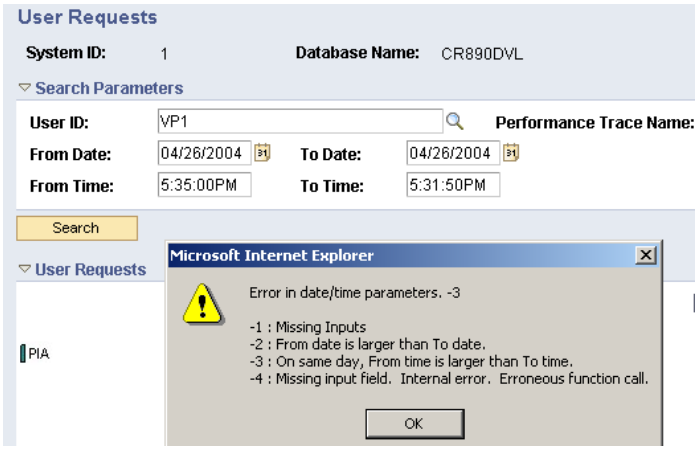
The following screen is showing a recycle of the PSMONITORSRV process on the Application Server domain:



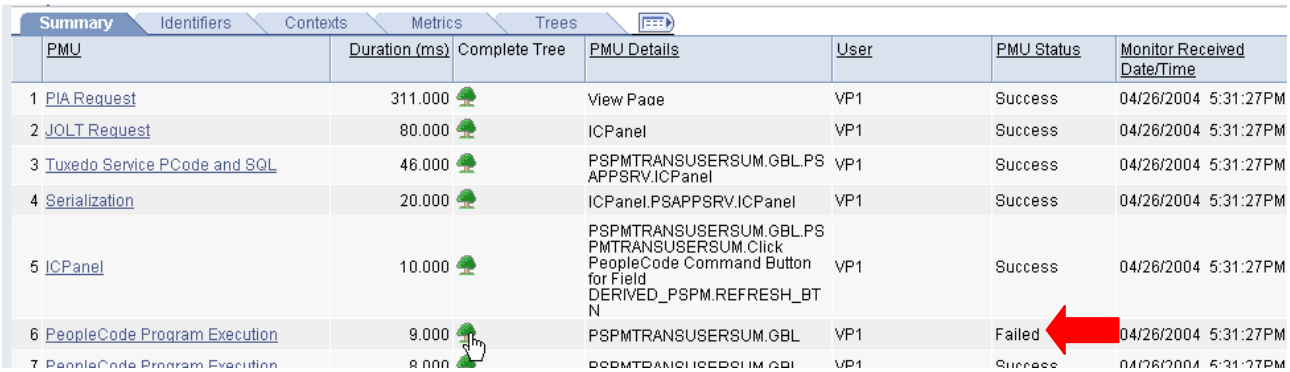
PMUs Containing PeopleCode Errors

If a user reports a PeopleCode error, navigate to the Completed PMUs page and search for PMU 404 (PeopleCode Program Execution) with PMU Status = Failed.

Note that monitoring PMU 404 requires the **06-Debug** Agent Filter Level. Drill down to the PMU tree to investigate the issue further.



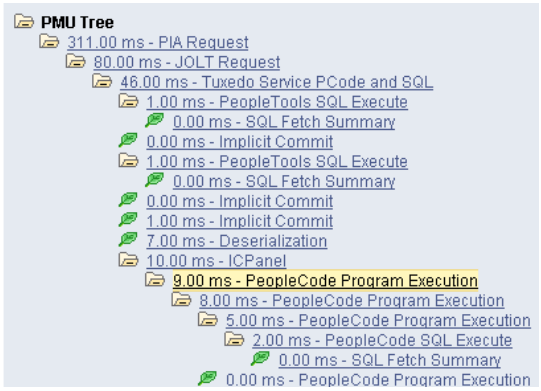
The following screen is showing an issue with the PeopleCode Program execution.



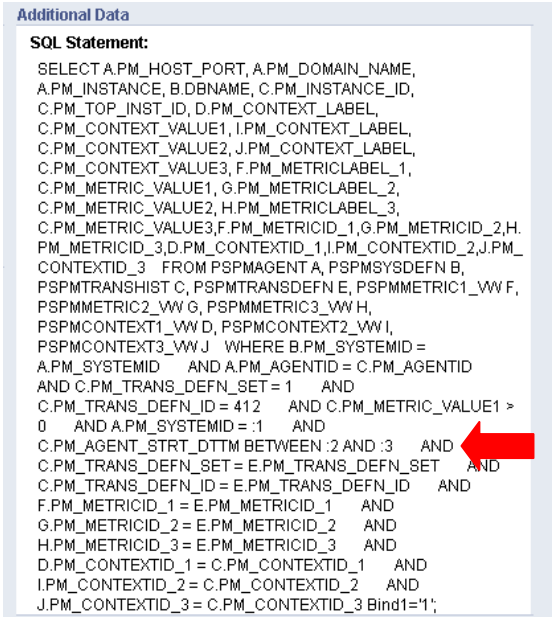
Summary | Identifiers | Contexts | Metrics | Trees

	PMU	Duration (ms)	Complete Tree	PMU Details	User	PMU Status	Monitor Received Date/Time
1	PIA Request	311.000		View Page	VP1	Success	04/26/2004 5:31:27PM
2	JOLT Request	80.000		ICPanel	VP1	Success	04/26/2004 5:31:27PM
3	Tuxedo Service PCode and SQL	46.000		PSPMTRANSUSERSUM.GBL.PSAPPSRV.ICPanel	VP1	Success	04/26/2004 5:31:27PM
4	Serialization	20.000		ICPanel.PSAPPSRV.ICPanel	VP1	Success	04/26/2004 5:31:27PM
5	ICPanel	10.000		PSPMTRANSUSERSUM.GBL.PSPMTRANSUSERSUM.Click PeopleCode Command Button for Field DERIVED_PSPM.REFRESH_BTN	VP1	Success	04/26/2004 5:31:27PM
6	PeopleCode Program Execution	9.000		PSPMTRANSUSERSUM.GBL	VP1	Failed	04/26/2004 5:31:27PM

Further investigation shows that the SQL statement fails due to the lack of date/time (bind variables :2 and :3).



PMU Tree
- 311.00 ms - PIA Request
  - 80.00 ms - JOLT Request
    - 46.00 ms - Tuxedo Service PCode and SQL
      - 1.00 ms - PeopleTools SQL Execute
        - 0.00 ms - SQL Fetch Summary
      - 0.00 ms - Implicit Commit
      - 1.00 ms - PeopleTools SQL Execute
        - 0.00 ms - SQL Fetch Summary
      - 0.00 ms - Implicit Commit
      - 1.00 ms - Implicit Commit
      - 7.00 ms - Deserialization
      - 10.00 ms - ICPanel
        - 9.00 ms - PeopleCode Program Execution
          - 8.00 ms - PeopleCode Program Execution
            - 5.00 ms - PeopleCode Program Execution
              - 2.00 ms - PeopleCode SQL Execute
                - 0.00 ms - SQL Fetch Summary
            - 0.00 ms - PeopleCode Program Execution



Additional Data

SQL Statement:

```
SELECT A.PM_HOST_PORT, A.PM_DOMAIN_NAME,
A.PM_INSTANCE, B.DBNAME, C.PM_INSTANCE_ID,
C.PM_TOP_INST_ID, D.PM_CONTEXT_LABEL,
C.PM_CONTEXT_VALUE1, I.PM_CONTEXT_LABEL,
C.PM_CONTEXT_VALUE2, J.PM_CONTEXT_LABEL,
C.PM_CONTEXT_VALUE3, F.PM_METRICLABEL_1,
C.PM_METRIC_VALUE1, G.PM_METRICLABEL_2,
C.PM_METRIC_VALUE2, H.PM_METRICLABEL_3,
C.PM_METRIC_VALUE3,F.PM_METRICID_1,G.PM_METRICID_2,H.
PM_METRICID_3,D.PM_CONTEXTID_1,I.PM_CONTEXTID_2,J.PM_
CONTEXTID_3  FROM PSPMAGENT A, PSPMSYSDEFN B,
PSPMTRANSHIST C, PSPMTRANSDEFN E, PSPMMETRIC1_VW F,
PSPMMETRIC2_VW G, PSPMMETRIC3_VW H,
PSPMCONTEXT1_VW D, PSPMCONTEXT2_VW I,
PSPMCONTEXT3_VW J  WHERE B.PM_SYSTEMID =
A.PM_SYSTEMID  AND A.PM_AGENTID = C.PM_AGENTID
AND C.PM_TRANS_DEFN_SET = 1  AND
C.PM_TRANS_DEFN_ID = 412  AND C.PM_METRIC_VALUE1 >
0  AND A.PM_SYSTEMID = :1  AND
C.PM_AGENT_STRT_DTTM BETWEEN :2 AND :3  AND
C.PM_TRANS_DEFN_SET = E.PM_TRANS_DEFN_SET  AND
C.PM_TRANS_DEFN_ID = E.PM_TRANS_DEFN_ID  AND
F.PM_METRICID_1 = E.PM_METRICID_1  AND
G.PM_METRICID_2 = E.PM_METRICID_2  AND
H.PM_METRICID_3 = E.PM_METRICID_3  AND
D.PM_CONTEXTID_1 = C.PM_CONTEXTID_1  AND
I.PM_CONTEXTID_2 = C.PM_CONTEXTID_2  AND
J.PM_CONTEXTID_3 = C.PM_CONTEXTID_3 Bind1='1';
```

SAMPLE QUERIES

In addition to the analytics menu items, Performance Monitor provides sample queries for listing Component Cache Misses, Timed Out SQL statements, and Server Restart Counts. All three queries can be run against the history table or archive table. You can access Sample Queries under

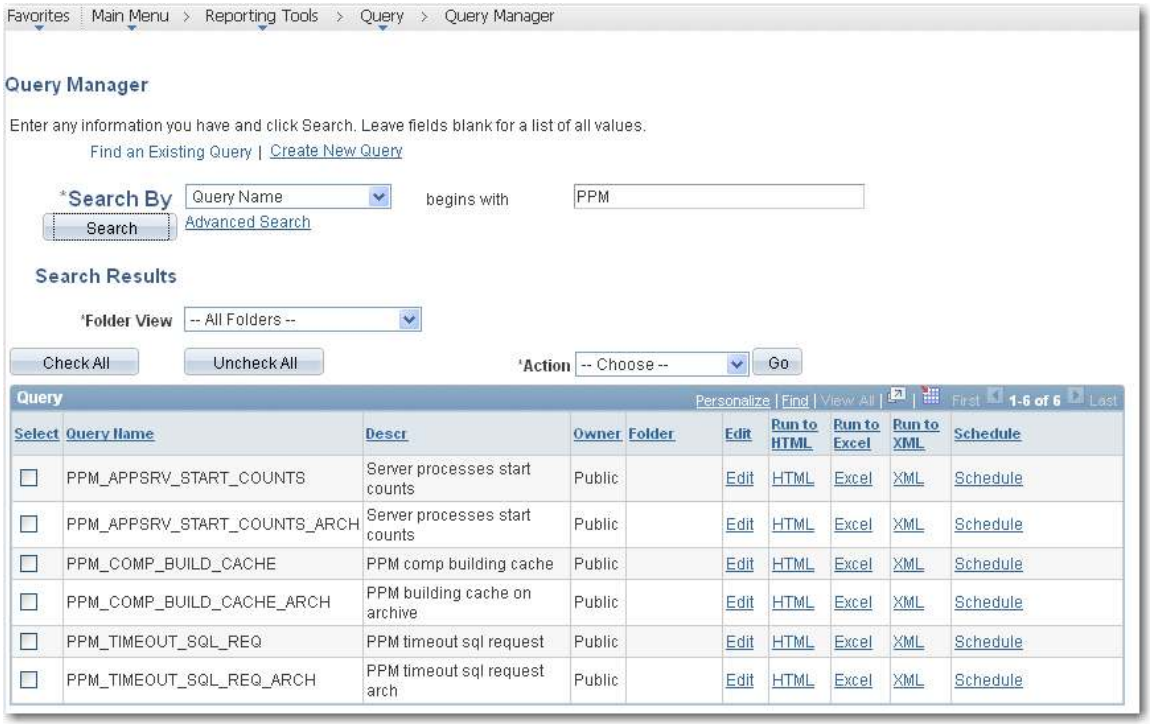
PeopleTools, Performance Monitor, Analytics, Sample Queries.

