

HP Operations Smart Plug-in for BEA WebLogic Server

for HP Operations Manager for Windows®

Software Version: 5.10

Configuration Guide

Document Release Date: December 2007
Software Release Date: December 2007



Legal Notices

Warranty

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

The information contained herein is subject to change without notice.

Restricted Rights Legend

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

Copyright Notices

© Copyright 2003-2007 Hewlett-Packard Development Company, L.P.

Trademark Notices

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

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

UNIX® is a registered trademark of The Open Group.

Documentation Updates

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

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

To check for recent updates, or to verify that you are using the most recent edition of a document, go to:

http://ovweb.external.hp.com/lpe/doc_serv/

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

Support

You can visit the HP software support web site at:

www.hp.com/go/hpsoftwaresupport

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

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

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

To find more information about access levels, go to:

www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

www.managementsoftware.hp.com/passport-registration.html

Contents

1	Introduction	9
2	Installing, Upgrading, and Removing the WebLogic SPI	11
	Installing the WebLogic SPI	11
	Upgrading the WebLogic SPI	11
	Policy Changes	11
	Prerequisites	12
	Upgrading the WebLogic SPI	12
	Removing the WebLogic SPI	14
	Task 1: Remove All WebLogic SPI Policies from the Managed Nodes	14
	Task 2: Remove WebLogic SPI Node Groups on the Management Server	14
	Task 3: Remove the WebLogic SPI Software from the Management Server	14
3	Configuring the WebLogic SPI	15
	Configuration Prerequisites	15
	Task 1: Add Managed Nodes	15
	Task 2: Verify the Application Server Status	15
	Task 3: Collect WebLogic Login Information	16
	Task 4: Collect Names of HPOM Managed Nodes	17
	Basic WebLogic SPI Configuration	18
	Configuration Prerequisite	18
	Task 1: Run Discover WebLogic	18
	Task 2: Verify the Discovery Process	20
	Task 3: Set Additional Properties	22
	Configuring WebLogic SPI for WebLogic Servers Running on HTTPS	24
	Additional WebLogic SPI Configuration	25
	WebLogic SPI in High Availability Environments	26
	Configuration Prerequisites	26
	Configuring WebLogic SPI for High Availability Environments	26
	Task 1: Create the WebLogic SPI monitoring configuration file	26
	Task 2: Create the clustered application configuration file	27
	Task 3: Configure WebLogic SPI for HTTPS or DCE Agent (Based on Requirement)	28
	Additional Discovery and Configuration Scenarios	29
	Use Case 1: Administration Port Turned On (WebLogic Servers are Running in HTTPS Mode)	29
	Use Case 2: Administration Port Not Turned On (WebLogic Server is Running on Virtual IP)	30
4	Customizing WebLogic SPI Policies	31
	WebLogic SPI Policy Groups and Types	31
	WebLogic SPI Policy Groups	31
	WebLogic SPI Policy Types	32

Basic Policy Customizations	33
Modifying Metric Policies	33
Threshold Level and Actions	33
Message and Severity	35
Advanced Policy Customizations	37
Creating New Policy Group	37
WebLogic SPI Collector/Analyzer Command with Parameters	38
Basic WebLogic Server Command Parameters	38
Using JMX Actions Command Parameters	40
Changing the Collection Interval for Scheduled Metrics	42
Changing the Collection Interval for Selected Metrics	43
Customize Threshold Values for Different Applications/EJB/Servlet/JDBC	43
Examples	44
Creating Custom Tagged Policies	45
Restoring Default WebLogic SPI Policies	46
Viewing Text-Based Reports	46
Automatic Command Reports	46
Manually Generated Reports	47
Sample Report	47
WebLogic SPI Graphs	48
Monitoring WebLogic Server on Unsupported Platforms	49
Monitoring Remote Nodes (Running on Platforms Not Supported by WebLogic SPI)	49
Implementing Remote Monitoring	49
Configuring Remote System Monitoring	50
Task 1: Configure the Remote WebLogic Server	51
Task 2: (Optional) Integrate HP Performance Agent	51
Task 3: Assign Local Node to WebLogic SPI node group	51
Configuring Remote Monitoring for Logfiles (Optional)	52
Configuring the Logfile Policy for Remote Logfiles	52
Limitations in Remote Monitoring	53
5 Integrating HPOM Reporting and Graphing Features with the WebLogic SPI	55
Integrating the WebLogic SPI with HP Performance Agent	57
Integrating the WebLogic SPI with HP Reporter	58
Viewing Reports from the HPOM Management Console	60
Reports Generated by Reporter	61
Removing the WebLogic SPI Reporter Package	65
Integrating the WebLogic SPI with HP Performance Manager	65
Viewing Graphs that Show Alarm Conditions	65
Viewing Graphs that Show Past or Current Conditions	66
Viewing Graphs from the HP Performance Manager Console	66
WebLogic SPI Metrics Available for Graphs	67
Removing the WebLogic SPI Grapher Package	69
6 User-Defined Metrics	71
Metric Definitions DTD	72
The MetricDefinitions Element	72
Example	72

The Metric Element	73
Example	73
The MBean Element	74
Example	75
FromVersion and ToVersion Elements	76
Example	76
Calculation and Formula Elements	76
Syntax	77
Functions	77
Examples	77
Sample 1	77
Sample 2	78
Sample 3: Metric Definitions File	79
Creating User-Defined Metrics	82
Task 1: Disable Graphing (if Enabled)	82
Task 2: Create a Metric Definitions File	82
Task 3: Configure the Metric Definitions File Name and Location	82
Task 4: Create a UDM Policy Group and Policies	83
Task 5: Deploy the Policy Group	84
Task 6: Enable Graphing	84
7 Troubleshooting the WebLogic SPI	85
The Self-Healing Info Tool	85
Log and Trace Files	86
UNIX Managed Nodes	86
Windows Managed Nodes	87
Troubleshooting the Discovery Process	89
Other Discovery Related Problems	91
Manually Deploying the Discovery Policies	92
Verifying the Java Home Directory	92
Troubleshooting the Configuration	94
Verifying the Node Name	94
Troubleshooting the Tools	95
Glossary	97
Index	103

1 Introduction

The HP Operations Smart Plug-in for BEA WebLogic Server (WebLogic SPI) allows you to manage BEA WebLogic servers from an HP Operations Manager for Windows (HPOM) console. WebLogic SPI adds monitoring capabilities otherwise unavailable to HPOM. For more information on HPOM, see to the HPOM console online help.

From the HPOM for Windows console, you can monitor the availability, use, and performance of WebLogic servers running on HPOM managed nodes. You can integrate WebLogic SPI with other HP Software products like HP Reporter and HP Performance Manager to get consolidated reports and graphs which help you analyze trends in server usage, availability, and performance.

The WebLogic SPI online help provides information about WebLogic SPI concepts and other topics that will help you understand the product.

This guide covers the following topics:

- [Installing, Upgrading, and Removing the WebLogic SPI](#)
- [Configuring the WebLogic SPI](#)
- [Customizing WebLogic SPI Policies](#)
- [Integrating HPOM Reporting and Graphing Features with the WebLogic SPI](#)
- [User-Defined Metrics](#)
- [Troubleshooting the WebLogic SPI](#)

2 Installing, Upgrading, and Removing the WebLogic SPI

Installing the WebLogic SPI

If HPOM for Windows is already installed, it is not necessary to stop the existing HPOM sessions before installing the WebLogic SPI. To install the WebLogic SPI on the management server, follow these steps:

- 1 Insert the HP Operations Smart Plug-ins DVD into the DVD drive of the management server. The HP Operations Manager InstallShield Wizard starts.
- 2 Click **Next**. The Program Maintenance window opens.
- 3 Click **Install Products**. The Product Selection window opens.
- 4 From the options listed select the **BEA WebLogic** check box and click **Next**.
- 5 Complete the installation by following the instructions that appear as you proceed. For more information see the *HP Operations Manager Smart Plug-ins DVD Installation and Upgrade Guide for Windows*.

Upgrading the WebLogic SPI



The existing WLSSPI_RPT_METRICS datasource is automatically deleted when you upgrade the WebLogic SPI from version 04.10.000 to 5.10. A new datasource is created and the existing data is lost.

The datasource will be deleted irrespective of whether you are using CODA or HP Performance Agent.

Detailed information about supported software, enhancements, fixes, and known problems and workarounds is available in the *HP Operations Smart Plug-in for BEA WebLogic Server Release Notes*, located on the HP Operations Smart Plug-ins DVD, in

`\Documentation\Releasenotes\WebLogic_AppServer_Releasenotes.html`.

Policy Changes

During the upgrade process, existing WebLogic SPI policies in the SPI for WebLogic Server policy group are saved to another policy group. If you have customized any of the WebLogic SPI policies, you must make the same customizations to the new version of the policy. You can compare your old policies with the newly installed policies and customize the new policies.

Prerequisites

- 1 Backup your existing WebLogic SPI configuration file, which is located at:
`<OvOWShareInstallDir>\SPI-Share\wasspi\wls\conf\SiteConfig`
On OVO for Windows 7.50, <OvOWShareInstallDir> can be C:\Program Files\HP
OpenView\Data\shared\

- 2 Install the new version of the WebLogic SPI by running the InstallShield Wizard (installer).

If the installer detects that an older version of the WebLogic SPI is installed, it upgrades the SPI to the new version. It performs the following tasks:

- Renames the existing SPI for WebLogic Server policy group to SPI for WebLogic Server - Saved Policies. The default policies you customized in the SPI for WebLogic Server policy group are available in the SPI for WebLogic Server - Saved Policies policy group.
- Updates the WebLogic SPI instrumentation on the management server.
- Installs new tools, policies, and a graph file on the management server.

Upgrading the WebLogic SPI

To upgrade the WebLogic SPI to version 5.10, follow these steps:

- 1 Install the WebLogic SPI version 5.10 software. See [Installing the WebLogic SPI](#) on page 11.
- 2 Refresh the SPI for WebLogic Server node group.

In the console tree, select **Tools** → **SPI for WebLogic Server** → **WebLogic Server Admin** → **Create WLSSPI Node Groups**.

All the nodes found in the WebLogic SPI service map are placed in the SPI for WebLogic Server node group.

- 3 Uninstall older versions of the modified WebLogic SPI policies (see [Policy Changes](#) on page 11) from existing nodes (versions 4.0 and earlier). This includes any customized policies you may have moved outside of the old SPI for WebLogic Server policy group and does not reside in the SPI for WebLogic Server - Saved Policies policy group.
 - a In the console tree, select **Policy management** → **Policy groups**.
 - b Select the **SPI for WebLogic Server - Saved Policies** policy group.
 - c Right-click the policies that have been modified in this version.
 - d Select **All tasks** → **Uninstall from**.
 - e Select the node group **SPI for WebLogic Server**. Select **OK**.
 - f Repeat these steps for any customized WebLogic SPI policies that do not reside in the SPI for WebLogic Server - Saved Policies policy group.
- 4 Deploy new instrumentation to the SPI for WebLogic Server node group:
 - a Right-click the **SPI for WebLogic Server** node group.
 - b Select **All Tasks** → **Deploy instrumentation**.
 - c Select **SPI for WebLogic Server** and **WLSSPI Discovery**.

- d Verify that the Remove Existing Instrumentation Before Deploying New Instrumentation check box is clear.
 - e Click **OK**.
- 5 Customize the modified policies in the WLSSPI policy group to match old customized policies. Compare old and new policies by opening them side by side.
- 6 Deploy the modified policies.
- To deploy the modified policies in the WLSSPI policy group to all WebLogic nodes, drag and drop the modified policies on the SPI for WebLogic Server node group.
- To deploy the modified policies to selected WebLogic nodes:
- a Right-click the modified policies.
 - b Select **All Tasks** → **Deploy on...**
 - c Select the nodes on which to deploy the policy group.
 - d Click **OK**.



After upgrading the WebLogic SPI, if you add an instance of BEA WebLogic Server on a managed node, you must run the Discover WebLogic tool on that node.

Removing the WebLogic SPI

To completely remove the WebLogic SPI delete all the WebLogic SPI program components and the WebLogic SPI policies.

Complete the tasks in the specified order.


Task 1: Remove All WebLogic SPI Policies from the Managed Nodes

- 1 In the console tree, select **Policy management** → **Policy groups**.
- 2 Right-click **SPI for WebLogic Server** and select **All Tasks** → **Uninstall from**. A node selection window appears.
- 3 Select the nodes on which the policies are installed.
- 4 Click **OK**.
- 5 Verify the policies are uninstalled. Check the status of the job in **Deployment jobs** under Policy groups. All WebLogic SPI policies must be uninstalled before you start the next task.

If there are customized policies (copies of WebLogic SPI default policies) residing in other HPOM policy groups, you should remove them as well.

Task 2: Remove WebLogic SPI Node Groups on the Management Server

If you created the SPI for WebLogic Server node group (by running the Create WLSSPI Node Groups tool or manually), you must remove the group.

- 1 In the console tree, select **Nodes** → **SPI for WebLogic Server**.
- 2 Open the Node Configuration editor.
 - a Select the Nodes folder in the console tree.
 - b Click the node icon  in the Configuration toolbar to open the editor. A node list appears.
- 3 Select the name of the node group you want to delete and press the **Delete** key. You can also right-click the node group and select **Delete**. The Confirm Delete window opens.
- 4 Click **Yes**.
- 5 Click **OK** to close Configure Managed Nodes window.

Task 3: Remove the WebLogic SPI Software from the Management Server

- 1 Insert the HP Operations Smart Plug-ins DVD into the DVD drive of the management server. The HP Operations Manager InstallShield Wizard starts.
- 2 From the first screen, select **Next**. The Program Maintenance window opens.
- 3 Select **Remove products**. The Product Selection window opens.
- 4 Select the **BEA WebLogic** check box and click **Next**.
- 5 Complete the removal by following the instructions that appear as you proceed.

3 Configuring the WebLogic SPI

This chapter explains how to configure the WebLogic SPI for use with HPOM. You must first complete all the configuration prerequisites. Then you must perform the basic configuration and complete additional configuration based on your environment.

Configuration Prerequisites

Complete the following tasks before configuring WebLogic SPI.

Task 1: Add Managed Nodes

For each WebLogic Administration Server and WebLogic managed server you want to manage from HPOM, ensure that all nodes on which the WebLogic servers are running are configured in HPOM, as managed nodes.