The date/time format is "MM/DD/yyyy hh[:mm:ss(A/P)M]" or "MM/DD/yyyy hh[:mm:ss]" if you are using 24-hour clock, where month (MM) and date (DD)are numeric.

For a list of sample queries available in PeopleTools, navigate to

Reporting Tools, Query, Query Manager.

Search by Query Name begins with PPM.

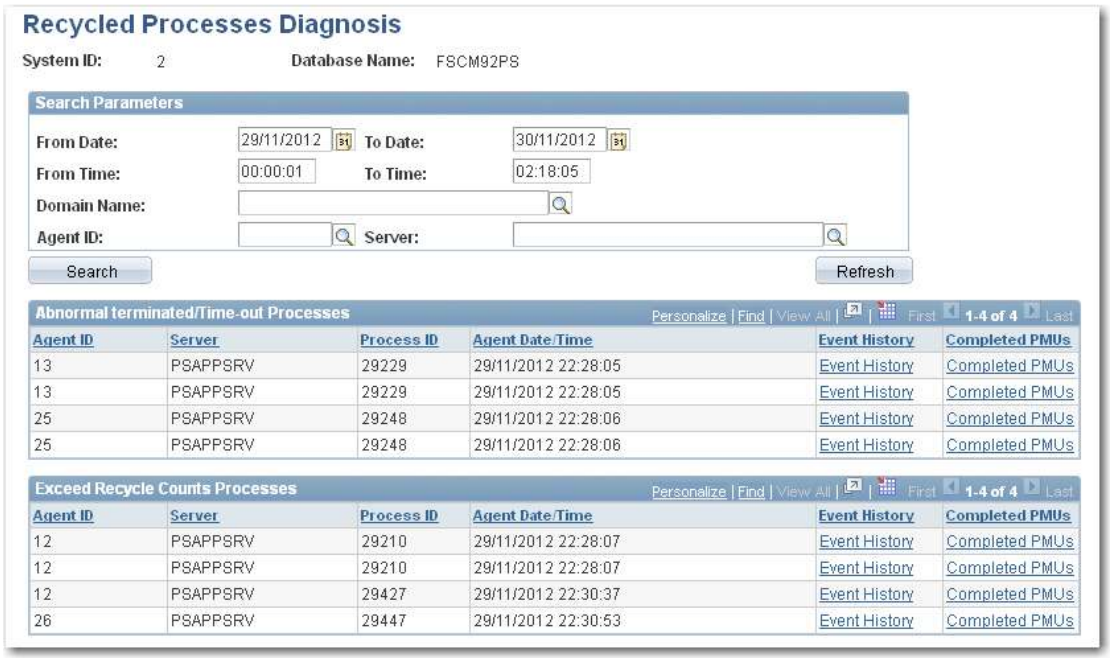


SERVER PROCESSES RECYCLING EVENTS

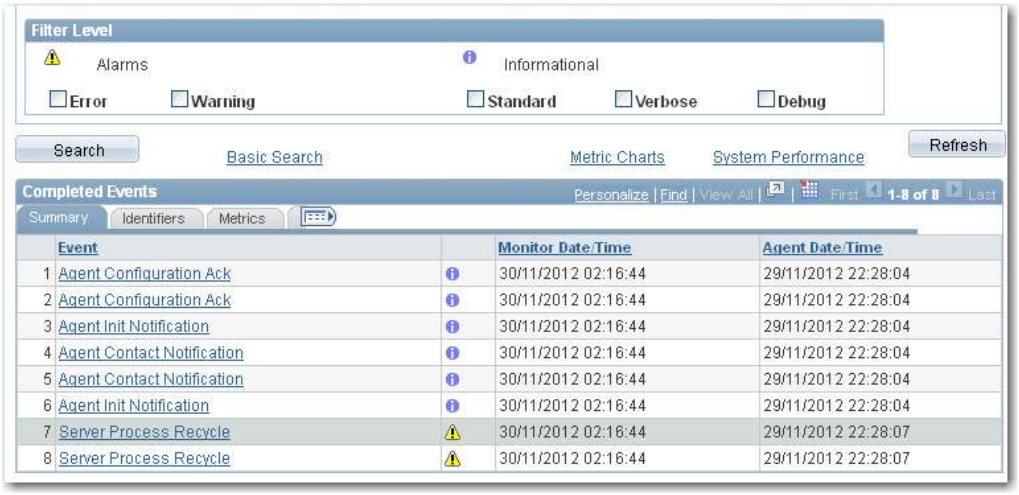
Navigate to

PeopleTools, Performance Monitor, System Performance, Recycled Processes Diagnosis.

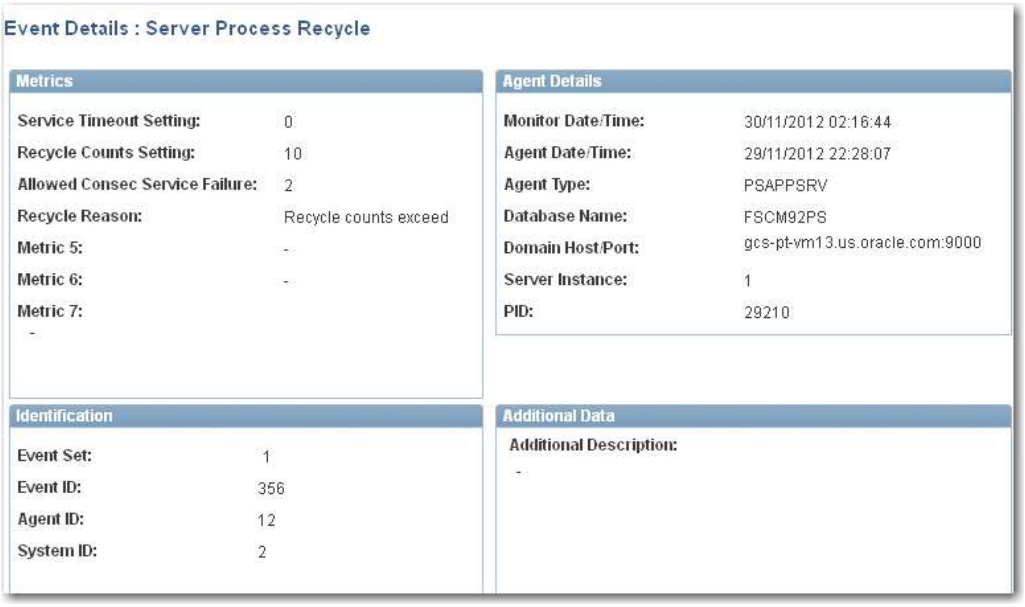
Here is an example of the search result.



Click on the Event History link for a Process ID of interest to drill into events associated with the recycle:



Here is the Event detail of the Agent Contact Notification



Similarly, you can click on the Completed PMUs link on the Recycled Processes Diagnosis page. The Completed PMUs executed by a process (PID = 29210) prior to the recycle event are shown. The functionality for monitoring server process recycling is available in PeopleTools 8.46 and later.

Chapter 6 – Common Questions - FAQ

What is a PMU?

PMU stands for Performance Measurement Unit. It represents the smallest unit of work executed in the PeopleSoft Internet Architecture. A PMU collects the following information:

- Start and end timestamp
- Application User ID responsible for initiating the PMU
- Status
- Identifier of the Agent executing the PMU
- Data collected during the lifetime of the PMU

A PMU can be related to another PMU through parent-child relationships and can therefore be represented in a tree structure.

How do I get the Verbose and Debug PMUs?

PeopleSoft Systems can be monitored at the following Agent Filter Levels:

- 01-Standby
- 02-Error
- 03-Warning
- 04-Standard
- 05-Verbose
- 06-Debug

Performance Monitor is currently delivered with PMUs associated with the **04-Standard**, **05-Verbose** and **06-Debug** Agent Filter Levels. The monitored system collects all the PMUs associated with the filter level that matches the system's current filter level and higher. For example, a monitored system at the **06-Debug** Agent Filter Level collects PMUs associated with the Agent Filter Levels **06-Debug**, **05-Verbose** and **04-Standard**.

There are two ways to get **Verbose** or **Debug** PMUs:

1. Set a system-wide **05-Verbose** or **06-Debug** Agent Filter Level for the monitored system. In the Performance Monitoring system, navigate to **PeopleTools, Performance Monitor, Administration, Agent Filters.** Pick the monitored system whose Agent Filter Level you want to change. Select the Agent Type and change the Agent Filter Level via the drop down list box. Click the Save and Notify Agents button. Normally, in a development or test environment, if you only want to get data for Top PeopleCode SQL, Top PeopleCode Events and Top PeopleCode Executions, raising the Agent Filter Level for the PSAPPSRV agent type should be sufficient.
2. Enable Performance Trace. You can safely turn on the Performance Trace feature even for a production environment. The additional data captured for the 05-Verbose or 06-Debug Agent Filter Levels only impacts the end user who initiated the Performance Trace. In the Performance Monitoring environment, navigate to **PeopleTools, Performance Monitor, Administration, System Definitions**. Pick the monitored system for which you want to enable Performance Trace. Check the Allow Performance Trace box. Next, Click the Save and Notify Agents button.

Generally speaking, using **05-Verbose** or **06-Debug** filter levels system wide is not recommended for a production environment due to the volume of data it can produce. Enabling a Performance trace at the user level is the recommended method.

Why are the Prompts for Usernames, Components, and Performance Trace Names Empty in PMU History and Analytics Pages?

Performance Monitor delivers a utility Application Engine program PSPM_LOOKUP, to cache the PeopleSoft User IDs, Component names and Performance trace names. The program gathers the values by looking at the performance data inserted

since its previous run. PeopleSoft recommends this program be scheduled to run periodically. The process can be initiated by navigating to **PeopleTools, Performance Monitor, Administration, Schedule Lookup Maintenance**.

Why Do I Get No Data on the Top PeopleCode SQL, Top PeopleCode Events, and Top PeopleCode Executions Pages?

The Top PeopleCode SQL and Top PeopleCode Events pages require data captured in **05-Verbose** PMUs by application servers (PSAPPSRV), and the Top PeopleCode Executions page requires **06-Debug** PMUs, again captured by application servers. The message appears, and acts as a reminder that the application server monitor agents for the monitored system are set to the **04-Standard** or **01-Standby** Agent Filter Level when you enter the corresponding page.

What Does the Message: "Verbose (Debug) PMU charts. Agent filter for PSAPPSRV is currently set to – Standard/Standby" Mean?