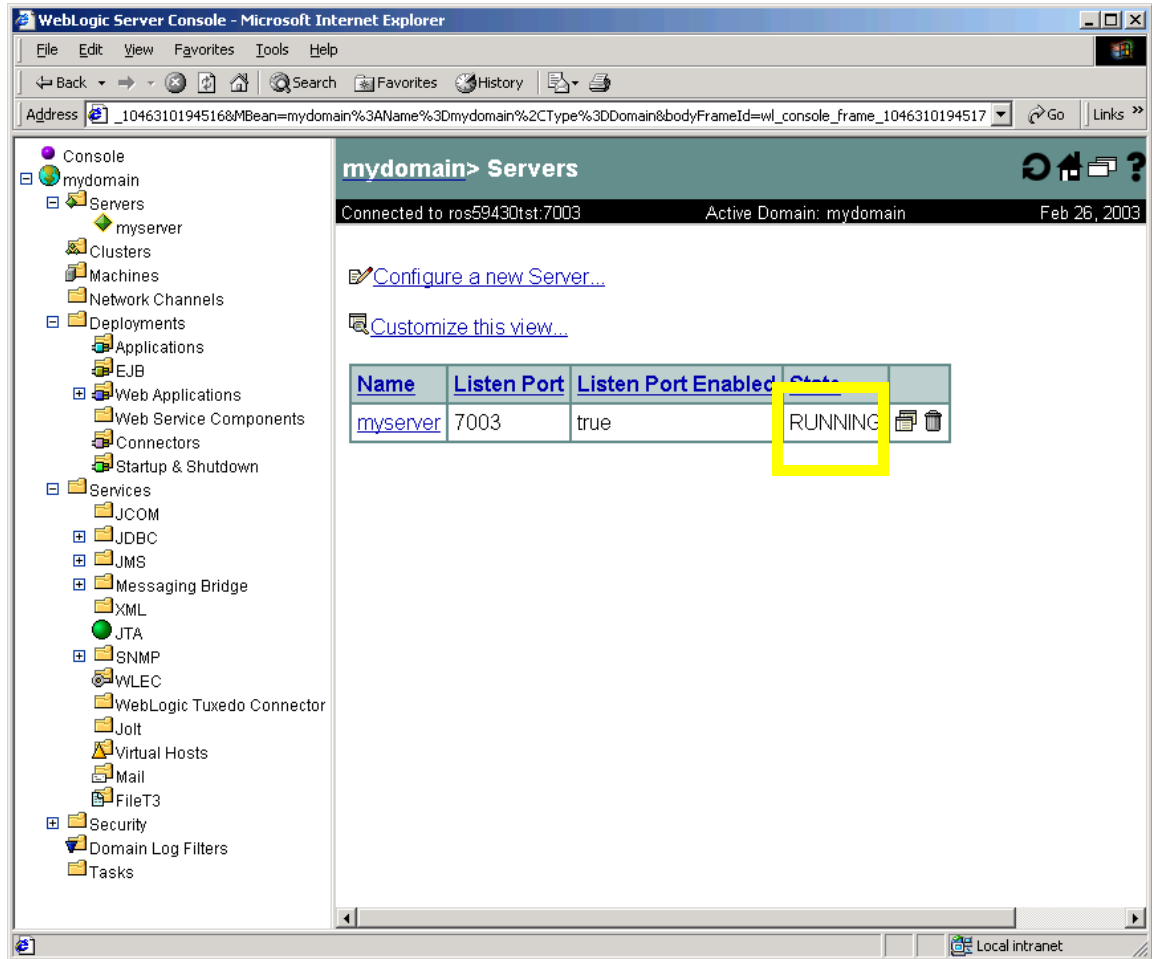
To add a UNIX managed node, follow these steps:

- 1 Install the HP Operations agent on the node. For more information, see the topic Agent Installation on UNIX computers in the HPOM console online help.
- 2 Specify each WebLogic Server node on UNIX to be managed. For more information, see the topic Configure Managed Nodes in the HPOM console online help.

To add a Windows managed node, specify each WebLogic Server node on Windows to be managed. For more information, see the topic Configure Managed Nodes in the HPOM console online help. The HP Operations agent is automatically installed when you complete this step.

Task 2: Verify the Application Server Status

You can verify if your application servers are running by checking the status of the server in the WebLogic administrative console. The WebLogic SPI discovery policies only discover application servers that are running.



Task 3: Collect WebLogic Login Information

Collect the WebLogic login and password for each WebLogic Administration Server.

If you do not want to use the existing login and password, create new ones. The WebLogic SPI discovery process uses the login and password to gather basic configuration information and the WebLogic SPI data collector uses the login and password to collect metrics.

Configuration of the WebLogic SPI is simplified if the login and password to access all WebLogic Administration Servers are the same.



If an instance of WebLogic Server has a server login name and password different from the default login and password, you must explicitly configure the login details for that server using the configuration editor before you launch the discovery tool. For more information about the discovery tool and configuration editor, see the WebLogic SPI online help.

WebLogic Server Version 6.x

If you are running WebLogic Server version 6.x, you can use the default administration login “system.”

To configure a different login, follow these steps:

- 1 In the WebLogic Server administration console, create the login and password that you plan to use for the WebLogic SPI, if it does not already exist.
- 2 Create an ACL named `weblogic.admin.mbean` if it does not already exist.
- 3 Create a permission named `access` for this ACL if it does not already exist.
- 4 Add the login (created in step 1) to the list of users for the `access` permission (created in step 3).

WebLogic Server Version 7.0 and Higher

If you are running WebLogic Server version 7.x or higher, you can use the administration login that was configured when you installed the WebLogic Server or you can use a user that belongs to the WebLogic Administrators or Monitors group.

To configure a user that belongs to the Administrators or Monitors group, use the WebLogic administration console. For more information about creating a user and assigning a user to a group, see the section, “Users and Groups” of the *Securing WebLogic Resources* manual (http://e-docs.bea.com/wls/docs70/secwlrres/usrs_grps.html or http://e-docs.bea.com/wls/docs81/secwlrres/usrs_grps.html).

- ▶ A user that belongs to the Monitors group cannot use the Start/Stop WebLogic tools (to start or stop WebLogic Servers from the HPOM console) or perform the JMX call “set” when implementing JMX actions (to assign a value to a specified attribute if you are creating UDMs).

Task 4: Collect Names of HPOM Managed Nodes

Collect the names of the HPOM managed nodes on which the WebLogic Administration Servers are running. You must select these managed nodes when you configure WebLogic SPI.

- ▶ You do *not* need to collect the names of the HPOM managed nodes on which only a WebLogic managed server is installed. On these managed nodes, as long as the WebLogic managed server is running, the WebLogic managed server is automatically discovered when you complete the basic WebLogic SPI configuration.

Basic WebLogic SPI Configuration

To complete basic WebLogic SPI configuration for WebLogic servers running on HTTP mode, complete the following tasks. For information about configuring WebLogic SPI for WebLogic servers running on HTTPS mode see [Configuring WebLogic SPI for WebLogic Servers Running on HTTPS](#) on page 24.

Configuration Prerequisite

Before launching the Discover WebLogic tool, deploy the following instrumentation files on the managed nodes:

- SHS Data Collector
- SPI Data Collector
- SPI for WebLogic Server
- WLSSPI Discovery

To deploy these instrumentation files, follow these steps:

- 1 From the HPOM console select **Operations Manager** → **Nodes**.
- 2 Right-click the managed node on which you want to run Discover WebLogic tool.
- 3 Select **All Tasks** → **Deploy instrumentation**. The Deploy Instrumentation window opens.
- 4 Select SHS Data Collector, SPI Data Collector, SPI for WebLogic Server, and WLSSPI Discovery from the list of instrumentation files and click **OK**.

To verify that these files deployed successfully, check Deployment Jobs under Policy management. There must be no error messages.

Task 1: Run Discover WebLogic

Discover WebLogic sets basic configuration properties needed for discovery, deploys the WebLogic SPI discovery policies, and updates the service map.

To run Discover WebLogic, follow these steps:

- 1 From the HPOM console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **SPI Admin**.
- 2 Double-click **Discover WebLogic**. The Edit Parameters window opens.
- 3 Select the managed nodes on which the WebLogic Administration Servers are running and click **Launch**.

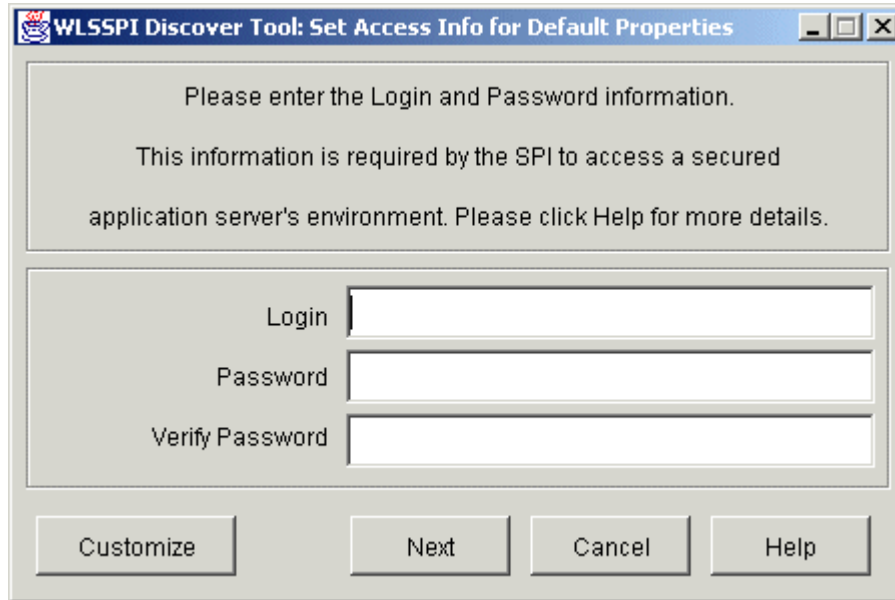
The Console Status window opens. After a few seconds the Introduction window opens. This window contains brief information about the Discover WebLogic tool.

- 4 Click **Next**.

A second Introduction window opens. This window contains information about the properties that may be required for the discovery process to work.

- 5 Click **Next**.

- 6 If you did not set the WebLogic SPI LOGIN and PASSWORD properties, the Set Access Info for Default Properties window opens.
 - ▶ If you already set the LOGIN and PASSWORD properties, the configuration editor opens. Go to step 7.



The screenshot shows a dialog box titled "WLS SPI Discover Tool: Set Access Info for Default Properties". The text inside reads: "Please enter the Login and Password information. This information is required by the SPI to access a secured application server's environment. Please click Help for more details." Below the text are three input fields labeled "Login", "Password", and "Verify Password". At the bottom of the dialog are four buttons: "Customize", "Next", "Cancel", and "Help".

Set the LOGIN and PASSWORD properties to the WebLogic login and password collected in [Task 3: Collect WebLogic Login Information](#) on page 16. You must set the LOGIN and PASSWORD even if you are using the default login and password (WebLogic version 5.1 and 6.x) or the login and password configured during the WebLogic Server installation (WebLogic version 7.0 or higher).

The LOGIN and PASSWORD properties set in this window are used as the default WebLogic login and password (they are set at the global properties level). If no NODE level or server-specific LOGIN and PASSWORD properties are set, this WebLogic login and password are used by WebLogic SPI to access all WebLogic Administration Servers. For more information about the configuration structure, see the topic [The Configuration](#) in the WebLogic SPI online help.

If the WebLogic Administration Server login and password are the same for all instances of WebLogic on all HPOM managed nodes, follow these steps:


- a Set the LOGIN and PASSWORD in the Set Access Info for Default Properties window.
- b Click **Next**.
- c Go to step 8.

If the WebLogic Administration Server login and password are different for each managed node but are the same for all instances of the WebLogic Administration Server on each managed node, you must customize the WebLogic SPI configuration by setting the LOGIN and PASSWORD properties at the NODE level (for more information about the configuration structure, see the topic [Configuration editor operation](#) in the WebLogic SPI online help).

- a Set LOGIN and PASSWORD to the most commonly used WebLogic login and password in the Set Access Info for Default Properties window.

- b Click **Customize** to start the configuration editor and set the LOGIN and PASSWORD properties at the NODE level.

If the WebLogic Administration Server login and password are different for each managed node and they are different for the instances of the WebLogic Administration Server on a managed node, you must customize the WebLogic SPI configuration by setting the LOGIN, PASSWORD, NAME, and PORT properties at the server-specific level. For more information about the configuration structure, see the topic The configuration in the WebLogic SPI online help.

- a Set LOGIN and PASSWORD to the most commonly used WebLogic login and password in the Set Access Info for Default Properties window.
 - b Click **Customize** to start the configuration editor and set the LOGIN, PASSWORD, NAME, and PORT properties at the server-specific level.
- 7 From the configuration editor, set the properties. For more information about using the configuration editor, see the WebLogic SPI online help.
 - 8 Click **Next** to save changes and exit the editor. The Confirm Operation window opens.
 - 9 Click **OK**. The discovery policies are deployed to the selected managed nodes.
 -  If you click Cancel, the discovery policies are not deployed. However, if you made changes to the configuration, those changes remain in the configuration on the management server. To make the changes to the selected managed nodes' configuration, you must launch the Discover WebLogic tool, select those managed nodes, Click **Next** in the configuration editor, and then click **OK**.
 - 10 Check the Console Status window for error messages. If none appear click **Close**.

If the window displays an error message, see [Troubleshooting the Discovery Process](#) on page 89 to diagnose and troubleshoot.

Task 2: Verify the Discovery Process

Depending on the number of managed nodes in your environment, verification may take several minutes to complete.

To verify if the discovery process is successfully completed, follow these steps:

- 1 Check if the following message appears in the message browser of the managed node:

```
Updating WLS SPI configuration in HPOM server for <node>
```

Verify that the following message appears in the message browser of the management server:

```
The SPI configuration for <node> was updated by discovery in the HPOM server. The updated configuration is as shown below
```

Depending on the number of managed nodes in your environment, it may take several minutes for these messages to appear for all managed nodes.

If these messages are present, the WLSSPI Discovery policies are successfully deployed.

If these messages are not present, go to [Task 3: Set Additional Properties](#) on page 22.