There are two ways to get **05-Verbose** or **06-Debug** PMUs, as explained in the previous FAQ, **How Do I Get the Verbose and Debug PMUs?**

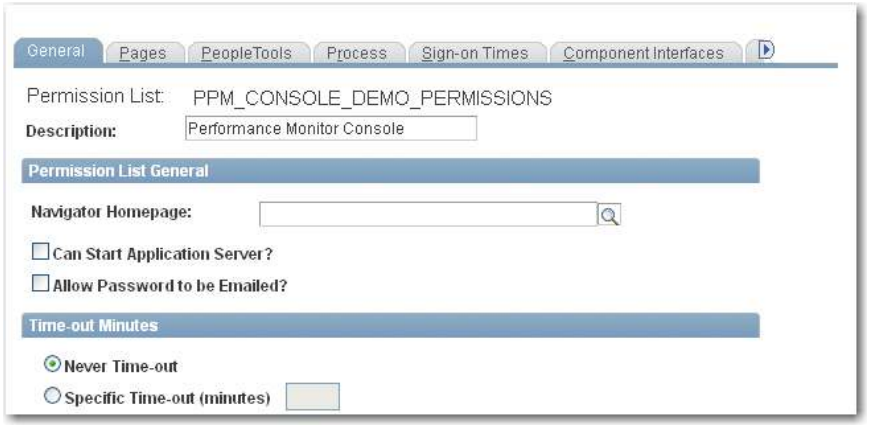
What is the Best Practice for Granting End User Access to Performance Trace in a Production Environment?

Performance Trace is a very powerful and practical tool for troubleshooting performance issues in a production environment. This is because it allows you to gather very detailed information to help determine the cause of a performance problem without impacting other uses on the system. The following is the list of recommended steps for granting Performance Trace access to the end users who report a performance issue. Repeat these steps for each of the monitored systems.

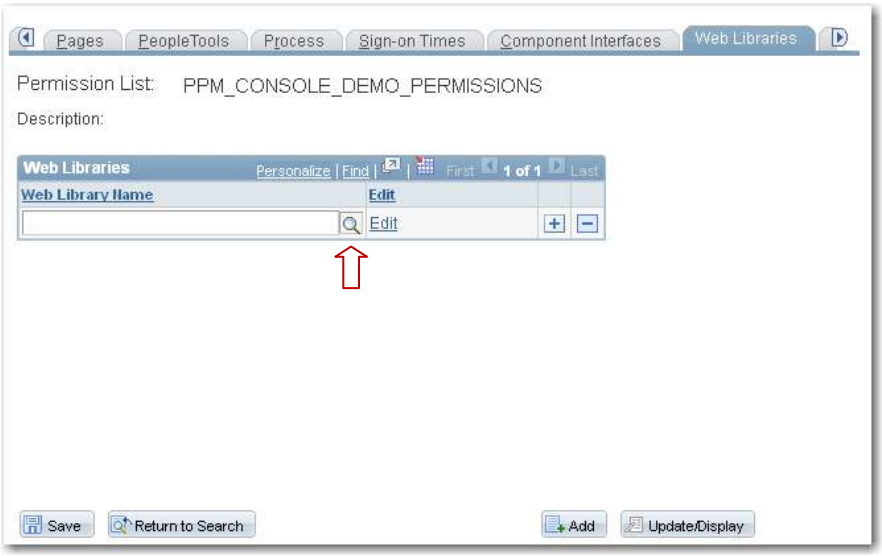
1. Create a new Permission List for Performance Trace Access in the monitored system. Navigate to

   PeopleTools, Security, Permissions & Roles, Permission Lists.

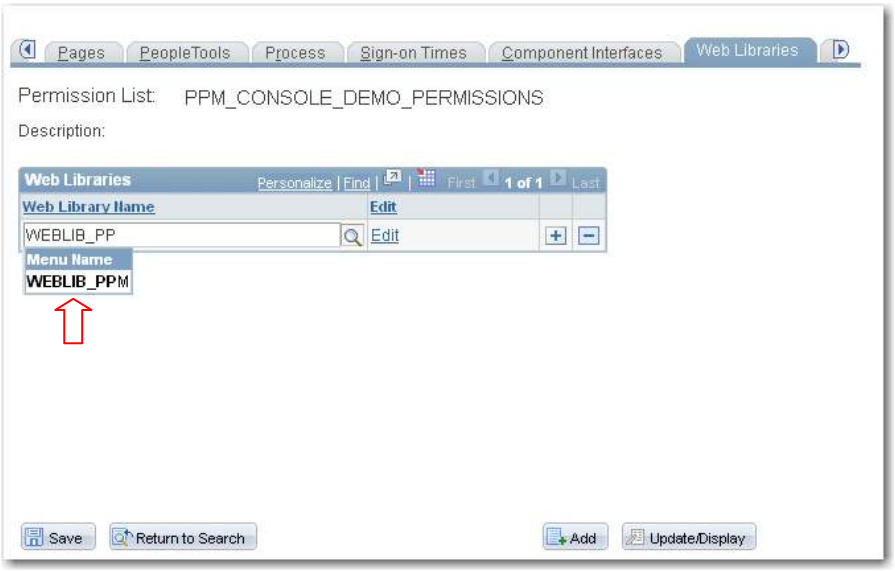
2. Add a permission list with a descriptive name, e.g. PPM_CONSOLE_DEMO_PERMISSION, that will have access to the PPM Console.



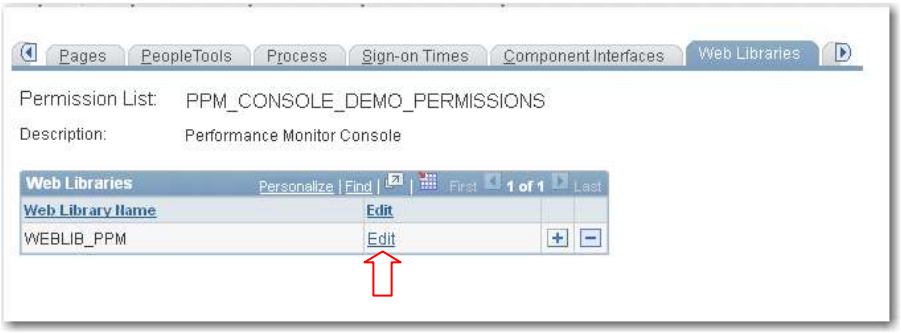
3. Navigate to the Web Libraries tab. Click on the search icon (the magnifying glass).



4. Select WEBLIB_PPM from the search results.

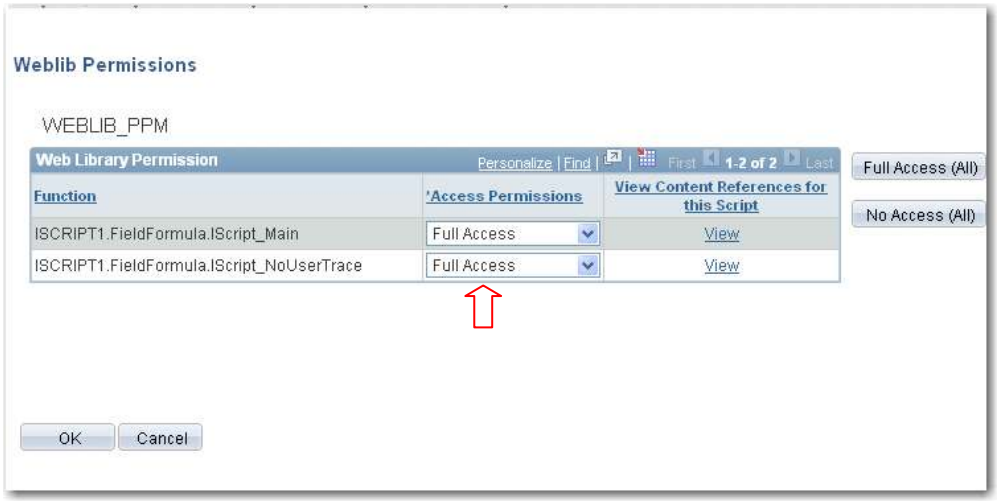


5. Click on the Edit link.

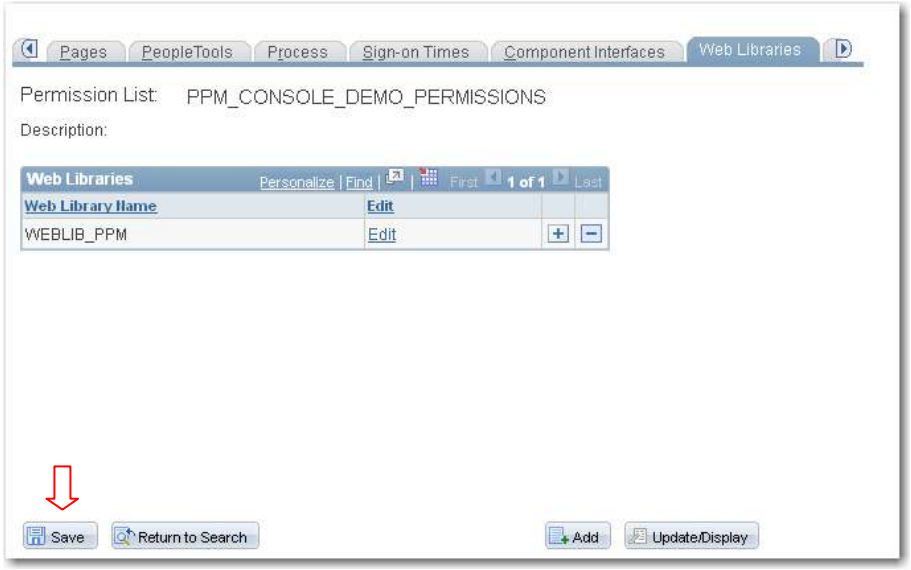


6. Click on the Full Access (All) button.

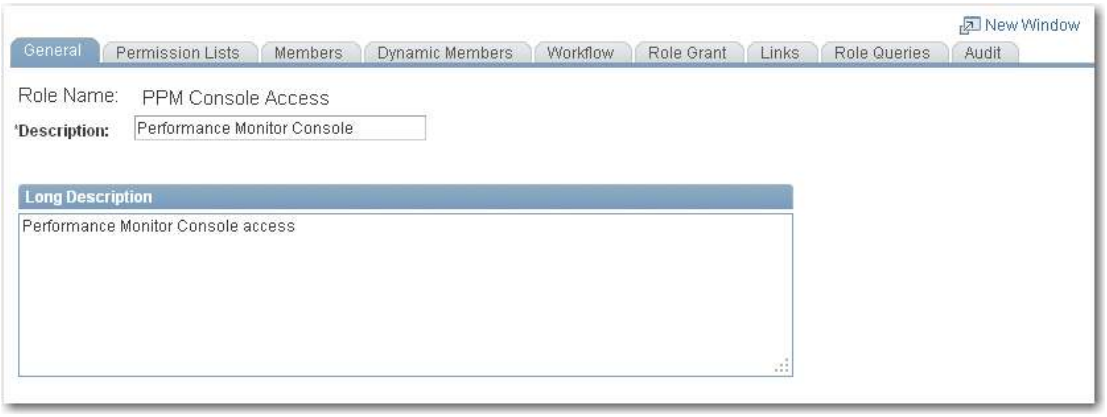
7. The Access Permissions for both functions should be set to “Full Access.” Click the Ok button.



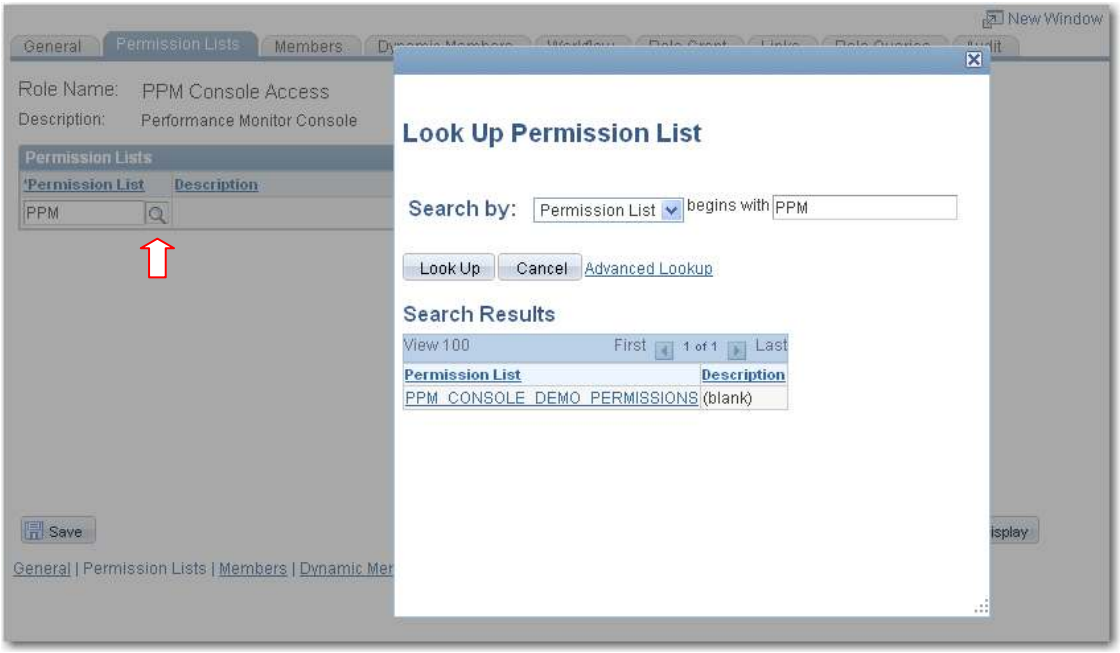
8. Click on the Save button on the Permission List page to save the result.



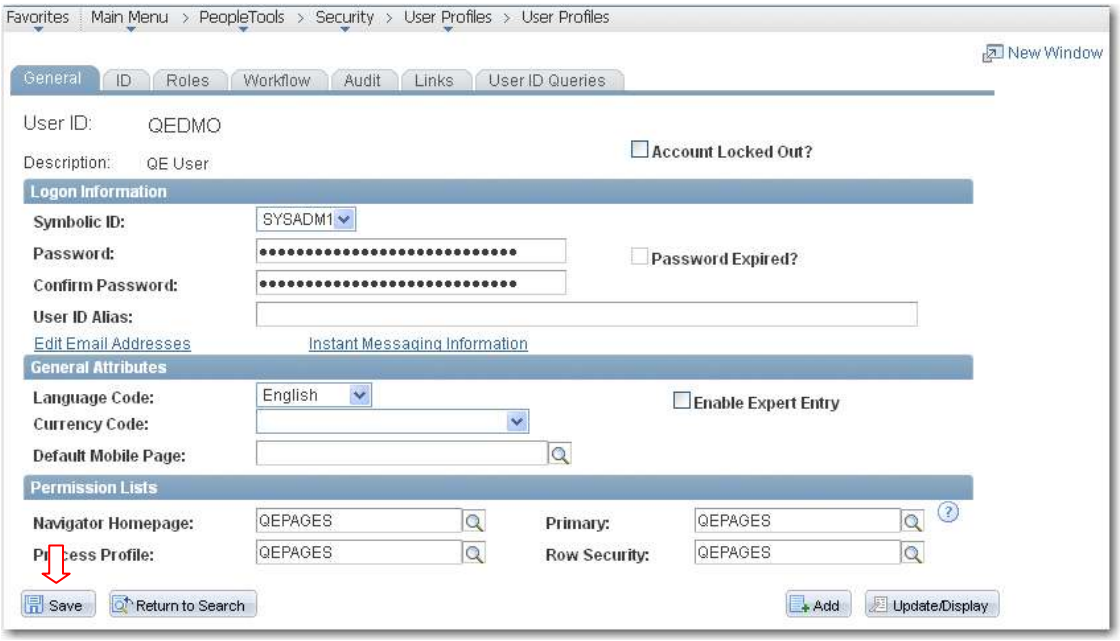
9. Next, add the permission list to a role that you can give to end users. Navigate to **PeopleTools, Security, Permissions & Roles, Roles.** Click on the Permission Lists tab. Here the “PPM Console Access” role has been added.



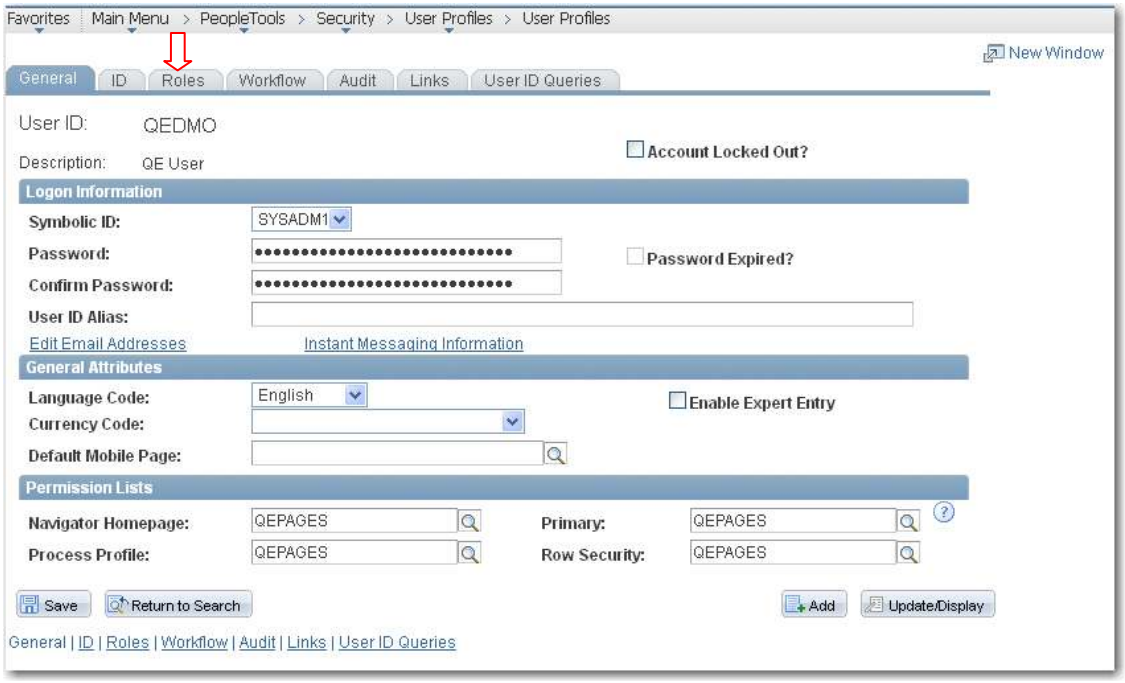
10. Click on the Search button to add the Permission List.



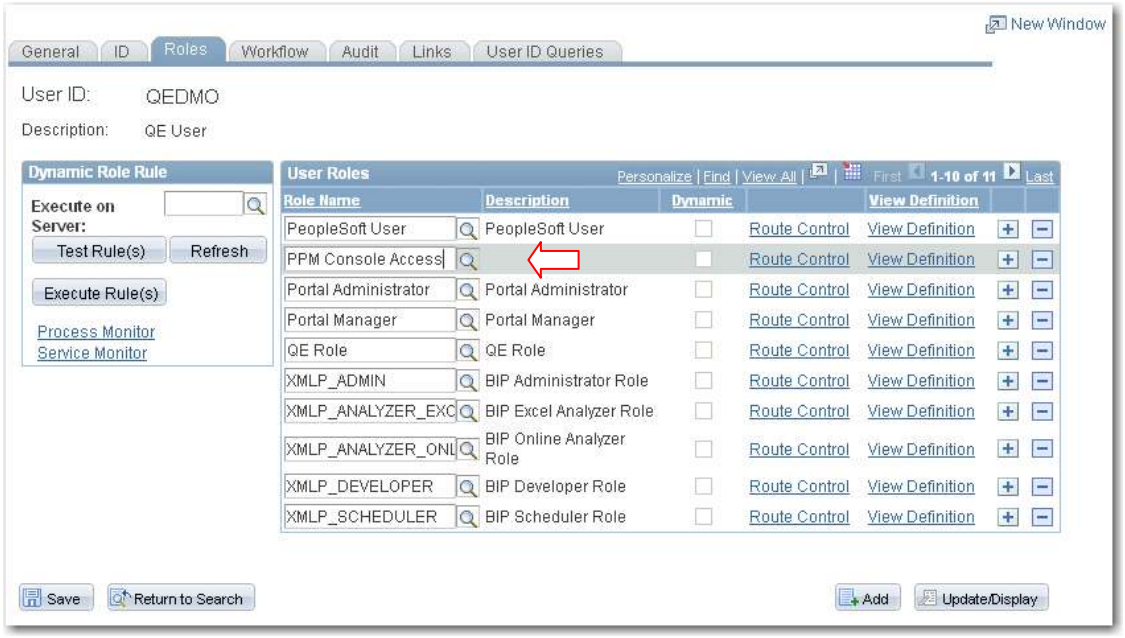
11. Save the result.



12. To give PPM Console page access to a user, you need to add the PPM Console Access role to a user's profile. Navigate to **PeopleTools, User Profiles, User Profiles**. Pick the User ID. Navigate to the Roles tab.



13. Add the PPM Console Access role, then click the Save button. The user now has access to the Performance Trace Console.