- 2 From the HPOM console, select **Operations Manager** → **Services** → **Applications** → **WebLogic**. The service map appears. It may take some time for the service map to appear completely.

Task 3: Set Additional Properties

LOGIN and PASSWORD are the basic properties needed by the discover process. However, depending on your environment, additional configuration properties may be needed by the discovery process. For a complete definitions of the properties, see the WebLogic SPI online help.

Properties	Description	When to Set
JAVA_HOME	The default directory where Java is installed. Use the highest version of Java available. If you are running WebLogic Server version 9.x, you must use Java version 1.4.1 or higher.	You must set the JAVA_HOME property in the following scenarios: <ul style="list-style-type: none">• If multiple versions of the WebLogic Server are installed on a node.• If more than one version of Java is installed.• If you did not use BEA's installation scripts to install the WebLogic Server and service packs.
HOME_LIST	A list of directories where the WebLogic Server is installed.	You must set the HOME_LIST property in the following scenarios: <ul style="list-style-type: none">• If you did not use BEA's installation scripts to install the WebLogic Server and service packs.• If the BEA registry.xml file is not accurate or cannot be found.
ADDRESS	The domain name or IP address where the WebLogic Server is listening.	If WebLogic Server is configured to a virtual IP address.

Properties	Description	When to Set
NODE_NAMES	The virtual IP address where the server is listening. If not set, the remote WebLogic Server is not discovered.	If a remote WebLogic Server is listening on a virtual IP.
ADMIN_PORTS	<p>The port number of WebLogic Admin server. The domain configuration file (<code>config.xml</code>) may be located in the default directory:</p> <ul style="list-style-type: none"> • <code><WebLogic_Install_Dir>\config\<WebLogic_Domain>\</code> (WebLogic 6.x) • <code><BEA_Home_Dir>\user_projects\domains\<WebLogic_Domain>\</code> (WebLogic 7.0 SP2 or higher) <p>Where, <code><WebLogic_Install_Dir></code> is the directory where the WebLogic Server is installed <code><BEA_Home_Dir></code> is the directory that contains the <code>registry.xml</code> file <code><WebLogic_Domain></code> is the WebLogic domain name.</p>	If the WebLogic Server's domain configuration file is not located in the default directory.
EXCLUDE_SAMPLES	When you set this property to "true," the WebLogic Server sample programs are not included in the discovery process. It is recommended that you set this to "true" at the default properties level to reduce the amount of time it takes for the discovery process to run. The discovery process can take several minutes to complete.	To reduce the amount of time for the discover process to run.



When you launch the Discover WebLogic tool, `JAVA_HOME` takes the default value. If you specified a different value for `JAVA_HOME` for any instance of WebLogic server, you must explicitly set that value for `JAVA_HOME` using the configuration editor. For more information about the `JAVA_HOME` property and configuration editor, see the WebLogic SPI online help.

To set one or more of these properties, follow these steps:

- 1 Repeat [Task 1: Run Discover WebLogic](#) on page 18. When you get to step 7, set one or more of the properties listed above.
- 2 Repeat [Task 2: Verify the Discovery Process](#) on page 20.
- 3 If verification is successful, go to [Additional WebLogic SPI Configuration](#) on page 25.

If verification is not successful, view the error messages in the message browser and follow the instruction text to correct the problem.

Configuring WebLogic SPI for WebLogic Servers Running on HTTPS

If the WebLogic admin server is running on t3s (HTTPS) and the WebLogic servers associated with it are running on t3 (HTTP), follow these steps:

- 1 From the HPOM console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **SPI Admin**.
- 2 Double-click **Discover WebLogic**.
- 3 Select the managed nodes on which the WebLogic Administration Servers are running.
- 4 Click **Launch**. The Console Status window opens. After a few seconds the Introduction window opens. This window contains brief information about the Discover WebLogic tool.
- 5 Click **Next**. A second Introduction window opens. This window contains information on the properties that may be required for the discovery process to function.
- 6 Click **Next**. The Set Access Info for Default Properties window opens.
- 7 Set the BEA WebLogic LOGIN and PASSWORD properties.
- 8 Set the ADMIN_PORTS and PROTOCOL property. The default value for PROTOCOL is **t3s**.
- 9 ADMIN_PORTS is the SSL port on which the application server is listening. The PROTOCOL property specifies if the application server port is using SSL or non-SSL.
- 10 If required, set the PASSPHRASE and KEYSTORE properties. Click **Next** to run discovery on the selected nodes.

KEYSTORE is the path to the SSL trust keystore file.

PASSPHRASE is the password that you set for the KEYSTORE in the SSL environment of the WebLogic Admin server.

For more information about setting the properties see the WebLogic SPI online help.



The properties KEYSTORE, PASSPHRASE, and PROTOCOL can be set at any level (global, group, node, or server). PROTOCOL is required if you use SSL. You must set KEYSTORE and PASSPHRASE only if you use a keystore and passphrase in your SSL environment.

After Discovery is successful:

- 1 From the HPOM console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **SPI Admin**.
- 2 Double-click **Configure WebLogic**.
- 3 Select the new servers that are discovered.
- 4 Click **Launch**. The Console Status window opens. After a few seconds the WLSSPI Configure Tool: Introduction window opens. This window contains information about the Configure tool.
- 5 Click **Next**. The Configuration Editor opens.
- 6 Set the value of the PROTOCOL property to t3 for the WebLogic servers. If you do not change the value of PROTOCOL property for the servers it will take up the default value (t3s) set for the Admin Server. For more information about configuration property precedence see the section Configuring WebLogic SPI in the WebLogic SPI online help.
- 7 Create a MONITOR USER in BEA under the active Security Realm
- 8 Use the newly created user credentials in the SERVER properties (set the LOGIN and PASSWORD properties of the server similar to the values set for Monitor User).

Repeat the steps mentioned above for every instance of WebLogic Server.

If the WebLogic admin server as well as the WebLogic servers associated with it are running on t3s (HTTPS) you can configure the admin server and the WebLogic servers as mentioned in the previous scenario. However, you must not set the PROTOCOL property to t3 because the servers are running on the same mode as the admin server. The PROTOCOL property must have the default value t3s only.

Additional WebLogic SPI Configuration

After you successfully complete the basic WebLogic SPI configuration, you must finish WebLogic SPI configuration by setting the properties that are not automatically discovered by the Discovery policies and install and configure additional components. Setting some of these properties and configuring additional components depends on your environment.

Properties	When to Set
START_CMD and STOP_CMD	To run the Start WebLogic and Stop WebLogic tools from the HPOM console.
LAUNCH_DIR	If absolute log file names are not configured in the WebLogic Server or when the WebLogic Server's installation and starting directories are in different locations (WebLogic 6.x only)

- If you are configuring user-defined metrics, see the JMX Metric Builder release notes for additional installation information and the JMX Metric Builder online help for additional configuration information.
- If you have installed HP Performance Manager (must be purchased separately), see [Integrating the WebLogic SPI with HP Performance Manager](#) on page 65 for additional installation and configuration information.
- If HP Reporter is installed (must be purchased separately), see [Integrating the WebLogic SPI with HP Reporter](#) on page 58 for installation and configuration information.

To update the configuration, follow these steps:

- 1 From the HPOM console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **SPI Admin**.
- 2 Double-click **Configure WLSSPI**. The Edit Parameter window opens.
- 3 Select the managed nodes to configure.
- 4 Click **Launch**. The Console Status window and then the Introduction window opens.
- 5 Click **Next**. The configuration editor opens.
- 6 Set the properties.
- 7 Click **Next** to save and exit the editor.

For a complete description of the WebLogic SPI properties and information about setting the properties using the configuration editor, see the section Configuration properties in the WebLogic SPI online help.

WebLogic SPI in High Availability Environments

High availability is a general term used to characterize environments that are business critical and therefore are protected against downtime through redundant resources. Very often, cluster systems are used to reach high availability.

You can configure WebLogic SPI to accommodate cluster environments where failovers allow uninterrupted WLS availability. WebLogic SPI monitoring, when synchronized with the cluster environment, can switch off from the failed node to the active node.

Configuration Prerequisites

The prerequisites for using WebLogic SPI in high availability environments are:

- Management Server: HPOM for Windows 8.00 or OVO for Windows 7.50
- Node: HP-UX MCSG cluster
- HPOM 8.x HTTPS and DCE Agent version (for details see Agent cluster support matrix)

Configuring WebLogic SPI for High Availability Environments

To configure WebLogic SPI for use in high availability environments complete the following tasks:

Task 1: Create the WebLogic SPI monitoring configuration file

WebLogic SPI uses a monitoring configuration file `<appl_name>.apm.xml` that works in conjunction with the clustered application configuration file.



`<appl_name>` is the namespace_name. For more information, see *HP Operations Manager for UNIX HTTPS Agent Concepts and Configuration Guide*.

The `<appl_name>.apm.xml` file lists all the WebLogic SPI templates on the managed node so that you can disable or enable these templates as appropriate, for inactive and active managed nodes.

To create this clustered application configuration file for your WLS environment, follow these steps:

- 1 Use the following syntax to create the `<appl_name>.apm.xml` file:

```
<?xml version="1.0"?>
<APMAApplicationConfiguration>
  <Application>
    <Name> ... </Name>
    <Template> ... </Template>
    <StartCommand>wasspi_wls_perl -S wasspi_wls_clusterSvrApp -opt
startMonitor $instance</StartCommand>
    <StopCommand>wasspi_wls_perl -S wasspi_wls_clusterSvrApp -opt
stopMonitor $instance</StopCommand>
  </Application>
</APMAApplicationConfiguration>
```

- 2 Enter the namespace_name within the `<Name></Name>` tag.

- 3 After the file is created, save it in the `$OvDataDir/bin/instrumentation` directory for DCE agent. For HTTPS agent save it in the `$OvDataDir/bin/instrumentation/conf` directory.

Sample `<appl_name>.apm.xml` file

```
<?xml version="1.0"?>
<APMApplicationConfiguration>
  <Application>
    <Name>wlsspi</Name>
    <Template>WLSSPI Error Log</Template>
    <Template>WebLogic Logs</Template>
    <Template>WLSSPI-05min</Template>
    <Template>WLSSPI-15min</Template>
    <Template>WLSSPI-1h</Template>
    <StartCommand>wasspi_wls_perl -S wasspi_wls_clusterSvrApp -opt
startMonitor $instance</StartCommand>
    <StopCommand>wasspi_wls_perl -S wasspi_wls_clusterSvrApp -opt
stopMonitor $instance</StopCommand>
  </Application>
</APMApplicationConfiguration>
```

To prevent the agent from running the policies on a passive node, you must mention the policy names within the `<template></template>` tag.



`<appl_name>.apm.xml` is dependent on the application namespace. It is not dependent on the instance level. Therefore, the start and stop actions are provided with the associated instance name as their first parameter when they are executed at package switch time. The environment variable `$instanceName` is set by CIAW when start or stop tasks are performed.

Task 2: Create the clustered application configuration file

The clustered application configuration file `apminfo.xml`, working in conjunction with the `<appl_name>.apm.xml` file of WebLogic SPI, allows you to associate WebLogic SPI monitored instances with cluster resource groups. As a result, when you move a resource group from one node to another, in the same cluster, monitoring stops on the failed node and starts on the new node.

To create the clustered application configuration file `apminfo.xml` follow these steps:

- 1 Use a text editor to create the file. The syntax is:

```
<?xml version="1.0" ?>
<APMClusterConfiguration>
  <Application>
    <Name>namespace_name</Name>
    <Instance>
      <Name><Instance Name></Name>
      <Package><Package Name></Package>
    </Instance>
  </Application>
</APMClusterConfiguration>
```

- 2 Enter `namespace_name` within the `<Name></Name>` tag.

- 3 Save the `apminfo.xml` file in the `$OvDataDir/conf/conf` directory for HTTPS Agent. For DCE Agent, save the `apminfo.xml` file in the `$OvDataDir/conf/OpC` directory.

Sample `apminfo.xml` file

```
<?xml version="1.0" ?>
<APMClusterConfiguration>
  <Application>
    <Name>namespace_name</Name>
    <Instance>
      <Name>instance_name</Name>
      <Package>test</Package>
    </Instance>
  </Application>
</APMClusterConfiguration>
```

Task 3: Configure WebLogic SPI for HTTPS or DCE Agent (Based on Requirement)

To configure WebLogic SPI for HTTPS or DCE agent, follow these steps:

- 1 Deploy instrumentation files and policies on the target cluster nodes.
- 2 Launch the Discover WebLogic tool with active cluster node as target. For details about launching the discovery tool, see the WebLogic SPI online help.
- 3 Launch the Configure WLSSPI tool with the active cluster node as target. The configuration editor opens.
- 4 Copy the `SiteConfig` file from active node to passive node. The file is located in the `$OvDataDir/wasspi/wls/conf` directory for DCE agent and in the `$OvDataDir/conf/wlsspi` directory for HTTPS agent.

Additional Discovery and Configuration Scenarios

This section provides information on how to run discovery under different setups. It includes examples about some of the common scenarios.

Use Case 1: Administration Port Turned On (WebLogic Servers are Running in HTTPS Mode)

If the Administration Port is enabled for the WebLogic Servers, there are two discovery scenarios:

Scenario 1: The WebLogic admin server is running on t3s (HTTPS) and the WebLogic servers associated with it are running on t3 (HTTP)

Discovery

- 1 Launch Discover WebLogic. See [Task 1: Run Discover WebLogic](#) on page 18.
- 2 In the Set Access Info for Default Properties set the BEA WebLogic LOGIN and PASSWORD properties (this window will appear *only if* you have not set the LOGIN and PASSWORD earlier).
- 3 In the configuration editor, set the ADMIN_PORTS property. ADMIN_PORTS is the SSL port on which the application server is listening.
 - ▶ If the WebLogic Admin server is running on a virtual IP (in a non-clustered environment), you must specify the virtual IP address when setting the ADMIN_PORTS property. Set ADMIN_PORTS value to *<ip address>*:port. For information about configuration in clustered environment see [WebLogic SPI in High Availability Environments](#) on page 26.
- 4 Set the PROTOCOL property to **t3s** (t3s is the default value for PROTOCOL). PROTOCOL specifies if the application server port is using SSL or non-SSL.
- 5 If required, set the PASSPHRASE and KEYSTORE properties. Click **Next** to run discovery on the selected nodes.

KEYSTORE is the path to the SSL trust keystore file.

PASSPHRASE is the password that you set for the KEYSTORE in the SSL environment of the WebLogic Admin server.

For more information about setting the properties see the section on configuration properties in the WebLogic SPI online help.

- ▶ The properties KEYSTORE, PASSPHRASE, and PROTOCOL can be set at any level (global, group, node, or server). PROTOCOL is required if you use SSL. You must set KEYSTORE and PASSPHRASE only if you use a keystore and passphrase in your SSL environment.

Configuration

After Discovery is successful:

- 1 Launch the Configure WLSSPI tool (For instructions on launching this tool see the section Configure WLSSPI in the WebLogic SPI online help).
- 2 In the configuration editor set the value of the PROTOCOL property to t3 for the WebLogic servers.
 - ▶ If you do not change the value of PROTOCOL property for the WebLogic servers to t3, PROTOCOL will take up the default value (t3s) set for the Admin Server. For more information about configuration property precedence, see the section on configuration properties in the WebLogic SPI online help.
- 3 Create a Monitor user in BEA under the active Security Realm.
- 4 Set the SERVER_LOGIN and SERVER_PASSWORD properties similar to the credentials set for the Monitor user.
- 5 Repeat steps 1 to 4 for every instance of WebLogic Server.

Scenario 2: The WebLogic admin server as well as the WebLogic servers associated with it are running on t3s (HTTPS):

- 1 Run Discover WebLogic and set the properties as mentioned in Scenario 1.
- 2 After Discovery is successful.
 - a Launch the Configure WLSSPI tool (For instructions on launching this tool, see Configure WLSSPI in the WebLogic SPI online help).
 - b Set the PROTOCOL property to the default value t3s.
 - c Create a Monitor user in BEA under the active Security Realm.
 - d Set the SERVER_LOGIN and SERVER_PASSWORD properties similar to the credentials set for the Monitor user.
 - e Repeat steps a to d for every instance of WebLogic Server.

Use Case 2: Administration Port Not Turned On (WebLogic Server is Running on Virtual IP)

If the WebLogic Server is running on a virtual IP and the Administration Port is not turned on, follow these steps to run discovery:

- 1 Launch Discover WebLogic. See [Task 1: Run Discover WebLogic](#) on page 18.
- 2 In the Set Access Info for Default Properties set the BEA WebLogic LOGIN and PASSWORD properties (this window will appear *only if* you have not set the LOGIN and PASSWORD earlier).
- 3 In the configuration editor, set the NODE_NAMES and ADDRESS properties.
- 4 Click **Next**. The Confirm Operation window opens.

Click **OK** to run the discovery on the selected managed nodes.

▶ For information about configuration in clustered environment see [WebLogic SPI in High Availability Environments](#) on page 26.

4 Customizing WebLogic SPI Policies

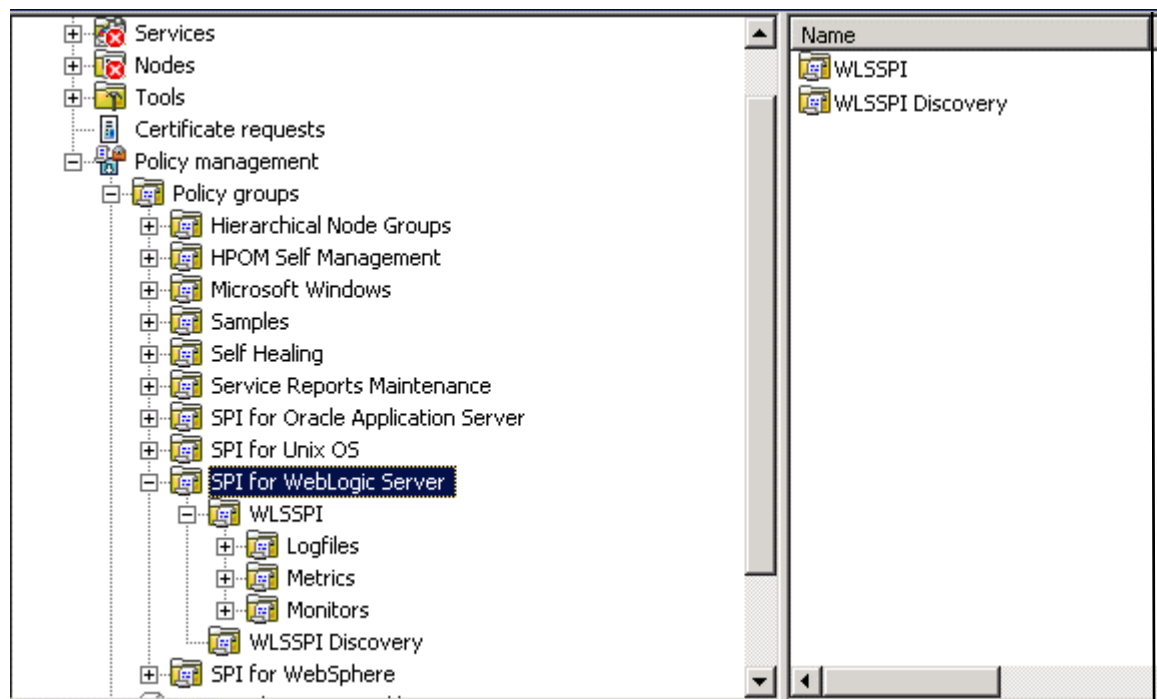
WebLogic SPI policies help you monitor the BEA WebLogic Servers. You can customize these policies depending on the requirements of your IT environment. This chapter includes general guidelines about the WebLogic SPI policies and explains how you can customize them. For more information, see the Policies section in the WebLogic SPI online help.

WebLogic SPI Policy Groups and Types

You can customize WebLogic SPI policies to suit the needs of your IT environment. However, these policies can also work without any modifications.

WebLogic SPI Policy Groups

The WebLogic SPI policies are organized under the top-level - SPI for WebLogic Server policy group (as shown in the following figure.)



The WebLogic policy group contains the following metric and logfile policies:

- **Metric policies:** generate messages according to threshold settings that monitor WebLogic availability and performance metrics.

- **Monitor policies:** pertain to all metrics scheduled to be collected in the specified collection interval (grouped according to collection intervals).
- **Logfile policies:** generate messages according to logfile and error text detected in both the WebLogic Server logfiles and in the WebLogic SPI logfiles.

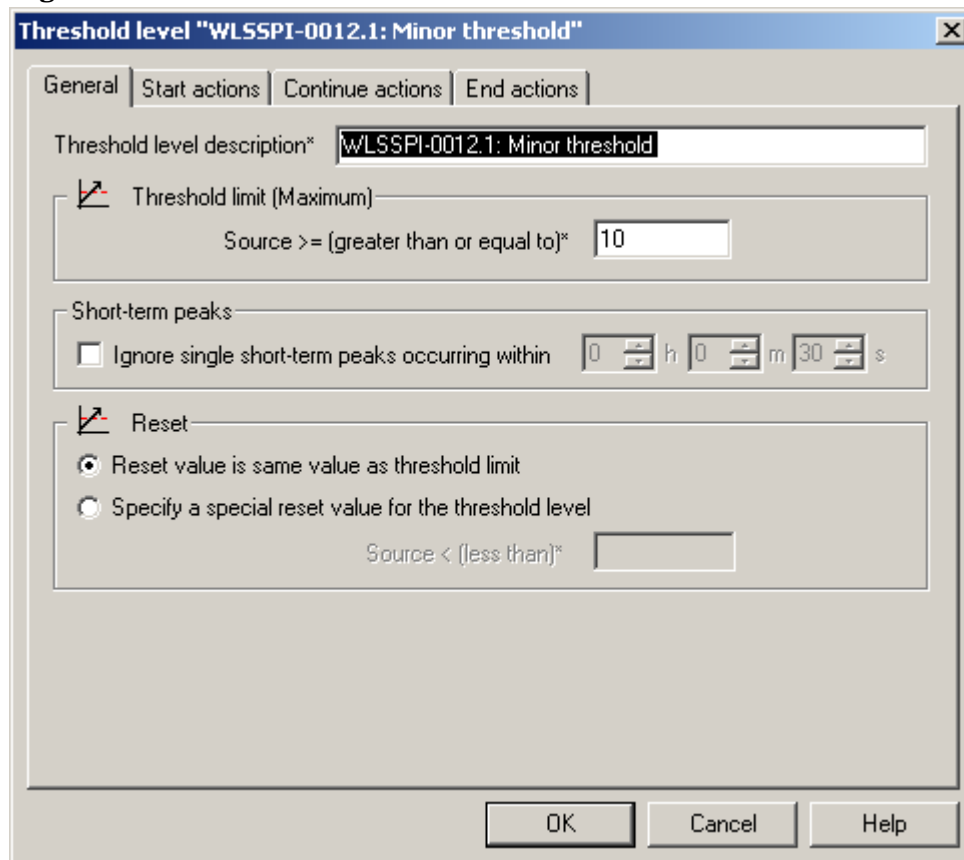
The WLSSPI Discovery policy automatically updates the WebLogic configuration information in the service map. Clear the AUTO_DISCOVER check box in the configuration editor if you do not want the discovery policy to automatically overwrite this configuration information. For information on using the configuration editor, see the WebLogic SPI online help.

WebLogic SPI Policy Types

Metric policies define how data is collected for the individual metric and set a threshold value that, when exceeded, generate alerts/messages in the Message Browser. You can change the threshold within a policy by double-clicking the policy, clicking on the **Threshold levels** tab, and clicking on **Threshold level** in the Level summary pane.

Incoming values for metric WLSSPI-0012.1 are compared against its threshold limits. In the following illustration, the default threshold is set at 10.

Figure 1 Threshold Level Window



Collector policies define all metrics for the WebLogic Server application that are scheduled for collection at the specified interval. Within the name of each collector policy is its collection interval (for example, WLSSPI-1h, where the collection interval is one hour). When you open any collector policy, you see all metrics (by number) collected within the interval following the `-m` option of the collector/analyzer command `wasspi_wls_ca`.

Basic Policy Customizations

This section covers basic policy customizations like changing threshold values, scheduling or deleting a metric from data collection, opening a metric policy or collector policy and so on.

Before you begin to customize any of the policies, you must make copies of the original policies so that the default policies remain intact.

Modifying Metric Policies

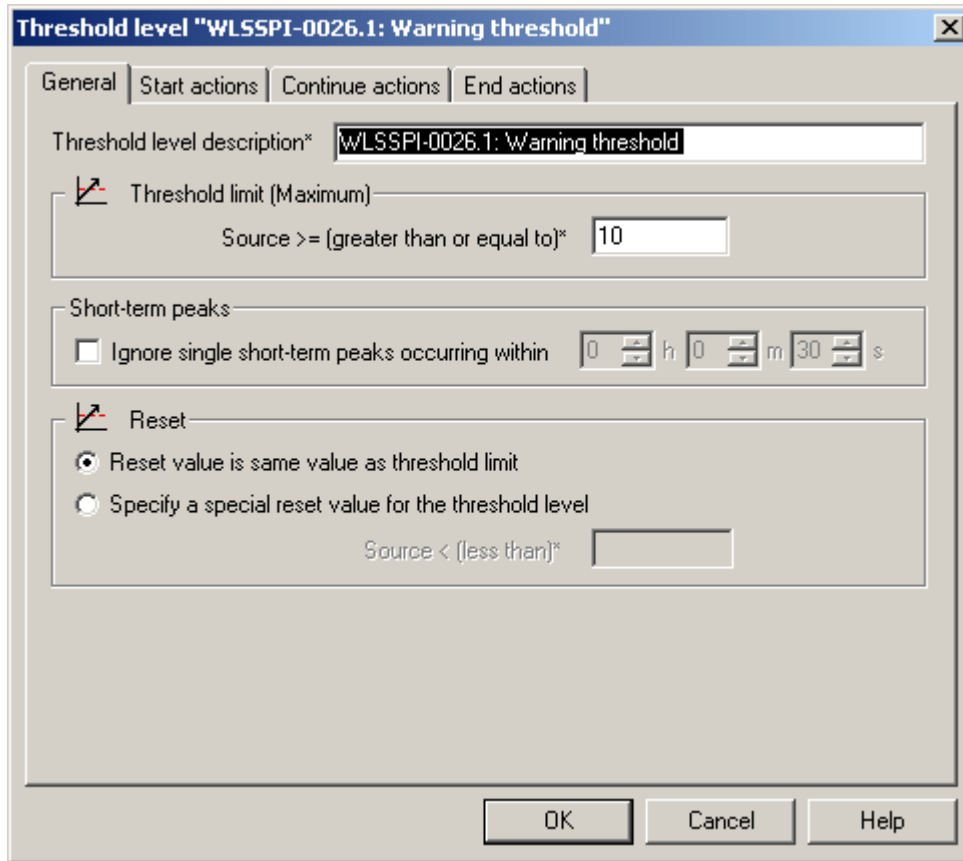
You can modify the metric attributes for all monitored instances of WebLogic Server. Some of these attributes are explained in the Configuration Properties section in the WebLogic SPI online help.

Threshold Level and Actions

To modify the threshold level and actions of a policy, follow these steps:

- 1 From the HPOM console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WLSSPI** → **Metrics**.
- 2 Double-click the policy for which you want to change the threshold value. The policy window opens.
- 3 Select the **Threshold levels** tab.
- 4 From the Level summary pane, click **Threshold level**. The Threshold Level window opens.

You can modify the metric attributes from this window.