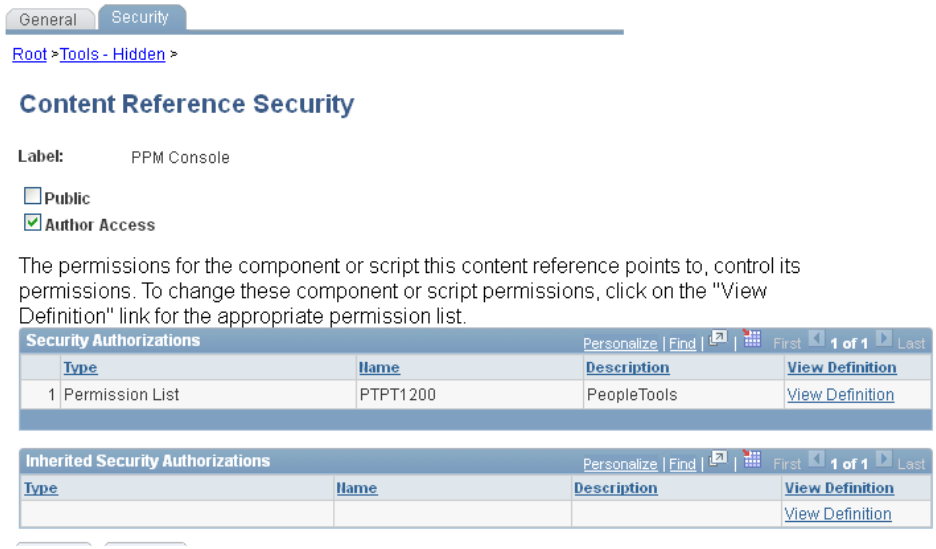


To verify that the security settings are correct for the Performance Trace Console (PPM Console), follow these steps for each of the monitored systems:

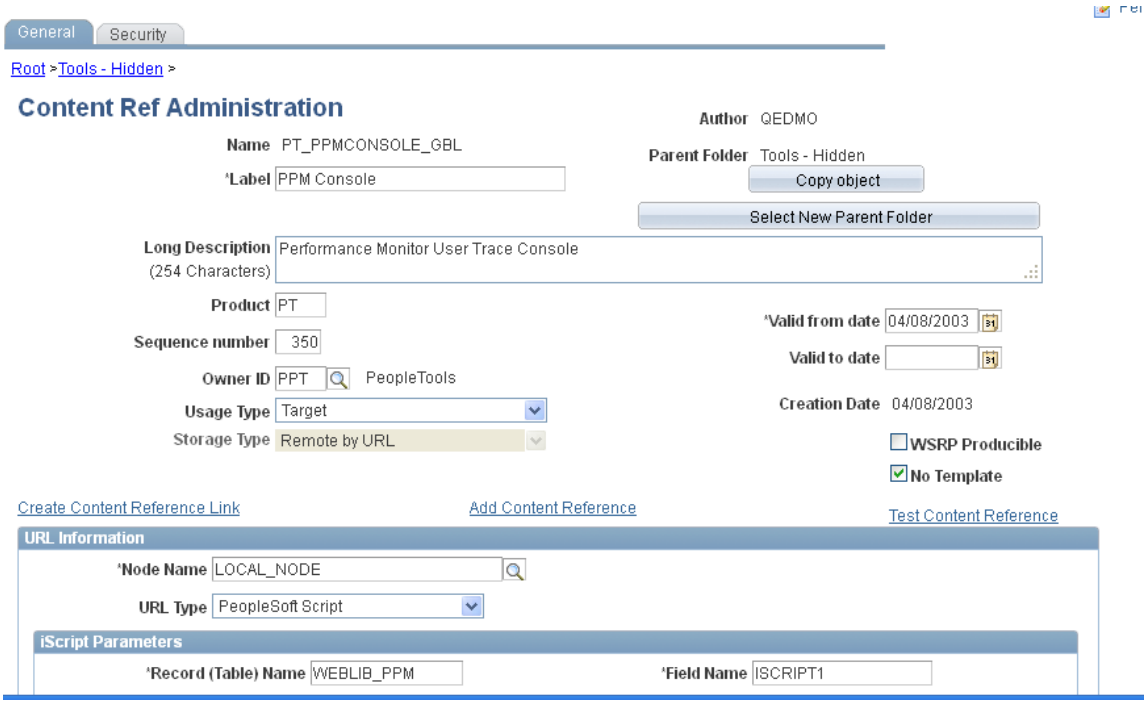
1. Navigate to

   PeopleTools, Portal, Structure and Content.

2. Select the Tools – Hidden folder link. Select the PPM Console edit link. Verify that the Security tab has the permission list that you added under the Security Authorizations section.



3. You should also click on the Test Content Reference link under the General tab to make sure the Performance Trace Console page shows up.



Then the following dialog will appear:

How Do We Measure the Network Latency Between a Client's Web Browser and the Web Server?

1. Navigate to the Current User Sessions page via

   PeopleTools, Performance Monitor, System Monitor, Current User Sessions

2. Click on the Details tab. The page displays the latency discovered during the User's login to the system. The User Session History page **PeopleTools, Performance Monitor, History, User Session History** displays the latencies for completed user sessions. Alternatively, you can use PSPing to clock the latency.

How Do We Obtain a List of Timed-out or Failed PMUs?

1. Navigate to the PMU History page

   PeopleTools, Performance Monitor, History, Completed PMUs.

2. Click on the Advanced Search hyperlink.
3. In the PMU Status dropdown list box, select Timeout. Enter other search criteria if needed and click on the Search button. You could also search for Failed and Aborted PMUs in a similar manner, by selecting the appropriate status in the PMU Status dropdown list box and clicking on the Search button.

Can I Create My Own PMUs and Events? Will it Work in PPM?

No, you cannot create new PMUs or events. Only PeopleSoft can instrument the PeopleTools runtime code. Adding new PMU and event definitions to Performance Monitor metadata does not automatically enable instrumentation.

Do I Need to Instrument My Customization for PPM?

No, there is no need to instrument or change your customization. PPM instruments the PeopleTools Component Processor, so that a PeopleCode customization will be monitored automatically.

Can I Use One Version of Performance Monitor to Monitor a PeopleSoft Environment at a Different Version?

Yes, as long as the PeopleTools is on Version 8.44 or higher. PeopleSoft will maintain the monitoring capability between the supported versions of PeopleTools. PeopleSoft recommends that the Performance Monitor environment (i.e. on the monitoring server) be at the highest possible version level. If it is, you will able to view the all the data collected from the monitored system. If the version level of the monitored system is higher than that of the monitored system, it's possible that the monitoring system will not be able to see all the performance data transmitted from the monitored system.

I am Using Portal with Multiple PeopleSoft Environments. Can a Performance Trace monitor a Transaction Across Different Environments?

Yes, as long as the user initiating the performance trace has the same User ID and is authenticated on the monitored systems.

Do I Need to Synchronize the System Time on All Servers to Get Accurate PMU Duration?

No. Performance Monitor works well even if the system clocks of the servers are not synchronized.

Can I Shutdown the PSMONITORSRV process?

There is one PSMONITORSRV process per application server domain and one per process scheduler domain regardless of whether a domain is used in a system being monitored or a system which monitors multiple systems. The PSMONITORSRV process monitors and collects the performance metrics and statistics provided by the monitoring agents, and is crucial to the proper functioning of PPM. It also issues kill query requests to stop run-away queries.

It is possible to boot a Tuxedo application server or process scheduler domain without the PSMONITORSRV process, but PPM reporting for that domain will be compromised as a result. Please refer to System and Server Administration PeopleBook for more information about the PSMONITORSRV server process

Can I Disable the PSPPMSRV process ?

The PSPPMSRV process is responsible for contacting the PPMI Monitor Servlet, receiving performance monitor data, and inserting into the database. This process is only needed in the Monitoring System. In a production environment, you only need to enable performance monitor agents ("EnablePPM Agent = 1" in the application server domain and process scheduler domain configuration files, and click the Enable PPM Agent checkbox in the web profile). You do not need the PSPPMSRV process on a Monitored System, assuming the performance monitoring system is on a separate environment.

You can disable the PSPPMSRV process by choosing No for the "Perf Collator" option in the application server quick configuration screen using the PSADMIN utility located under the <PS_HOME>/appserv directory.

Why is Collator URL created on IPV6 and not IPV 4 even though IPV 6 is disabled on the machine?

IPV6 needs to be completely disabled. Once this is done, restart all the servers (Web Server and Application Server).Refer to http://support.microsoft.com/kb/929852 on how to disable it.

IPv6 is the next generation Internet Protocol address standard intended to supplement, and eventually replace, the IPv4 protocol most Internet services use to transact on the Internet today. (used with Windows 7,Windows 2008 R2)

Keep in mind, registry setting change will be the superset of the "networking" tab of Local area connection properties, that it affects all the interfaces, whereas the networking tab affects only specific interfaces.

You can verify on the disabled components by running the following command-

```
reg query HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip6\Parameters /v DisabledComponents
```

If still seen, delete '*DisabledComponents'* under

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip6\Parameters\
```

To verify, setting it to "0xffffffff" will disable all IPv6 components (except the IPv6 loopback interface))

Why is the PSAPPSRV process causing High Memory Use when enabling Performance Monitor ?

The PSAPPSRV processes use Java when the Performance Monitor is enabled. You can control the amount of memory Java will use for each PSAPPSRV process by editing the psappsrv.cfg. Change the JavaVM Options to the settings below. When the PSAPPSRV processes now boot they should be around 300 MB each when monitoring is enabled.

```
JavaVM Options=-Xms256m -Xm512m  (append to existing options)
```

Why are all Monitored Systems recorded under One System Definition ?

First check how you have created the two database. If you are cloning the database, you need to check if the GUID in the PSOPTIONS are the same. Try the following SQLs against both monitored databases.

```
SELECT GUID FROM PSOPTIONS
```

If the GUID is the same, what you need to do is to stop Application Server domains, blank out GUID in PSOPTIONS, and start up Application Server domains.

Why are the graphs under System Performance Page not correctly displayed ?

Charts are using style sheets for adding color and other effects to the chart. For PeopleTools 8.50 you just need the PSCHARTSTYLE Style Sheet. For PeopleTools 8.51 a new sub style sheet was introduced (PSAVSCHART) and both the style sheets are needed for rendering charts (PSAVSCHART+PSCHARTSTYLE).

PSSTYLEDEF is the main style sheet and PSCHARTSTYLE and PSAVSCHART are sub style sheets. It is important to have PSCHARTSTYLE as a sub style sheet even though PSCHARTSTYLE is present in the database. If using PSSTYLEDEF for all charts then make sure PSCHARTSTYLE is attached to this style sheet. PSCHARTSTYLE and PSAVSCHART are two

required sub style sheets of the main style sheet used for creating chart. Please verify if these two sub style sheets are attached to style-sheets you would like to use.

PSCHARTSTYLE is a sub style sheet of PSSTYLEDEF (which is default style sheet for all of Tools - so please check the PSOPTIONS table for default stylesheet setting for your environment).

It is important to have PSCHARTSTYLE as a sub style sheet even though PSCHARTSTYLE is present in the database. If you are using your own style sheet for charts, then please attach PSCHARTSTYLE as a sub style sheet.

Where can I find documentation about Performance Monitor Database Sizing ?

See Documentation for Performance Monitor Database Sizing & Hardware (Doc ID 637822.1)

Chapter 7 – Data Volume Management

Performance Monitor can easily collect a large number of rows over a short period of time. For example, in a system monitored at the **04-Standard** Agent Filter Level:

- 5-7 PMUs are generated per PIA request.
- 7-9 PMUs are generated per Portal request.
- Each PMU occupies 2.4 to 3.8 KB in a Unicode database and 1.3 to 2.1 KB in an ANSI database.
- Roughly 3 Events are generated per web server.
- 1 Event per application server process in a sampling interval.
- Additionally, the tuxedo domain monitor collects 8 Events per sampling interval. Each Event occupies 0.7 to 1.1 KB in a Unicode database and 0.4 to 0.6 KB in an ANSI database.

The row size estimates include the Index space requirements. Refer to Chapter 4 of the Performance Monitor PeopleBook for PeopleTools 8.5 on how to estimate the size of a performance monitoring database.

The Performance Monitor provides a number of tools and configuration options to manage the data volume inserted into the database.

The Reaper Application Engine Program

The Performance Collator (PSPPMSRV) inserts a row for every PMU Start, Update and Stop. If a PMU Stop is received, the previous Start and Update rows are marked for deletion. It is the job of the Reaper Application Engine process to delete the rows marked for deletion. Reaper AE also harvests the timed-out PMUs and inserts them into the PMU History table (PSPMTRANSHIST) with a time-out status of "Reaper Timeout".

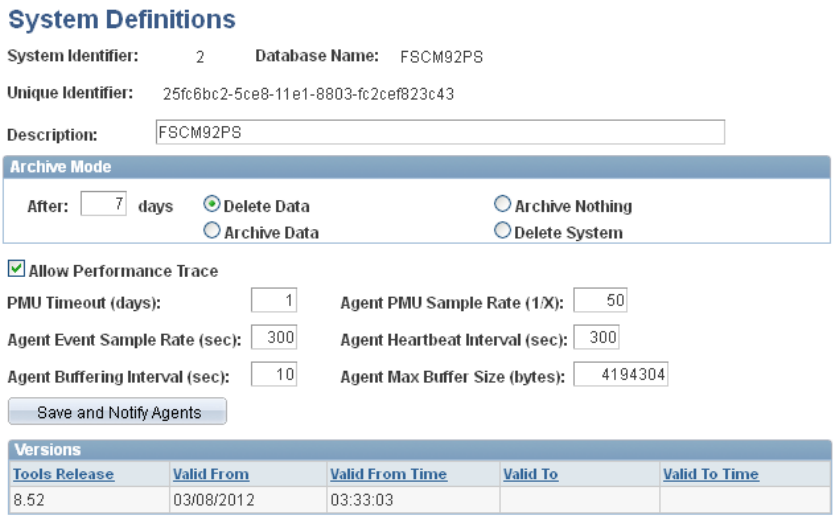
By default, PeopleSoft delivers a 15 minute recurrence, **PerfMon Reaper Recurrence**, for scheduling the Reaper AE process.

Archiving Performance Data

PPM provides the functionality to archive history data into a set of separate tables (PSPMTRANSARCH and PSPMEVENTARCH). For monitoring production environments, PeopleSoft suggests archiving data older than 7 days.

To configure this process for a particular System, navigate to **PeopleTools, Performance Monitor, Administration, System Definitions.**

To globally configure all Systems listed in the System Defaults Page, navigate to **PeopleTools, Performance Monitor, Administration, System Defaults.** Note that any of the parameters apply to all of the agents in a monitored system.



PeopleSoft recommends the Archive Application Engine program be scheduled to run daily to move the rows into the Archive tables. While scheduling the Archive program, we recommend that you select the %Run UpdateStats at the end checkbox. The history table statistics are updated which will provide better response times for the analytics and search pages.

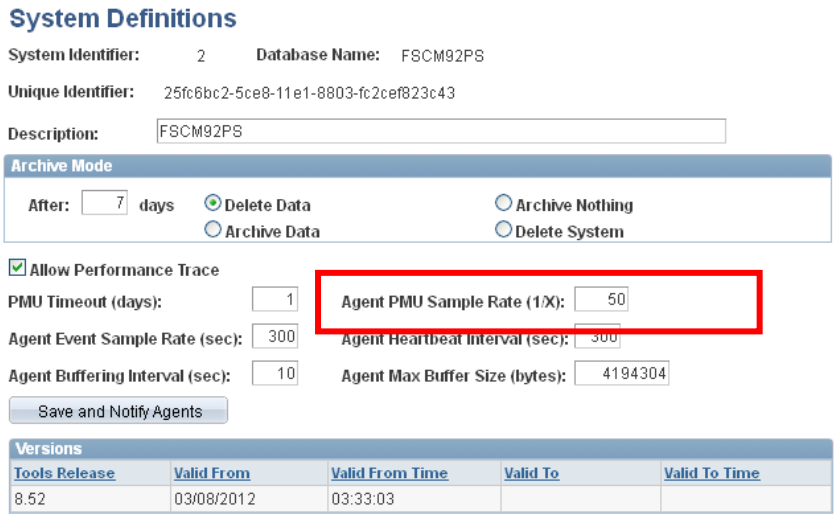
PMU SAMPLING

The Performance Monitor can be configured to be less obtrusive by specifying an Agent PMU Sample Rate. When this option is used, performance data is collected only for every nth PMU specified.

In the following example, an Agent PMU Sample Rate of 50 has been specified, which means that performance data is collected for every 50th PMU. If a parent PMU has its data collected, then all of its child PMUs data will also be collected. For example, if the PIA PMU is being sampled then all the application server PMUs triggered by the PIA PMU are sampled as well.

The Signon PMU is an exception and is always collected regardless of the sample rate.

The sample rate can be configured in the System definitions page **PeopleTools, Performance Monitor, Administration, System Definitions**. A positive value greater than 0 is required to turn on the sampling. To enable PMU sampling immediately, make sure the integration broker gateway is configured (see Step 8 in Chapter 2), then click on the Save and Notify Agents button. You can double check by going to the Show Agents monitor URL described in Chapter 2 (see example below).



Agents -- CurrentTime: Fri Aug 24 05:55:41 EDT 2012

System: 25fc6bc2-5ce8-11e1-8803-fc2cef823c43

Id	LastComm	Filter	Buf-Size	Send-Itvl	HeartBeat	Sample-Itvl	UserTrac	SamplingRate	SamplingFilter
17	Aug 24, 2012 5:54:16 AM	4	4194304	10000	300000	300000	true	50	0
19	Aug 24, 2012 5:53:59 AM	4	4194304	10000	300000	300000	true	50	0
21	Aug 24, 2012 5:54:04 AM	4	4194304	10000	300000	300000	true	50	0
20	Aug 24, 2012 5:55:35 AM	4	4194304	10000	300000	300000	true	50	0
11	Aug 24, 2012 5:54:01 AM	4	4194304	10000	300000	300000	true	50	0
12	Aug 24, 2012 5:54:05 AM	4	4194304	10000	300000	300000	true	50	0
13	Aug 24, 2012 5:54:05 AM	4	4194304	10000	300000	300000	true	50	0
14	Aug 24, 2012 5:54:07 AM	4	4194304	10000	300000	300000	true	50	0
15	Aug 24, 2012 5:55:20 AM	4	4194304	10000	300000	300000	true	50	0

COLLATOR ROW LIMIT

Performance Monitor allows you to specify a hard limit on the number of rows inserted into the PPM tables. When the limit is reached, new data received by the PSPPMSRV process is discarded and an error is logged in the Application Server log file.

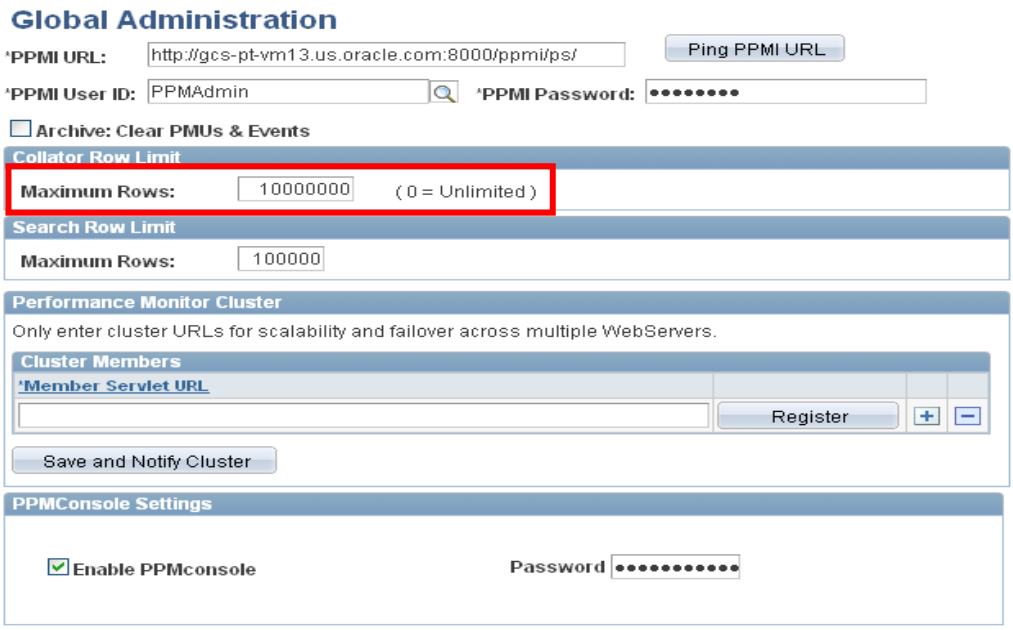
Three tables - PSPMTRANSHIST, PSPMEVENTHIST and PSPMTRANSCURR - are considered while enforcing the row limit. The archive tables, PSPMTRANSARCH and PSPMEVENTARCH are not subject to the collator row limit.

Some sort of intervention – manually removing rows or running a scheduled maintenance process - is needed to clear the data from the tables before the data insertion resumes. The row limit value can be configured in the Global Administration page by navigating to **PeopleTools, Performance Monitor, Administration, Global Administration** and is applied for the data gathered across all the monitored systems.

Since there is a loss of data when the limit is reached, PeopleSoft recommends you set this parameter to a big enough value that it is triggered only as a last resort. A value of "0" indicates that an unlimited number of rows can be inserted.

When the collator row limit is reached, the data collator process (PSPPMSRV) stops inserting data into the Performance Monitor database. This causes all of the monitor agents of all monitored system to appear as Stale in the System Performance and Server Status summary pages. However, the monitoring agents, the PPMI Monitor Servlet, and the PPMI servlet can still be actively collecting and publishing performance data.

Changes to the collator row limit do not require the rebooting of the monitoring system to go into effect.



TABLESPACE REQUIREMENTS FOR DB2/OS390

The Performance Monitor transaction tables are created in the PSIMGR tablespace. PeopleSoft recommends that customers use page level locking for this tablespace to get enhanced performance.

It is also important to configure the secondary tablespace size growth quantity to a reasonably high value so that the database doesn't grow the tablespace too frequently. PeopleSoft DBAs recommend the value to be set to at least 144,000

CHAPTER 8 – Track And Report Concurrent Usage

Signon/Signoff

At present Performance Monitor (PPM) when enabled captures user login and log off as 2 separate PMUs (108 and 109) where in 108 is the log off data and 109 is the login data. The metrics that gets collected under login data are UserID, session ID and IP Address of the logged in user. The metrics that gets collected under logoff data are UserID, session ID, IP Address of the logged off user and the reason for session termination.

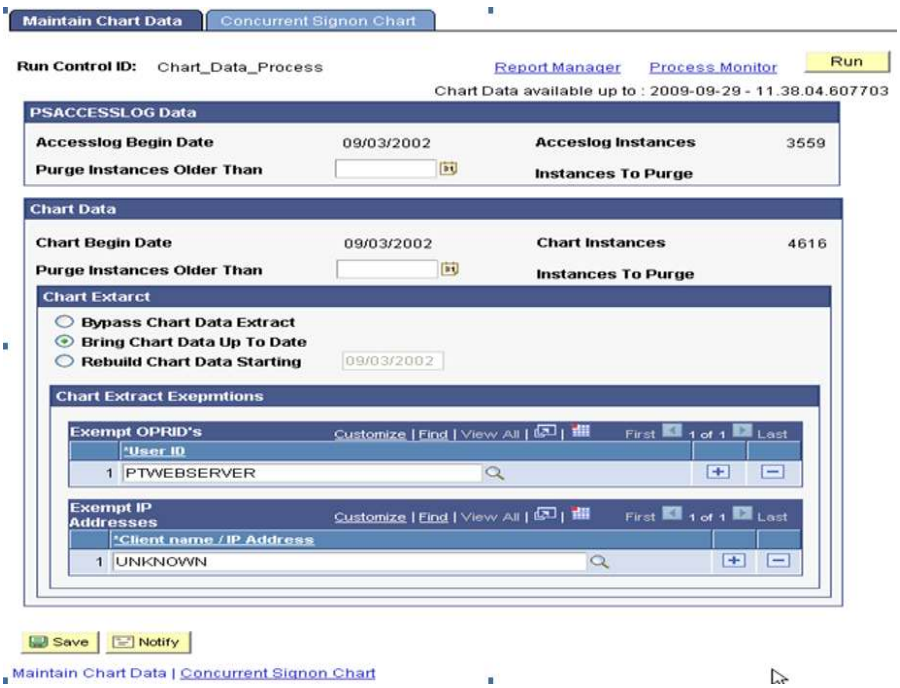
The types of log off that are tracked are normal log off, session expiration, session exception. The new type that will be tracked is session abandoned case. Session abandonment happens when the Web Server Timeout occurs, after the User browsed away from PeopleSoft or closed the browser all together without signing off from their session.

It will also capture and track Switch User Events. This will ensure the original OPRID is considered Signed off and the Switched to OPRID is considered Signed On at the moment the Switch User occurs

Configuration, Administration, and Monitoring Specifications

This is the page that will show up when first accessing the component. There is no search dialog to access this component or its pages. Since this page should have very limited access – the RunControl will not be saved by OPRID – there is one universal Run Control for this Page and the Process it submits. Navigate to:

Navigation: PeopleTools > Performance Monitor > History > Concurrent Access Console



On this page the Administrator can see how many rows are in PPM and the Chart table and the Oldest date contained in each table. The page displays the last Date Available in the Chart Data Table so the Administrator knows how up-to-date the Chart

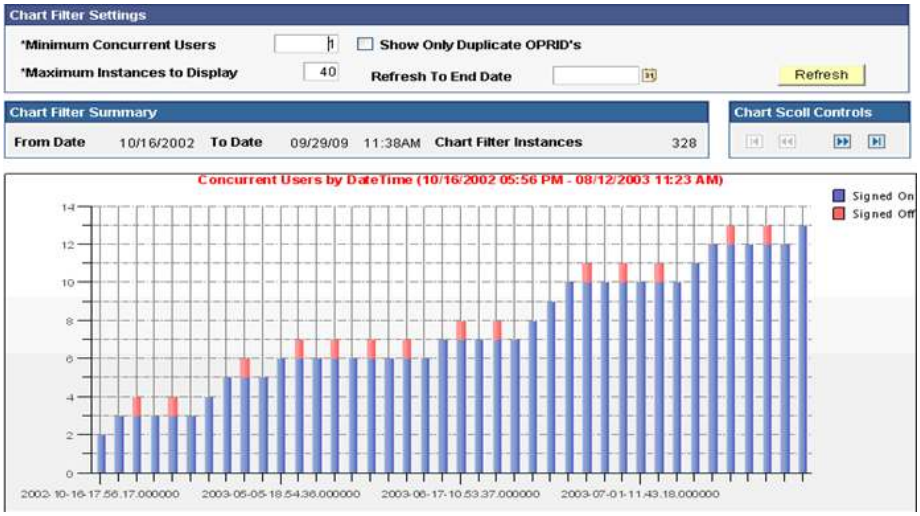
is and if it may be time to run the Extract. Pushing the Run button will invoke tradition Process Scheduler Dialogs to submit an AE Batch program. The page will not automatically refresh when the batch job is complete, the Administrator must monitor the Process and restart the component when the job is done or just come back latter and check the Chart History Summary information again.