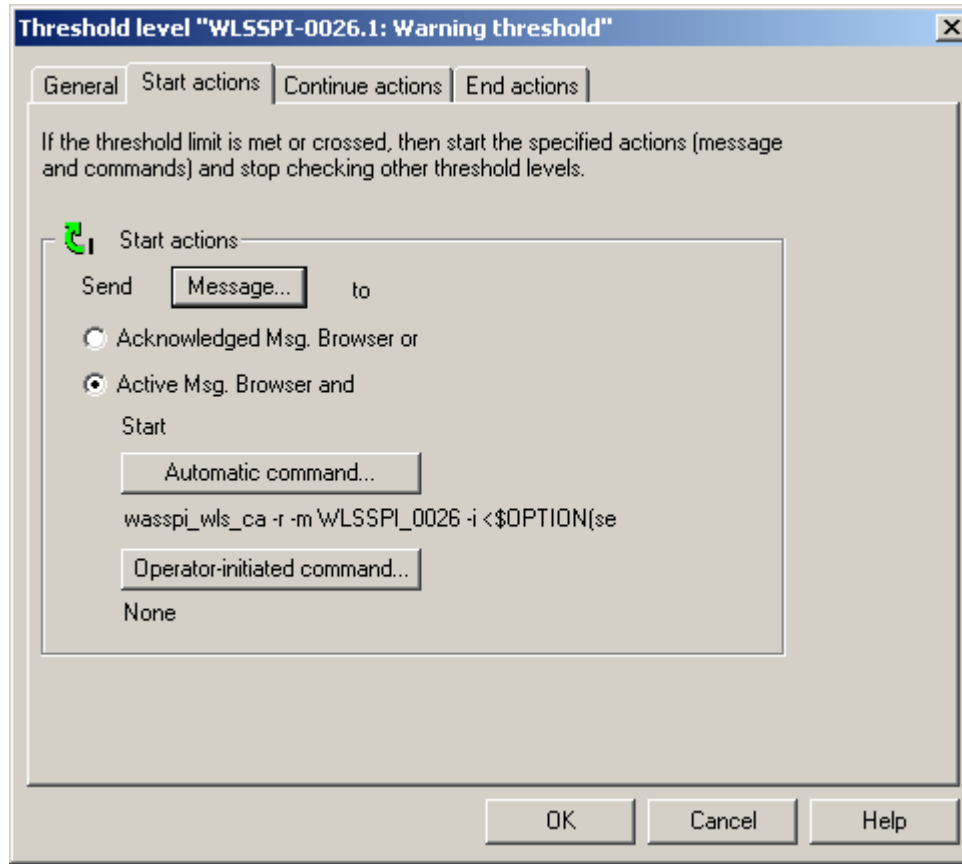


The figure shows the threshold limit is set to 10 for WLSSPI-0026. The incoming values for this metric show the total number of times per minute clients must wait for an available (Enterprise Java) bean. After the value exceeds 10, the server response time gets affected. This generates a warning message.

The following metric attributes can be modified:

- *Threshold limit.* The value that triggers a message if it is met or crossed.
- *Short-term peaks.* A minimum time period over which the monitored value must exceed the threshold before generating a message. For a message to be sent, the value must be greater than the threshold each time the value is measured during a duration that you select. If the duration is set to 0 or the box is left empty, an alarm is generated as soon as HPOM detects that the threshold has been equaled or crossed.
- *Reset.* A limit below which the monitored value must drop (or exceed, for minimum thresholds) to return the status of the monitored object to normal.

As the following illustration shows, the Threshold Level window has three action tabs. You can click any of the action tabs to set the related action.



- *Start actions.* Actions carried out the first time that the threshold is crossed.
- *Continue actions.* Actions carried out at each subsequent polling interval if the reset value is not reached.
- *End actions.* Actions carried out after the threshold crosses the reset value.

In each of the actions tabs, you can set the type of actions to perform.

The WebLogic SPI lets you generate Performance Manager graphs or reports, or to add custom programs. You can generate the reports and graphs through:

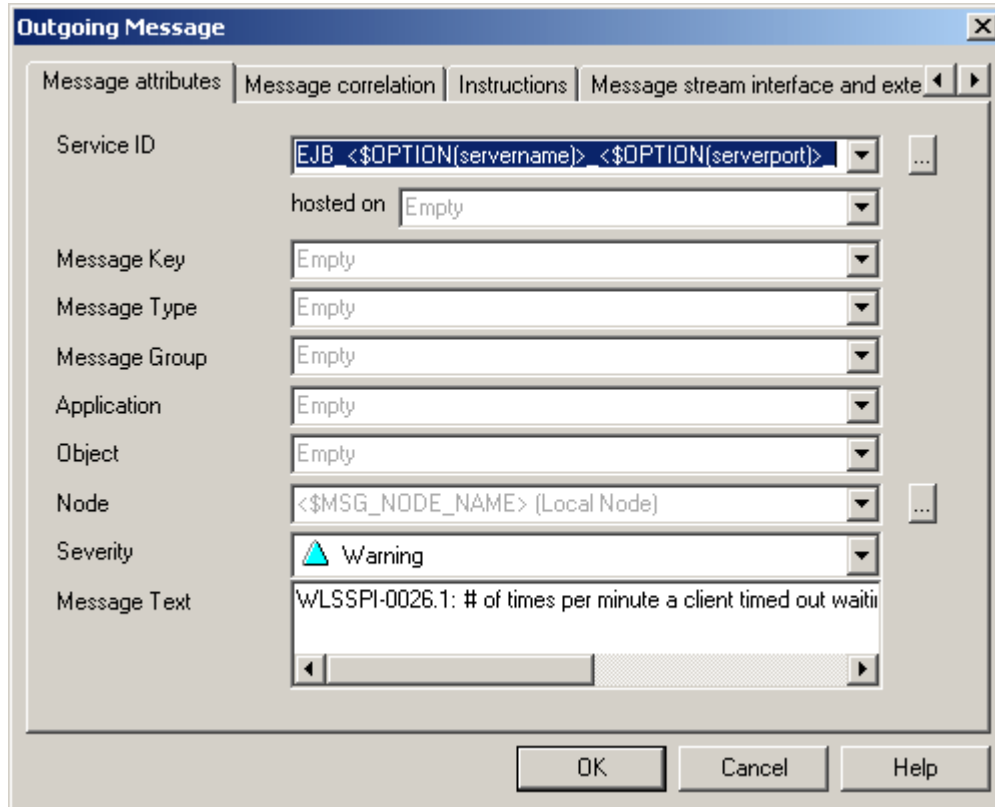
- *Automatic command.* A command run when the rule is matched. The automatic command delivered with the WebLogic SPI generates a snapshot annotations report that shows the data values at the time the action was triggered from an exceeded threshold. You can view the report in the message annotations.
- *Operator-initiated command.* A command attached to the message that the rule sends to the message browser. You can run this command from the message browser. The operator-initiated command delivered with the WebLogic SPI lets you click the **Perform Action** button to view a graph of the metric whose exceeded threshold generated the message, along with other related metric values.

Message and Severity

To modify the message and severity of a policy, follow these steps:

- 1 From the HPOM console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WLSSPI** → **Metrics**.

- 2 Double-click the policy for which you want to modify the severity and message text. The Measurement Threshold window opens.
- 3 Double-click the threshold level description. A new window opens. Click the **Start Actions** tab.
- 4 Click **Message**. The Outgoing Message window opens.



You can modify the following attributes:

- *Severity*. Indicates the importance of the event that triggers this message.
 - *Message Text*. You can modify the text of the message but do *not* modify any of the parameters—beginning with \$ and surrounded by <> brackets—in a message.
- 5 Click **Save and Close** in the policy window to save the changes.

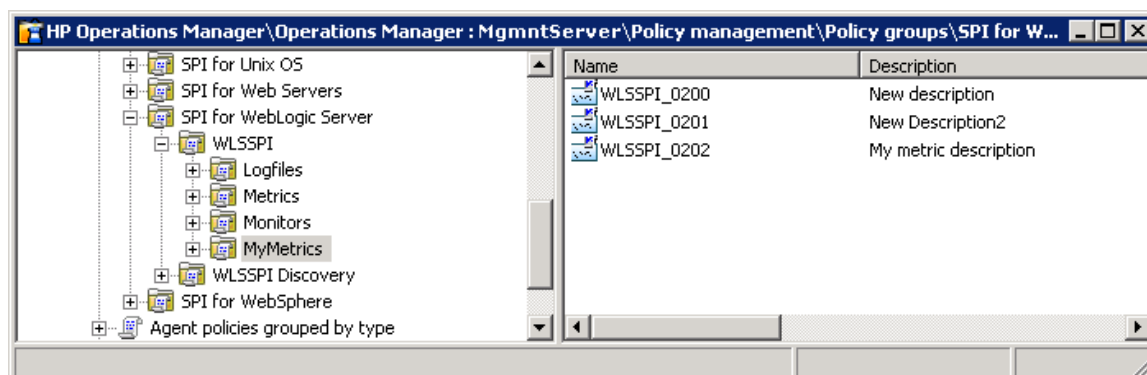
Advanced Policy Customizations

Advanced policy customizations include making copies of default policy groups to customize a few settings and deleting whole groups of metrics within a policy's command line.

Creating New Policy Group

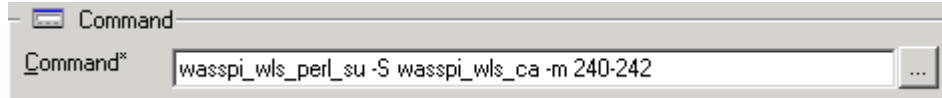
You can separate the custom policies that you create from the original default policies by creating new policy groups. Before you create a new policy group you must first determine the metrics and policies you want to modify. To create a new policy group, follow these steps:

- 1 Create a new policy group:
 - a In the HPOM console, select **Operations Manager** → **Policy Management** → **Policy Groups** → **SPI for WebLogic Server**.
 - b Right-click the policy group you want to copy and select **Copy**.
For example, right-click the **Metrics** policy group under WLSSPI and select **Copy**.
 - c Right-click the group under which this policy group is located and select **Paste**.
For example, right-click WLSSPI and select **Paste**.
 - d Right-click the new group and select **Rename**. Type in a new name.
For example, right-click Copy of Metrics and select **Rename**.
- 2 Rename the original policies within the new policy group:
 - a Double-click a the new policy group to get a list of the policies.
 - b Double-click a policy. The policy window opens.
 - c Click **File** → **Save As**. The Save As window opens.
 - d Enter a new policy name and click **OK**.
 - e Click **File** → **Exit** to close the policy window.
- 3 Delete all original policies within the new policy group. To do this, select the policies and press the **Delete** key. The Confirm Multiple Item Delete window opens.
Click **Yes** to confirm delete; otherwise click **No**.
- 4 Alter the renamed policies within the new group as necessary.



WebLogic SPI Collector/Analyzer Command with Parameters

The “wasspi_wls_perl -S wasspi_wls_ca” command is used in every collector policy. You can view the default command line parameters within each collector policy in the Command box in HPOM console. Double-click the policy to open the policy window. The Command box is within the policy window.



Basic WebLogic Server Command Parameters

The wasspi_wls_ca command is required to start the WebLogic SPI data collections. You can add other parameters to this command. The following table list the parameters used by the default collector policies.

Parameter	Description	Syntax with Example
-e	(exclude) Allows you to exclude specific servers; may not be used with -i option.	Syntax: -e <server_name> Example: -e server2,server4
-i	(include) Allows you to list specific servers to monitor. This option may not be used with -e option.	Syntax: -i <server_name> Example: -i server1,server3
-m	(metric) Specifies the metric numbers or number ranges on which to collect data.	Syntax: -m <metric_number,metric_number_range> Example: -m 1,3-5,9-11,15
-matchver	(match version) Specifies the exact WebLogic Server version to monitor. This option may not be used with the -minver nor -maxver options. If no matching versions are found, the command does not run.	Syntax: -matchver <version_number> Example: -matchver 6.1
-maxver	(maximum version) Specifies the highest WebLogic Server version to monitor. Use with -minver to specify a range of versions. If no versions are found, the command does not run.	Syntax: -maxver <version_number> Example: -matchver 7
-minver	(minimum version) Specifies the lowest WebLogic Server version to monitor. Use with -maxver to specify a range of versions. If no matching versions are found, the command does not run.	Syntax: -minver <version_number> Example: -matchver 6.1

Parameter	Description	Syntax with Example
-r	(report) Generate an ASCII report for the specified metrics.	Syntax: -r
-t	(tag) Allows you to create a new policy group by adding a prefix to an existing collector policy along with the metric numbers.	Syntax: wasspi_wls_ca -m <metric_number> -t <prefix>- Example: wasspi_wls_ca -m 220-223 -t DEV-
-x	Allows you to specify a property and value. Syntax: -x <property>=<property_value>\nProperty can be one of the following: <ul style="list-style-type: none"> alarm: When off, overrides any alarming condition as set up in the metric policy. Example: -x alarm=off prefix: Default: JMXUDM_. Specify the prefix of the metric ID. Example: -x prefix=SALES_ print: When on, prints the metric name, instance name, and metric value to STDOUT in addition to any configured alarming or logging. Example: -x print=on graph: When off, prevents graphing function. Example: -x graph=off report: When off, prevents reporting function. Example: -x report=off 	

Examples

- To collect specific data on all configured servers:
wasspi_wls_ca -m 10-14,25,26
- To collect data from specific servers only:
wasspi_wls_ca -m 245,246,260 -i server1,server3
- To not collect data from specific servers:
wasspi_wls_ca -m 220-225 -e server1,server2



Do not create separate collector/analyzer policies for the metrics that are separated by hyphen (-). For example, 71 - 76.

Using JMX Actions Command Parameters

This section describes the command parameters you can use to run JMX actions. JMX actions are one or more JMX calls (invoke, get, set) performed on an MBean instance or type. A single JMX call can be performed from the command line. Multiple JMX calls can be specified in an XML file or as a Metric sub-element in a UDM file.