If the data is extremely large, the Administrator may decide to purge some of the older rows. By specifying a "Purge Instances Older Than" Date for the PPM or Chart Data, the page will update showing how many rows will be purged in "Instances To Purge". The Purges will be done as part of the AE batch processing. Options are provided for the Administrator to bypass doing any Chart extract processing (just do Purge Processing), To only extract Instances that are newer than currently exist in the Chart Data, or to pick a date that already exists in the Chart Data Table – in which case those Instances will be rebuilt.

There may be situations where it may be desirable to exempt certain OPRID's or IP Addresses from being monitored. The Administrator can specify those settings on this page.

Concurrent Signon Chart

This is the page that will show up when accessing the component. There is no search dialog to access this component or its pages.



The graph first comes up showing up to the "Maximum Instance to Display" bars on the chart, with the most current data point showing as the last column of the graph. The Chart Filter Setting Group box contains the settings used to filter the Type and Amount of data displayed in the Chart. Changes to any of these setting will only be applied after the refresh button is pushed.

Minimum Concurrent Users is a required field. Any record of current Usage below this limit will not be included in the Chart content.

The Show Only Duplicate ORPID's checkbox when clicked will override the Minimum Concurrent Users setting to 2 and gray the Minimum Concurrent Users field. The Chart Content will be refreshed with only those instances were the Same OPRID was logged on more than once at the same time. Maximum Bars to Display is a required field. This setting gives the User control over how much information to display on the Chart at a time. The drill down capability of the chart become unusable at about 100 bars on the chart. The user will be able to display a maximum of 999 bars.

The Chart Filter Summary group box shows the Date Range and total number of rows currently available for display in the chart based on the filter settings. The amount of data may be too large to display in a single graph and maintain readability/usability. This information is updated every time the Refresh button is pushed to apply new filter settings.

The scroll buttons for First, Prev, Refresh, Next and Last are provided to update the Chart display:

- The First Button will refresh the Chart so that the very first data point available will be shown as the first bar of the graph (or will position to the First/Last Scroll Date if specified).
- The First and Prev buttons will be grayed out if the chart is displaying the first row of data. When the Prev/Next Button is pushed the graph moves to the prior/next set of data based on the "Maximum Bars to Display" setting. The Chart above is showing 20 of the data points from the 328 available, so these buttons allow the user to look at all 328 rows 20 at a time.

- The Last Button will refresh the Chart so that the very last data point available will be shown as the last bar of the graph (or will position to the First/Last Scroll Date if specified).
- The Last and Next buttons will be grayed out if the chart is displaying the first row of data.

The graph itself initially display the Very Last Data point available as the last Bar of the Graph. The Heading of the Graph contains the date range that is visible. The visible date range displayed in the heading will be any time the content of the graph is changed.

The Graph displays the concurrent user information as of the current state of the Session in a 2D stacked bar graph format. The Time count shown by each bar represents the fact that this number of User's were signed on to the system at the this point in time. By using the 2D graph based on the Current state of the Session you can see of those people that were signed on at that time, how many are still signed on, how many eventually signed out normally, and how many were signed because their session expired.

Each Bar of the chart will display hover text information about the colored bar under the mouse. The standard chart format will be used which is in the form 'Value X for point Y of Series Z' X being the number of Users, Y being the DTTM and Z being the Legend text associated with the bar color.

Each Bar of the Chart is Drillable, meaning that the user can click on the bar to see the detail data represented by all the colors in the Bar. The following secondary appears when a Bar is clicked. Note: this functionality becomes unusable when the graph contains around 100 or more bars of data.

Concurrent Users as of 2003-07-01-12.06.13.000000

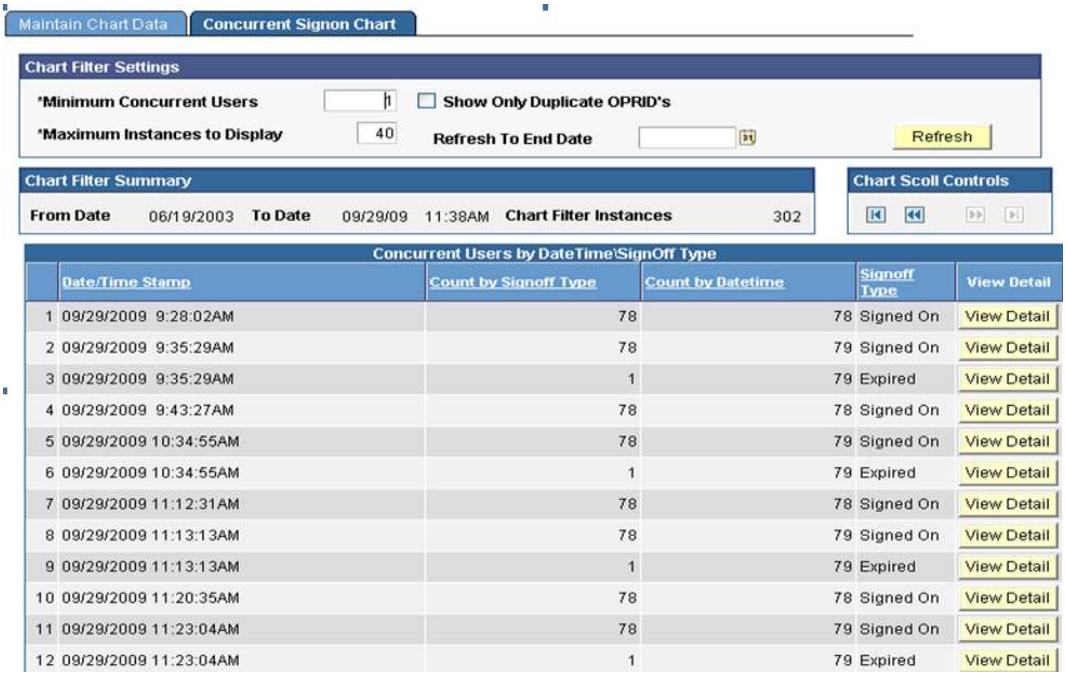
Customize | Find | View All | First 1-10 of 10 Last

	User ID	Client name / IP Address	Log In Time	Log Out Time	Signon Type	Signoff Type
1	QEADMIN	hlamcmpq.corp.peoplesoft.com	10/16/02 5:24PM	10/16/02 5:24PM	PIA	Signed On
2	QEDMO	hlamcmpq.corp.peoplesoft.com	10/16/02 5:56PM	10/16/02 5:56PM	PIA	Signed On
3	QEADMIN	hlamcmpq.corp.peoplesoft.com	10/29/02 6:21PM	10/29/02 6:21PM	PIA	Signed On
4	QEADMIN	HLAM032002.corp.peoplesoft.com	02/20/03 11:49AM	02/20/03 11:49AM	PIA	Signed On
5	QEADMIN	HLAM032002.corp.peoplesoft.com	02/20/03 4:01PM	02/20/03 4:01PM	PIA	Signed On
6	QEADMIN	HLAM042503.corp.peoplesoft.com	05/05/03 6:54PM	05/05/03 6:54PM	PIA	Signed On
7	QEADMIN	HLAM042503.corp.peoplesoft.com	06/03/03 2:16PM	06/03/03 2:16PM	PIA	Signed On
8	QEADMIN	HLAM042503.corp.peoplesoft.com	06/19/03 11:55AM	06/19/03 11:55AM	PIA	Signed On
9	QEADMIN	HLAM042503.corp.peoplesoft.com	06/19/03 11:55AM	06/19/03 11:55AM	PIA	Signed On
10	QEADMIN	HLAM042503.corp.peoplesoft.com	06/19/03 12:05PM	06/19/03 12:05PM	PIA	Signed On

Return

In Accessibility Mode the Chart will be replaced by a Grid and instead of Clicking a Bar there will be a Button to click to get to the Detail Page. The chart scroll button will remain in control of the display, any time the refresh button is pushed to apply changes in the filter settings, the grid will update just as the Chart would.

The following screen is showing you all signons/expired sessions for one Userid for the selected time period.



Maintain Chart Data | Concurrent Signon Chart

Chart Filter Settings

*Minimum Concurrent Users 1 Show Only Duplicate OPRID's

*Maximum Instances to Display 40 Refresh To End Date Refresh

Chart Filter Summary

From Date 06/19/2003 To Date 09/29/09 11:38AM Chart Filter Instances 302

Chart Scoll Controls

Concurrent Users by DateTime\SignOff Type

	Date/Time Stamp	Count by Signoff Type	Count by Datetime	Signoff Type	View Detail
1	09/29/2009 9:28:02AM	78	78	Signed On	View Detail
2	09/29/2009 9:35:29AM	78	79	Signed On	View Detail
3	09/29/2009 9:35:29AM	1	79	Expired	View Detail
4	09/29/2009 9:43:27AM	78	78	Signed On	View Detail
5	09/29/2009 10:34:55AM	78	79	Signed On	View Detail
6	09/29/2009 10:34:55AM	1	79	Expired	View Detail
7	09/29/2009 11:12:31AM	78	78	Signed On	View Detail
8	09/29/2009 11:13:13AM	78	79	Signed On	View Detail
9	09/29/2009 11:13:13AM	1	79	Expired	View Detail
10	09/29/2009 11:20:35AM	78	78	Signed On	View Detail
11	09/29/2009 11:23:04AM	78	79	Signed On	View Detail
12	09/29/2009 11:23:04AM	1	79	Expired	View Detail

Appendix A – Revision History

AUTHORS

Bor-Ruey Fu, Senior Product Manager, Performance.

Sekhar Korupolu, Developer, PeopleTools.

John Houghton, Senior Principal CoE Engineer

Peter Schwarz, Senior Principal Technical Support Engineer

Terry Martin, Senior Principal Technical Support Engineer

REVIEWERS

The following people reviewed this Red Paper:

- Mary Manchukian, Staff Support Analyst, PeopleSoft Global Support Center - Server Tools.
- Scott A Forsgren, Development Manager, PeopleTools Lifecycle Management.
- Todd Plantenga, PeopleSoft Global Support Center - Server Tools.
- Rene Zheng, Developer, PeopleTools.
- Su May Halim, Developer, PeopleTools.
- Soo Ong, Staff Support Analyst, PeopleSoft Global Support Center - Server Tools.
- Hervé Thierry, Principal Consultant, PeopleSoft Global Services – French.

REVISION HISTORY

1. 12/31/2004: Created document.
2. 1/12/2005: Version 1.0.
3. 1/18/2005: Version 1.2.
4. 6/2006: Version 1.2.1. Correct reference to permission list PTPT1200. Add PeopleTools 8.46 known issue. Add Sample Query view in Query Manager menu. Add new information in process scheduler domain log in the setup check list. Add index on PSPMTRANSHIST to improve performance. Update Perf Collator option number (option 9) for PeopleTools 8.47. Add more questions to FAQ. Add Known Issues 8-10. Update workaround information in Know Issue 5.
5. 4/2009 - Major rewrite of the Chapters 2 and 3. General reformatting and editing. Version 1.3.
6. 2/2011: Version .1.4. Editing of URL in Chapter 2, point 10, “Enter the Global Administration URL and User ID information.”
   changed from http://<host>[:port]/*monitor*/<web site>/ to httpp://<host>[:port]/*ppmi*/<web site>/ Version 1.4
7. 11/2102: Version 1.5. Updated RedPaper to reflect changes up to PeopleTools 8.52
8. 4/11/2013 – Version 1.6 to reflect Concurrent Signon, and other updates from PeopleTools 8.53

Appendix B – Known Issues and Bug Fixes

RECOMMENDED WORKAROUNDS

Slow System Performance Page and Component Trace Page

[PT8.44, 8.45, 8.46] Add two new indexes PM_TRANS_DEFN_ID, and PM_PERF_TRACE to table PSPMTRANSHIST. Here are the steps.

1. Open Application Designer (PSIDE).
2. Open Record, PSPMTRANSHIST.
3. With the record window active, select Tools, Data Administration, Indexes.
4. Click Add Index button. Leave Unique, cluster checkboxes unchecked. Set platform to All. Enter a comment, such as Date/Time, Name, Workaround to improve performance. Select PM_TRANS_DEFN_ID from Record Field on the right and click ≤ button to add to the new index. Set A/D (sort order in ascending or descending) to Asc. Click OK.
5. Repeat Step 4 and select the PM_PERF_TRACE record field.
6. Select Build, Current Definition. At this point, the Build Scope box will contain PSPMTRANSHIST. Check the Create Indexes and Build Script File check boxes in the Build Execute Options group box . Then click the Build button. Write down the file name and output directory for the SQL script. Make sure Application Designer does not display error messages.
7. Use your SQL tool and log into the database. Run the SQL script. Alternatively, you can combine steps 6 and 7 in Application Designer. To do so, check the Execute and Build Scripts checkbox, as opposed to the Build Current Definition Build Script (only) option you specified in step 6.

[PT8.44, 8.45, 8.46] Modify and rebuild SQL view (PSPMSESSIONS_VW) definition to improve performance in the System Performance page.

1. Open Application Designer (PSIDE).
2. Open Record, PSPMSESSIONS_VW. Click on Record Type tab. Click “Click to open SQL editor”.
3. Replace SQL definition with