Parameter	Description	Syntax with Example
-a Required	(action) Indicates a JMX action is performed.	Syntax: -a
-i	(include) Allows you to list specifies servers on which to perform the JMX actions. If this parameter is not specified, the JMX actions are performed on all configured servers.	Syntax: -i <server_name> Example: -i server1,server3
-m	(metric) Specifies the metric ID containing the action to perform. This metric ID must be defined in a UDM file. This option may not be used with the -mbean nor -xml options.	Syntax: -m <metric_id> Example: -m TestUDM_1000

Parameter	Description	Syntax with Example
-mbean	<p>Performs a JMX call on the specified MBeans. This option may not be used with the -m or -xml options.</p> <p>Syntax: -mbean <objectname> <action></p> <p>Example: -mbean *:* ,Type=JMSServerConfig -get MessagesMaximum where <action> (a JMX call) is one of the following:</p> <ul style="list-style-type: none"> -get: Returns the value of the specified attribute. <p>Syntax: -mbean <objectname> -get <attribute></p> <p>Example: -get MessagesMaximum</p> -invoke [-type]: Executes an MBean operation with the specified parameters. A type parameter must be specified for operations which accept parameters. -type supports operation overloading. If an operation does not require parameters, -type can be ignored. <p>Syntax: -mbean <objectname> -invoke <operation> [-type <parameter_type> <parameter_value>]...</p> <p>where <parameter_type> is one of the following: short, int, long, double, float, boolean, java.lang.Short, java.lang.Integer, java.lang.Long, java.lang.Double, java.lang.Float, java.lang.Boolean, or java.lang.String.</p> <p>Example: -invoke stagingEnabled -type java.lang.String examplesServer</p> -set: Assigns the specified value to the specified attribute. <p>Syntax: -mbean <objectname> -set <attribute> <value></p> <p>Example: -set MessagesMaximum 250000</p> 	
-o	(object) Specifies an MBean instance.	<p>Syntax: -o <mbean_instance></p> <p>Example: -o examplesJMSServer</p>
-xml	Specifies the XML file that contains one or more JMX actions to perform. This option may not be used with the -m nor -mbean options.	<p>Syntax: -xml <filename></p> <p>Example: -xml myJMXActions.xml</p>

Examples

- Set the maximum threads for an alarming WebLogic execute queue to 50 (<\$OPTION(instance name)> specifies an alarming instance):

```
wasspi_wls_perl -S wasspi_wls_ca -a
-mbean "PetStore:* ,Type=ExecuteQueueConfig"
-set ThreadsMaximum 50 -o <$OPTION(instance name)>
```
- Set the MessagesMaximum attribute to 25000 on multiple MBean instances:

```
wasspi_wls_perl -S wasspi_wls_ca -a
-mbean *:* ,Type=JMSServerConfig -set MessagesMaximum 250000 -i
examplesServer
```

- Set the MessagesMaximum attribute to 25000 on a specific MBean instance:

```
wasspi_wls_perl -S wasspi_wls_ca -a
-mbean *:*,Type=JMSServerConfig -set MessagesMaximum 250000 -i
examplesServer -o examplesJMSServer
```

- Invoke an operation on multiple MBean instances:

```
wasspi_wls_perl -S wasspi_wls_ca -a
-mbean *:*,Type=ApplicationConfig -invoke staged
-i examplesServer
```

- Get the MessagesMaximum attribute (after a set command, used to verify that the attribute was set):

```
wasspi_wls_perl -S wasspi_wls_ca -a
-mbean *:*,Type=JMSServerConfig -get MessagesMaximum
-i examplesServer
```

- Use the sample UDM TestUDM_1000 in the wls_UDMMetrics-sample.xml file:

```
wasspi_wls_perl -S wasspi_wls-ca -a -m TestUDM_1000
-i examplesServer
```

- Use the sample actions xml file:

```
wasspi_wls_perl -S wasspi_wls-ca -a
-xml /<wasspi_wls_conf_dir>/JMXActions-sample.xml
-i examplesServer
```

Where, <wasspi_wls_conf_dir> is var/opt/OV/wasspi/wls/conf for DCE agent and /var/opt/OV/conf/wlsspi for HTTPS agent.

Changing the Collection Interval for Scheduled Metrics

You can change the collection interval for all scheduled metrics by changing the Polling Interval in the respective collector policy. For example, to change the collection interval of default metrics from 5 minutes to 10 minutes for the WLSSPI-05min collector policy, follow these steps:

- 1 From the HPOM console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WLSSPI** → **Monitor**.
- 2 Double-click the collector policy **WLSSPI-05min**. The Measurement window opens.
- 3 Click **File** → **Save As**. The Save As window opens.
- 4 Change the existing name in the Name box to **WLSSPI-10min**.
- 5 Set the new interval.
 - a Click the **Schedule** tab.
 - b From the Schedule Task drop-down list select “Once per interval”.
 - c Set the interval to 10 minutes.
- 6 Deploy the new policy.
 - a Right-click **WLSSPI-10min** and select **All Tasks** → **Deploy on....**
 - b Select the nodes on which to deploy the policy.
 - c Click **OK**.

Changing the Collection Interval for Selected Metrics

You can change the collection interval of selected metrics, according to the requirements of your environment. For example, you can change the collection interval from 5 minutes to 10 minutes for metrics B070 to B081 of collector policy WLSSPI-05min.

Follow these steps:

- 1 Rename the selected metrics to reflect the new interval.
 - a From the HPOM console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WLSSPI** → **Monitor**.
 - b Double-click the collector policy **WLSSPI-05min**. The Measurement Threshold window opens.
 - c Click **File** → **Save As**. The Save As window opens.
 - d In the Name box change the existing name to **WLSSPI-10min**. Click **OK** to save or click **Cancel** to discard changes.
- 2 In the Command box, delete all metrics after the -m except 70-81.
- 3 Set the new interval.
 - a Click the **Schedule** tab.
 - b From the Schedule task drop-down list, select “Once per interval” and set the interval to 10 minutes.
 - c Click **Save and Close** to confirm the changes and close the policy window.
- 4 Edit the original policy to remove the modified metrics.
 - a Right-click the collector policy **WLSSPI-05min** and select **All Tasks** → **Edit**. The policy window opens.
 - b In the Command box, delete metrics 70-81 after -m.
 - c Select **Save and Close** to save the changes.
- 5 Deploy the modified policies.
 - a Right-click **WLSSPI-10min** and select **All Tasks** → **Deploy on....**
 - b Select the nodes on which you want to deploy the policy.
 - c Click **OK**.
 - d Right-click **WLSSPI-05min** and repeat steps b-d.

Customize Threshold Values for Different Applications/EJB/Servlet/JDBC

In your environment some applications may be more critical than others, also, within an application some of the EJBs/Servlets/JDBC datasource may be critical, and others may not. You can set threshold values per application or per EJB/Servlet/JDBC datasource depending on their criticality.

To do so, you must copy the existing condition and modify it. Follow these steps:

- 1 Copy the existing condition.
 - a From the HPOM console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WLSSPI** → **Metrics**.

- b Double-click the metric. The policy window opens. For example, double-click **WLSSPI_0253**.
 - c Click the **Threshold Levels** tab.
 - d Select the existing rule and click **Copy**.
- 2 Select the copy of the rule and click **Specify Instance Filters**. The New Rule window opens.
 - 3 Click the **Condition** tab. Type a rule description in the Rule description box.
 - 4 In the Object Name Matches field enter the following details (enter only the necessary fields, see [Examples](#) on page 44):


```
<ServerName.var1>:<ServerPort.var2>:<NodeName.var3>:<ApplicationName.var4>:
<EJBName/ServletName/JDBC DataSource.var5>:<Instance Name.var6>
```

var1, var2, var3, var4, var5, and var6 are user defined variables. These variables must be different from HPOM policy variables.
 - 5 Click the **Actions** tab.
 - 6 Select the new rule and click **Modify**. The Threshold Level window opens.
 - 7 Change the Threshold Level Description and in the Threshold limit box modify the threshold limit. Click **OK**.
 - 8 Click **OK** in the New Rule window.
 - 9 Click **Save and Close** in the Measurement Threshold window.
 - 10 Deploy the policy on the desired nodes.
 - a Right-click the policy and select Deploy On.
 - b Select the nodes on which you want to deploy the policy.
 - c Click **OK**.

Before customizing the threshold value, you may want to see the list of applications/EJBs/Servlets/JDBC datasource running on a server. For this you can use the following WebLogic SPI tools:

- View WebLogic Servers: This gives you details of all running application servers and the corresponding ports.
- View Deployed Apps: Gives a list of all applications deployed on a particular server.

For more information about these tools, see the WebLogic SPI online help.

Examples

Here are a few examples that will help you in entering details in the Object Pattern field:

- Example 1: If you want to set threshold for the application MedrecEAR, and if the application name is unique across all the nodes then, enter the following:

```
<*.var1>:<*.var2>:<*.var3>:MedrecEAR:<*.var5>:<*.var6>
```

- Example 2: If you want to set the threshold for the application MedrecEAR that is available on node 1 and node 2, then to set the threshold only on node 1, enter the following:

```
<*.var1>:<*.var2>:node1:MedrecEAR:<*.var5>:<*.var6>
```

- Example 3: If you want to set the threshold for the Servlet- FileServlet under the application MedrecEAR, and FileServlet is unique across all the nodes, enter the following:

```
<*.var1>:<*.var2>:<*.var3>:MedrecEAR:FileServlet:<*.var6>
```

Creating Custom Tagged Policies

You can customize a policy by using the tag option (`-t` on the command line) that allows the collector/analyzer to recognize customized policies that have a tag attached to the name. This option gives you the flexibility of using more than a single set of policies to define conditions related to specific installations of WebLogic Server. It also prevents policies from being overwritten when you upgrade the WebLogic SPI.

When multiple nodes are managed by a number of groups, you can use this option to create specially tagged policies that are separate from your original setup. In such a case, make copies of the policies, rename them with the tag, edit the collector policy to pick up the tagged names, and then assign the policies to various groups.

For example, you may create a group of policies and change each policy name to include CLIENT01 in it. You may name a metric policy as CLIENT01-WLSSPI_0012 (retaining the name of the metric used) and name the collector policy as FIRST_CLIENT-WLSSPI-05min. Similarly, you may set up another group for SECOND_CLIENT and modify the policy names to include SECOND_CLIENT.

To create a new tagged policy group, follow these steps:

- 1 Copy the original policy group.
 - a Right-click the policy group you want to copy and select **Copy**.
For example, right-click the **Metrics** policy group under WLSSPI and select **Copy**.
 - b Right-click the group under which this policy group is located and select **Paste**.
For example, right-click WLSSPI and select **Paste**.
 - c Right-click **Copy of Metrics** and select **Rename**. Rename the new group to identify the new metric policies.
For example, rename the group to CLIENT01Metrics.
- 2 Rename the original policies within the new policy group.

The names of the metric polices in the new group must contain the new name followed by the original metric number. For example, you can rename a copy of WLSSPI_0001 as CLIENT01-WLSSPI_0001.

The name you give to the new collector policy must also contain the identifying name. You must also modify the scheduled collection to include the new group by inserting the `-t` property in the Command box. The Command box is in the policy window that appears when you double-click the collector policy.

For example: `wasspi_wls_ca -m 1,12,16 -t CLIENT01-`

- a Right-click the policy and select **All Tasks** → **Edit**. The policy window opens.
 - b Click **File** → **Save As**. The Save As window opens.
 - c Type a new policy name and click **OK**.
- 3 Select the original policies within the new policy group and press the **Delete** key to delete all the original policies. The Confirm Multiple Item Delete window opens.
 - 4 Click **Yes** to confirm delete.

Restoring Default WebLogic SPI Policies

To restore the default WebLogic SPI policy groups on your management server, you must remove and then reinstall the WebLogic SPI. For more information, see [Removing the WebLogic SPI](#) on page 14 and [Installing the WebLogic SPI](#) on page 11.

Viewing Text-Based Reports

Some policies have actions defined with threshold violations or error conditions. These actions automatically generate reports. The reports are snapshots of data values collected from the server around the time that the alarm occurred.



The reports discussed in this section are different from HP Reporter reports that show consolidated data generated as web pages in a management-ready presentation format. See [Integrating the WebLogic SPI with HP Reporter](#) on page 58.

Automatic Command Reports

Many metrics generate Automatic Command Reports. These reports are generated as soon as an alarm is triggered in HPOM. Automatic Command reports are generated for a single WebLogic Server instance with the exceeded threshold.

When an Automatic Command report is executed from HPOM, the server is queried for additional data. If you set the HPOM console message browser to display the SUIAON column, you can see an “S” under the “A” column (see the following figure), which indicates that a generated report is available in the Annotations area of the Message Properties.

Severity	S	U	I	A	O	N	Received	Group
Warning	-	-	X	-	-	-	1/14/2003 10:06:34 AM	OpC
Warning	-	-	-	-	-	-	1/16/2003 12:57:15 PM	WINOSSPI-NO...
Normal	-	-	-	-	-	-	1/17/2003 9:34:23 AM	VP_SM
Critical	0	-	X	S	S	X	1/17/2003 9:34:27 AM	WebLogic
Warning	-	-	X	-	-	-	1/17/2003 9:39:24 AM	OpC
Critical	-	-	-	-	-	-	1/23/2003 9:05:22 AM	VP_SM

Summary: 812 Critical, 0 Warning, 0 Information, 7 Normal, 5 Error, 0 Audit, 0 Debug

To view Automatic Command reports, do one of the following:

- Double-click a message in the HPOM message browser. The Message Properties window opens. Select the Annotations tab.
- Right-click a message and select **Annotations**. The Message Properties window opens.

The reports are available in the Message Properties window. These reports show data values of a single server. Column descriptions in the window provide further information.

Manually Generated Reports

Reports are generated for all WebLogic Server instances configured on the managed node. In contrast to Automatic Command reports that are generated for a single WebLogic server instance, manually generated reports reflect the current state of all WebLogic server instances on the managed node.

To manually generate a report, follow these steps:

- 1 From the HPOM console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **Metric Reports**.
- 2 Double-click the report you want to see. The Select Where to Launch This Tool window opens.
- 3 Select the managed node for which you want to see reports and click **Launch**. The Tool Status window opens.
- 4 View the report in the tool output field.
- 5 Click **Close** to close the window.

Sample Report

```
Report for Application Server_01
Oct 16, 2001 3:22:20 PM
Metric B011_ExQThrdUtilPct

Execute Queues                Idle Threads    Waiting Requests
-----
_weblogic_admin_html_queue    2               0
_default                      11              0
_weblogic_admin_rmi_queue     10              0

Execute Queues                Longest Waiting Request
-----
_weblogic_admin_html_queue    Oct 16, 2001 3:22:20 PM
_default                      Oct 16, 2001 3:22:20 PM
_weblogic_admin_rmi_queue     Oct 16, 2001 3:22:20 PM

Execute Queues | Threads      Current Request
-----
_weblogic_admin_html_queue | ExecuteThread[1]    null
_weblogic_admin_html_queue | ExecuteThread[2]    null

Execute Queues | Threads      Current Request
-----
default | ExecuteThread[1]      null
default | ExecuteThread[2]      null
default | ExecuteThread[3]      null
default | ExecuteThread[4]      null
default | ExecuteThread[5]      null
default | ExecuteThread[6]      null
default | ExecuteThread[7]      null
default | ExecuteThread[8]      null
default | ExecuteThread[9]      null
default | ExecuteThread[10]     null
default | ExecuteThread[11]     weblogic.rmi.internal.BasicExecuteRequest@f0c95
default | ExecuteThread[12]     Socket Reader Request
default | ExecuteThread[13]     Socket Reader Request
default | ExecuteThread[14]     Read Multicast Msg Fragment
```

WebLogic SPI Graphs

Some policies have operator actions associated with them that allow you to generate a graph. To view these graphs, follow these steps:

- 1 Double-click a message in the HPOM message browser. The Message Properties window opens.
- 2 Click the **Commands** tab. You can generate a graph if an operator-initiated command is configured and data is collected.
- 3 Click **Start** to generate the graph.