```
SELECT T3.PM_CONTEXT_VALUE1
 , T3.PM_AGENTID
 , T3.PM_AGENT_STRT_DTTM
  FROM PSPMTRANSHIST T3
 WHERE T3.PM_TRANS_DEFN_ID = 116
   AND PM_TRANS_STATUS = 1
   AND PM_TRANS_DURATION <> 0
   AND T3.PM_CONTEXT_VALUE1 IN (
 SELECT T.PM_CONTEXT_VALUE1
  FROM PSPMTRANSHIST T
 WHERE T.PM_TRANS_DEFN_SET = 1
   AND T.PM_TRANS_DEFN_ID = 109
   AND T.PM_MON_STRT_DTTM > %TimeAdd(%CurrentDateTimeIn, -720)
   AND T.PM_PARENT_INST_ID <> PM_TOP_INST_ID
   AND T.PM_CONTEXT_VALUE1 NOT IN (
 SELECT T2.PM_CONTEXT_VALUE1
  FROM PSPMTRANSHIST T2
 WHERE T2.PM_TRANS_DEFN_SET = 1
   AND T2.PM_TRANS_DEFN_ID = 108
   AND T2.PM_CONTEXT_VALUE1 = T.PM_CONTEXT_VALUE1
```

```
AND T2.PM_MON_STRT_DTTM > %TimeAdd(%CurrentDateTimeIn, -720)))
```

4. Save the change.
5. Rebuild the view. Build -> Current Definition. Make sure Create Views is checked. The “Index creation options” settings in the create tab is set to “Recreate index if it already exists”.

Getting error “HttpTargetConnector:PSHttpFactory init or setCertificate fails (158,99999)" when changing Agent Filter Level on Performance Monitor Administration pages

The secureFilekeystorePasswd in integration.propeties file is encrypted. It needs to be plain text in PeopleTools 8.49. For PeopleTools 8.50+, secureFilekeystorePasswd should be encrypted but it is not.

To unencrypt the password in integrationgateway.properties file,

5. Go to setting Integration Gateway CERTIFICATE section. In "secureFilekeystorePasswd=" line make sure it says no to encrypt the password.
6. Restart the web server.
7. Go to PeopleTools > Performance Monitor > Administration > System Defaults > Apply to Current Systems (push button) and test. The error should not occur any more.

Problem exists in PT 8.51 and is worked by development with Bug 11849975 - ERROR WHEN CHANGING STATUS OF AGENTS FOR PERFORMANCE MONITOR

NOTE: PeopleTools 8.4x - 8.49, secureFilekeystorePasswd should not be, but is encrypted. PeopleTools 8.50+, secureFilekeystorePasswd should be encrypted, but it is not.

WORKAROUND:
It works if you enter the 'password' and then encrypt. (it comes delivered (PeopleTools 8.51) with 'password' which is unencrypted and that wasn't working)

Performance Problems Running the PPM Archive Process (PSPMARCH Psjob)

The issue is caused by some missing indexes and an unoptimized SQL statement.

1. Please add the following indexes and replace the SQL statement by modifying the Application Engine program PSPMARCH in Application Designer.

```
CREATE INDEX ADMIN.bgpspmeventhist
ON ADMIN.PSPMEVENTHIST(PM_AGENTID, PM_MON_DTTM )
PCTFREE 10
STORAGE(INITIAL 10M NEXT 40M
PCTINCREASE 0 MAXEXTENTS 9999)
TABLESPACE PSINDEX;

CREATE INDEX ADMIN.bgpspmtranshist
ON ADMIN.PSPMtransHIST(PM_AGENTID, PM_MON_STRT_DTTM )
PCTFREE 10
STORAGE(INITIAL 10M NEXT 40M
PCTINCREASE 0 MAXEXTENTS 9999)
TABLESPACE PSINDEX;
```

2. Altered SQL statements in PSPM_ARCHIVE app engine code

```
/*-- SELECT THE ROWS FROM PSPMTRANSHIST ELIGLIBLE FOR ARCHIVING */ &TransHistSQL.Open("SELECT
X.PM_INSTANCE_ID, X.PM_TRANS_DEFN_SET, X.PM_TRANS_DEFN_ID, X.PM_AGENTID, X.PM_TRANS_STATUS,
X.OPRID, X.PM_PERF_TRACE, X.PM_CONTEXT_VALUE1, X.PM_CONTEXT_VALUE2, X.PM_CONTEXT_VALUE3,
X.PM_CONTEXTID_1, X.PM_CONTEXTID_2, X.PM_CONTEXTID_3, X.PM_PROCESS_ID,
%DateTimeOut(X.PM_AGENT_STRT_DTTM), %DateTimeOut(X.PM_MON_STRT_DTTM), X.PM_TRANS_DURATION,
X.PM_PARENT_INST_ID, X.PM_TOP_INST_ID, X.PM_METRIC_VALUE1, X.PM_METRIC_VALUE2, X.PM_METRIC_VALUE3,
X.PM_METRIC_VALUE4, X.PM_METRIC_VALUE5, X.PM_METRIC_VALUE6, X.PM_METRIC_VALUE7, X.PM_ADDTNL_DESCR,
Z.PM_ARCHIVE_MODE FROM PSPMTRANSHIST X, PSPMAGENT Y, PSPMSYSDEFN Z WHERE X.PM_AGENTID=Y.PM_AGENTID
```

```
AND Y.PM_SYSTEMID=Z.PM_SYSTEMID AND (Z.PM_ARCHIVE_MODE='1' OR Z.PM_ARCHIVE_MODE='2') AND
X.PM_MON_STRT_DTTM < (SYSDATE - PM_MAX_HIST_AGE)");

/*-- SELECT THE ROWS FROM PSPMEVENTHIST ELIGLIBLE FOR ARCHIVING */ &EventHistSQL.Open("SELECT
X.PM_INSTANCE_ID, X.PM_EVENT_DEFN_SET, X.PM_EVENT_DEFN_ID, X.PM_AGENTID,
%DateTimeOut(X.PM_AGENT_DTTM), %DateTimeOut(X.PM_MON_DTTM), X.PM_PROCESS_ID,
X.PM_FILTER_LEVEL,X.PM_METRIC_VALUE1, X.PM_METRIC_VALUE2, X.PM_METRIC_VALUE3, X.PM_METRIC_VALUE4,
X.PM_METRIC_VALUE5, X.PM_METRIC_VALUE6, X.PM_METRIC_VALUE7, X.PM_ADDTNL_DESCR, Z.PM_ARCHIVE_MODE
FROM PSPMEVENTHIST X, PSPMAGENT Y, PSPMSYSDEFN Z WHERE X.PM_AGENTID=Y.PM_AGENTID AND
Y.PM_SYSTEMID=Z.PM_SYSTEMID AND (Z.PM_ARCHIVE_MODE='1' OR Z.PM_ARCHIVE_MODE='2') AND X.PM_MON_DTTM
< (SYSDATE - PM_MAX_HIST_AGE)");
```