Monitoring WebLogic Server on Unsupported Platforms

The WebLogic SPI supports monitoring WebLogic Server-installed systems running on HP-UX, Solaris, Linux, Windows 2000, and AIX. It is also possible to configure the WebLogic SPI to monitor a WebLogic Server installed on systems running on unsupported platforms—systems we refer to as “remote systems.”

This section explains how to determine if your environment is favorable to setting up remote monitoring. If you determine that your environment meets the criteria described below, and you have some expertise in using the WebLogic SPI, this section offers an example to get you started.

Monitoring Remote Nodes (Running on Platforms Not Supported by WebLogic SPI)

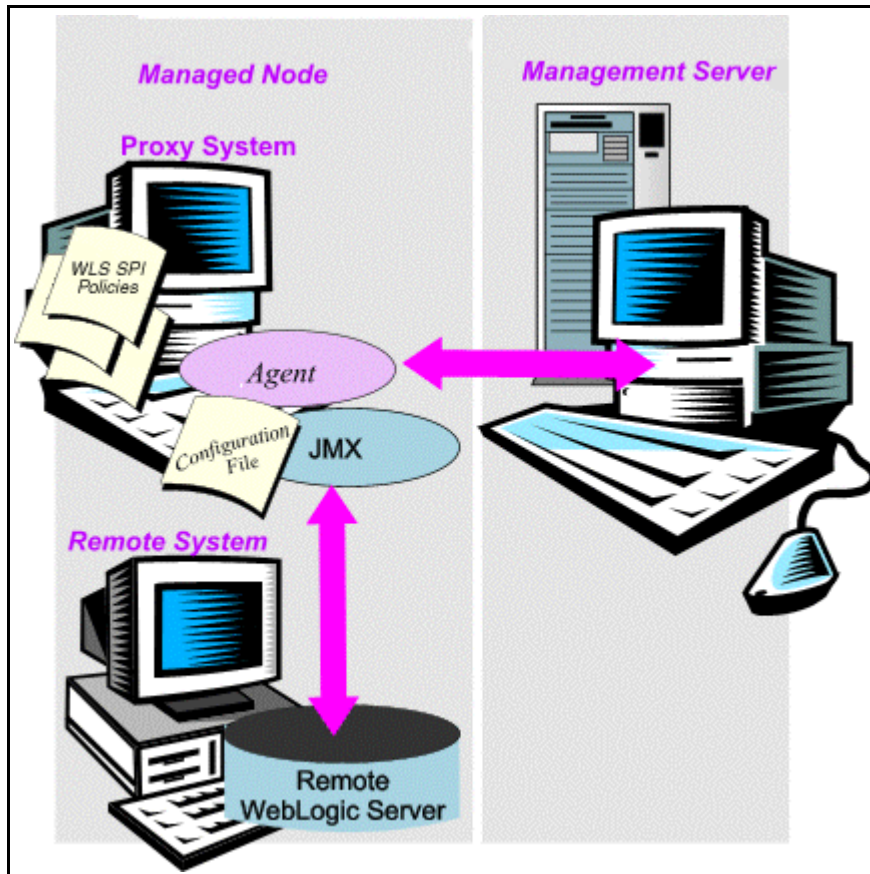
For a WebLogic Server installation on a system running on a platform other than HP-UX, Solaris, Linux, Windows 2000, or AIX, you can use the WebLogic SPI to monitor that remote system if the following conditions apply. The last condition is optional.

- The remote system is covered by a purchased license (using Tier 1 pricing).
- The WebLogic SPI runs on at least one managed node on a supported platform: HP-UX, Solaris, Linux, Windows 2000, or AIX.
- (Optional, for logfile monitoring) The remote system runs on a platform supported by the HP Operations Agent software.

Implementing Remote Monitoring

In a standard configuration, WebLogic SPI programs or policies are deployed on the local, managed node. In a non-standard configuration, the local system is used as a proxy through which remote metric information becomes accessible.

Remote system data collection and interpretation relies on the local, managed node to act as the proxy on which data collection is configured.



Configuration entries requirement:

Within the configuration, entries for both local and remote systems are included. You can include multiple remote system entries in a local system’s section. [Example Configuration](#) on page 51, shows how the remote entry appears with system IP address.

Policy deployment requirement:

Policies must be deployed on the local node.

HP Operations Agent deployment requirement (optional logfile monitoring):

To access remote WebLogic logfiles, the HP Operations Agent software must be installed on the remote system. Using standard HPOM processes, you can modify the standard logfile policies included with the WebLogic SPI to specify the correct logfile names and then deploy them on the remote system.

- ▶ Monitoring remote systems using logfile versioning is not supported.

Configuring Remote System Monitoring

You can monitor WebLogic servers on remote systems (running on operating systems other than HP-UX, Solaris, Linux, Windows 2000, or AIX) by completing the following tasks.

Task 1: Configure the Remote WebLogic Server

Using the Configure WLSSPI tool of the SPI Admin tools group, configure each local managed node that communicates with a remote WebLogic server. In the configuration, add entries for remote WebLogic servers.

- 1 Launch the Configure WLSSPI tool. For details, see the Tools section in the WebLogic SPI online help.
- 2 Select a WebLogic managed node from which to monitor the remote WebLogic Server.
- 3 In the configuration, include an entry for each remote WebLogic server at the server specific level:

ADDRESS=<DNS server name or IP address>.

The example configuration below shows how local and remote WebLogic servers are configured in the same file. For the remote servers the ADDRESS=<IP_address> line is added:

```
ADDRESS=15.75.27.109 or
ADDRESS=harley.hp.com
```

Example Configuration

```
#
#####
HOME=/opt/bea/wlsserver6.0spi
JAVA_HOME=/opt/bea/jdk130

SERVER1_NAME=classact
SERVER1_PORT=7001
SERVER1_LOGIN=server1_admin
SERVER1_PASSWORD=server1_password

SERVER2_NAME=harley
SERVER2_PORT=7002
SERVER2_LOGIN=server2_admin
SERVER2_PASSWORD=server2_password
SERVER2_ADDRESS=harley.hp.com
```

In this example, SERVER1 is the local server, running on a HP-UX managed node. SERVER2 is running on an HP Operations Agent-managed node, that is a system on a platform unsupported by WebLogic SPI. The remote system is configured similar to that of the local system but contains the new line SERVER2_ADDRESS=harley.hp.com.

Task 2: (Optional) Integrate HP Performance Agent

The HP Performance Agent collection occurs on the managed node, not the remote system. Therefore, if you use HP Performance Manager and want to graph the remote system data, ensure that HP Performance Agent integration is enabled on the local managed node.

Task 3: Assign Local Node to WebLogic SPI node group

Assign the local managed node to the SPI for WebLogic Server node group.

Configuring Remote Monitoring for Logfiles (Optional)

Monitoring remote system logfiles is supported if the following are true:

- 1 The HP Operations Agent is running on the remote system.
- 2 The system does not re-version logfiles when they roll.

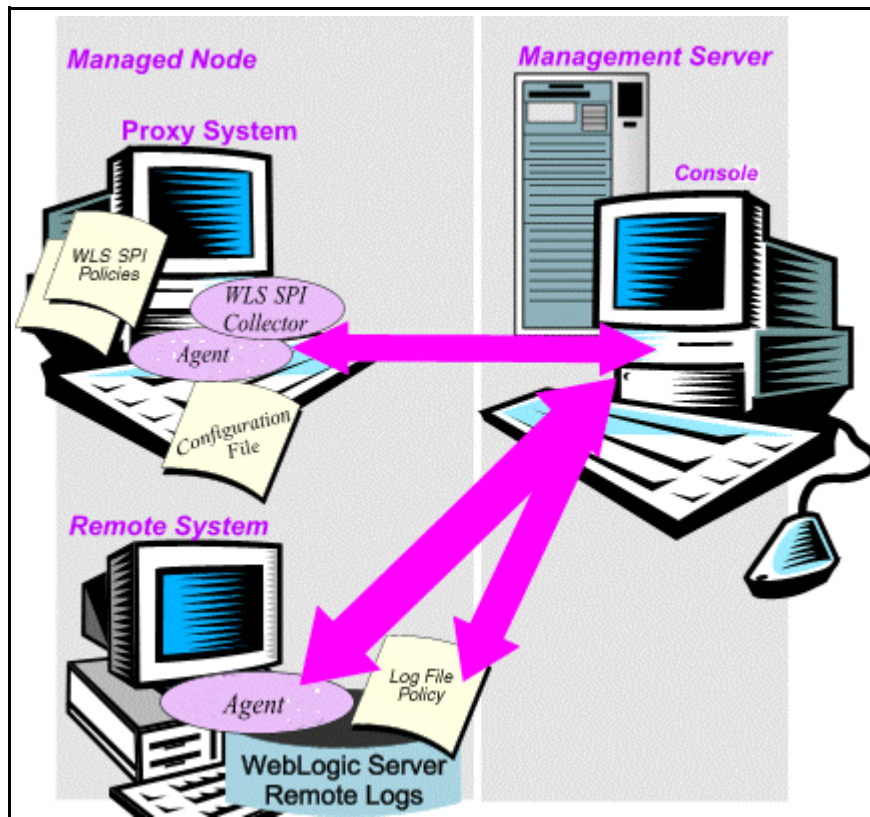
To set up logfile monitoring, in the HPOM console, copy the WebLogic SPI logfile policy and then configure, assign, and deploy the copied logfile policy to the remote system.

Configuring the Logfile Policy for Remote Logfiles

To configure the logfile policy for remote logfiles, follow these steps:

- 1 From the HP Operations Agent console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WLSSPI**.
- 2 Select **Logfiles** and double-click the log policy.
- 3 In the Logfile pathname box, type the location of the logfile on the remote system: `/<path>/<filename>`.
- 4 Assign and deploy the logfile policy to the remote HP Operations Agent-managed node.

WebLogic server logfile monitoring is possible because of the Logfile policy and the HP Operations Agent present on the remote system.



Limitations in Remote Monitoring

- The WebLogic SPI and the HP Operations Agent do not support access to logfiles that are re-versioned each time the logs are rolled.
- WebLogic logfiles on the remote system cannot be monitored if an HP Operations Agent is not present on the remote system.
- You cannot run WebLogic SPI tools on remote systems.
- WebLogic SPI does not support application servers with the same name.

5 Integrating HPOM Reporting and Graphing Features with the WebLogic SPI

The WebLogic SPI can be integrated with the following HP Software products. These products must be purchased separately.

- **HP Reporter**

Reporter produces management-ready web page reports that show historical and trends related information.

After you integrate HP Reporter with WebLogic SPI, Reporter generates a variety of reports, every night, that show consolidated information about the performance and availability of WebLogic Servers on configured managed nodes. See [Integrating the WebLogic SPI with HP Reporter](#) on page 58.

- **HP OpenView Performance Insight**

HP OpenView Performance Insight is a network management system that collects, processes, and reports data. This data is used to generate reports. For more information, see the *HP Open View Performance Insight Administration Guide*.

For information on WebLogic SPI reports and integrating WebLogic SPI with HP OpenView Performance Insight, see the *Application Server Report Pack User Guide*.

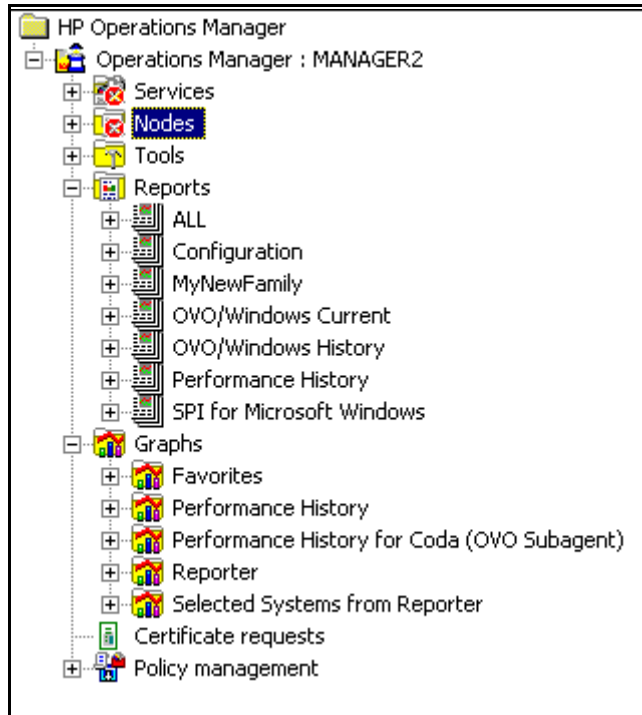
- **HP Performance Manager**

HP Performance Manager provides graphing capability.

After you integrate HP Performance Manager with the WebLogic SPI, you can view the graphs the following day. However, the graphs are available only if performance data is logged in the default performance subagent CODA or HP Performance Agent. CODA is automatically deployed on all HPOM managed nodes.

See [Integrating the WebLogic SPI with HP Performance Manager](#) on page 65.

Figure 2 The Management Server Console Tree



Integrating the WebLogic SPI with HP Performance Agent

If your IT environment requires you to generate graphs and reports from historical data or to store large volumes of performance data, you may want to use the HP Performance Agent to collect and store performance data. HP Performance Agent must be purchased separately.

The data collected by HP Performance Agent is used by HP Reporter, HP Performance Insight and HP Performance Manager.



If you are running HP Performance Agent 4.x for Linux, you are not required to configure the WebLogic SPI data collector to use HP Performance Agent. By default, the WebLogic SPI detects and uses this version of HP Performance Agent to collect and store performance data.

To configure the WebLogic SPI data collector to use HP Performance Agent, follow these steps:

- 1 Create a `nocoda.opt` file on the managed node, in the following directory:

Operating System	File Location
HP-UX, Linux, Solaris	<code>/var/opt/OV/conf/dsi2ddf/</code>
AIX	<code>/var/lpp/OV/conf/dsi2ddf/</code>
Windows	<code>C:\Program Files\HP Openview\data\conf\dsi2ddf\</code>

If the directory `dsi2ddf` does not exist, create it.

- 2 Edit the `nocoda.opt` file to contain a single line:

ALL

- 3 Save the file.

Integrating the WebLogic SPI with HP Reporter

You must install HP Reporter in your environment to access WebLogic SPI reports from the HPOM console.

Before integrating the WebLogic SPI with HP Reporter, you must configure the WebLogic SPI by deploying the software, configuring server connection, and assigning/deploying policies on target managed nodes.

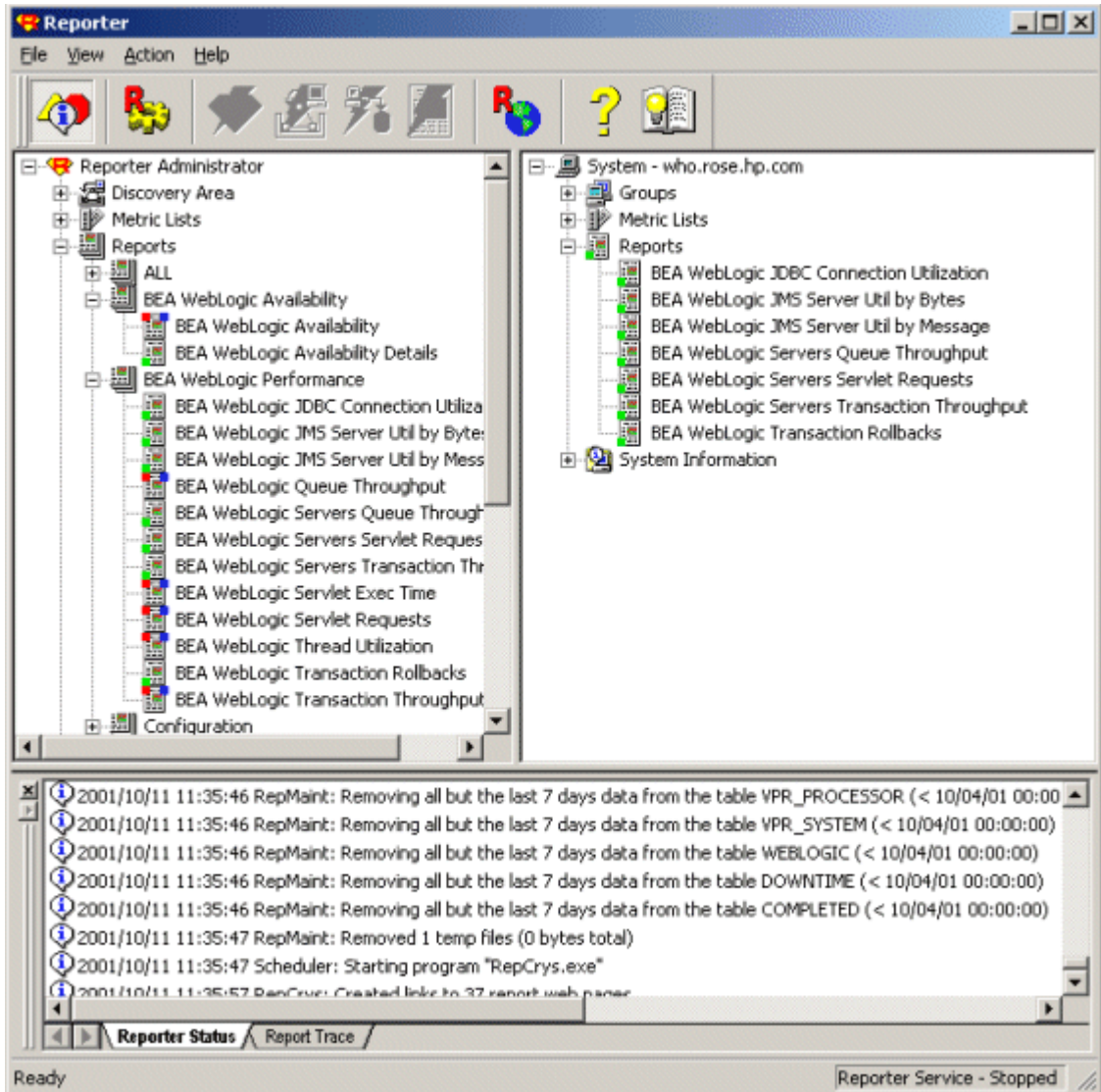
To integrate the WebLogic SPI with HP Reporter, follow these steps:

- 1 Install the WebLogic SPI report package on the Windows system running Reporter.
 - a Insert the HP Operations Smart Plug-ins DVD into the DVD drive of the Windows system running Reporter. The HP Operations Manager InstallShield Wizard opens.
 - b Click **Next**. The Program Maintenance window opens. Click **Install Products**. The Product Selection window opens.
 - c From the options listed, select the **Reporter** option of BEA WebLogic and click **Next**.
 - d Complete the installation by following the instructions that appear as you proceed.
- 2 To see the Reporter window, click **Start** → **All Programs** → **HP OpenView** → **Reporter** → **Reporter**.
- 3 Check the Reporter window (see the illustration that follows) to note changes in Reporter's configuration.



On Windows 2000 managed nodes, when installing the WebLogic SPI report package, you may get an error message indicating that the installer has detected an older version of the installer on your system. You can safely ignore the message and continue.

In the Reporter status pane (at the bottom of the Reporter window), you can view information on programs that are running and any errors occurring on the managed nodes. You can check the status pane to see whether Reporter is updated with the WebLogic SPI reports.



In the Reporter Help, you can find instructions for assigning WebLogic SPI reports to the target nodes. To access Help, follow these steps:

- a Right-click **Reports** or **Discovered Systems** in the left panel of the Reporter main window.
 - b Select Report Help or Discovered Systems Help.
 - c Read the topic - To assign a report definition to a Discovered Systems Group.
- 4 Add group and single system reports by assigning reports as desired. For complete information, see the Reporter Help and the online *Concepts Guide*.

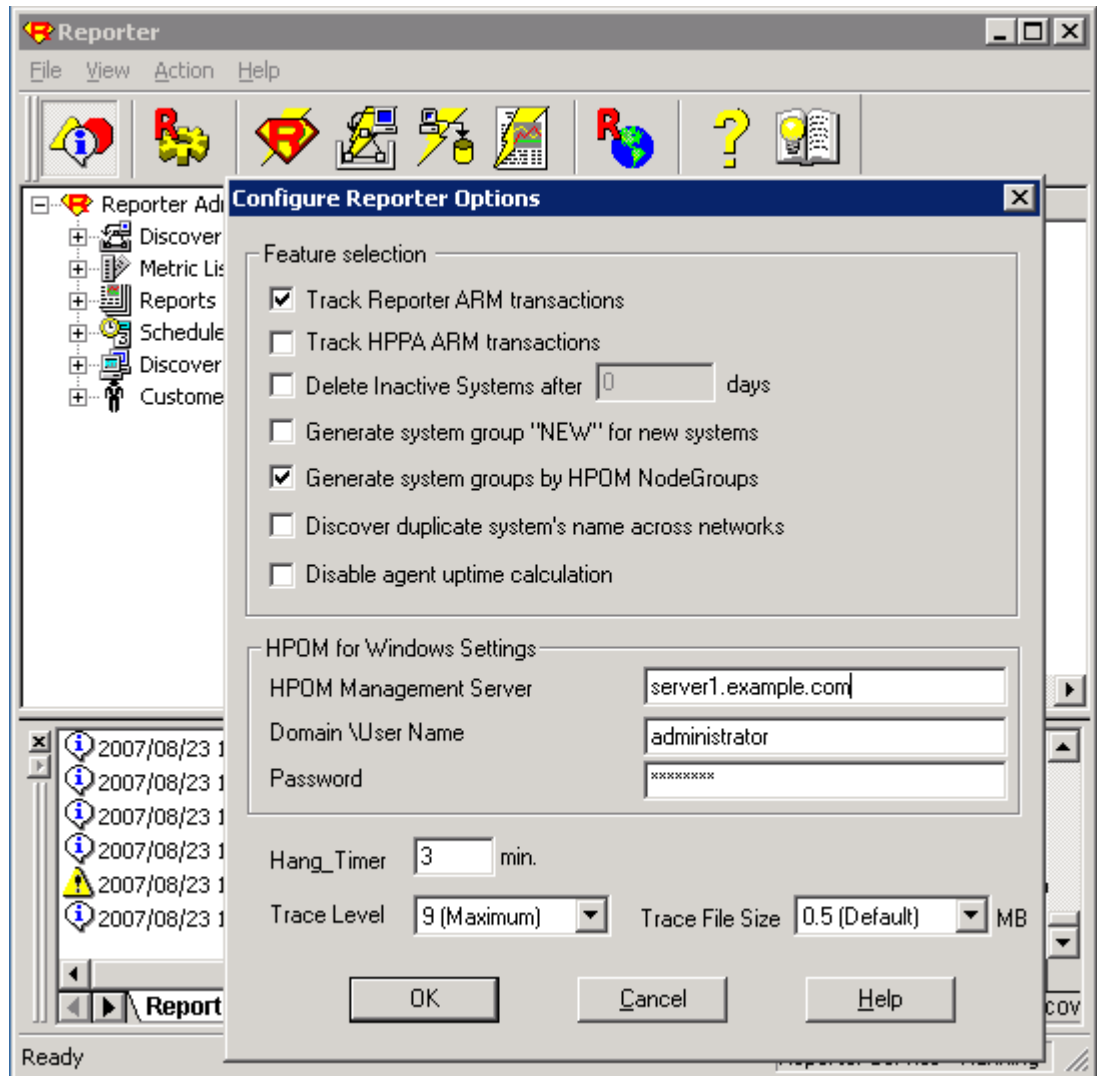


For group and single system WebLogic SPI reports you are required to identify systems by their full name. For example, **abc.xyz.com** is acceptable while **abc** is not.

Viewing Reports from the HPOM Management Console

To view WebLogic **Reports** from the HPOM Console, follow these steps:

- 1 Close the HPOM for Windows console (if it is open).
- 2 Open the HP Reporter window and from the menu bar click **File** → **Configure** → **Options**. The Configure Reporter Options window opens.



- 3 In the HPOM for Windows Settings section, specify the name of the management server and user details. The user must be an HPOM administrator (a member of the HP-OVE-Admins group). Click **OK**.
- 4 From the menu bar again, click **Action** → **Run** → **Run All**.
This will discover the node data from HPOM and generate the reports. This may take some time.
- 5 After the HP Reporter tasks complete, open the HPOM console. **Reports** will be visible in the console tree.

Reports Generated by Reporter

The reports available through the integration of Reporter and the WebLogic SPI show consolidated data on server performance and availability on all WebLogic Server systems. In addition, other reports show data for single systems. These reports are available one day after you install the WebLogic SPI report package on the Reporter Windows system. See [Integrating the WebLogic SPI with HP Reporter](#) on page 58 if you have not yet completed the report package installation.

The following tables show pre-defined reports.

Table 1 Reports for All Systems - WebLogic Performance

Report Title	Description	WebLogic Version
EJB Free Pool Wait Rate - Top 20	Shows the number of requests for an EJB per minute that had to wait for an instance of the EJB to become available from this EJB's free pool. The top 20 servers are selected based on the highest average number of requests per minute over the reporting period.	7.0, 8.1, 9.x
EJB Timeout Rate - Top 20	Shows the number of requests for an EJB that timed out per minute while waiting for an instance of the EJB to become available from this EJB's free pool. The top 20 servers are selected based on the highest average number of timeouts per minute over the reporting period.	7.0, 8.1, 9.x
EJB Transaction Throughput - Top 20	Shows the average number of transactions processed per second by EJBs. The top 20 servers are selected based on the highest average number of transactions processed per second over the reporting period.	6.1
Server Queue Throughput - Top 20	Shows the average number of requests processed by a server's execute queue per second. Note that a server may have more than one execute queue. The top 20 server queues are selected based on the highest average number of requests processed per second over the reporting period.	6.1, 7.0, 8.1, 9.x
Server Queue Utilization - Top 20	Shows the utilization of the server's execute queue thread pool as a percent of the number of threads configured for the pool. The top 20 server queues are selected based on the queue utilization mean value.	6.1, 7.0, 8.1, 9.x

Table 1 Reports for All Systems - WebLogic Performance

Report Title	Description	WebLogic Version
Servlet Average Response Time - Top 20	Shows the average response time for the top 20 servlets. The top 20 servlets are selected based on the highest average number of requests per second for the servlet over the reporting period.	6.1, 7.0, 8.1, 9.x
Servlet Request Rates - Top 20	Shows the number of servlet requests per second by a server. The top 20 servers are selected based on the highest average number of servlet requests per second for the server over the reporting period. Along with the servlet name, the associated application name is also displayed.	6.1, 7.0, 8.1, 9.x
Transaction Throughput - Top 20	Shows the average number of transactions processed per second for each server. The top 20 servers are selected based on the highest average number of transactions processed per second over the reporting period.	6.1, 7.0, 8.1, 9.x

Table 2 WebLogic Availability

Report Title	Description	WebLogic Version
Server Availability	Contains a daily histogram showing the percentage of uptime. In addition, a trend line provides the number of measurements performed, indicating how much data was available to determine availability. Uptime and downtime are measured by the WebLogic SPI. A lower than expected trend line may indicate systems were unavailable or the data collection was not running.	6.1, 7.0, 8.1, 9.x

Table 3 Reports for Single System

Report Title	Description	WebLogic Version
DB Connection Pools Throughput vs. Connection Utilization - Top 20	<p>Compares the throughput vs. the utilization of the DB connection pools on the server.</p> <p>Throughput is the number of connections allocated by a DB connection pool per second. The utilization of a connection pool is the number of connections being used as a percent of the maximum capacity configured for the pool.</p> <p>The top 20 Servers are selected based on the highest average of throughput over the reporting period.</p>	6.1, 7.0, 8.1, 9.x
Stateful and Entity EJB Cache Hits - Top 20	<p>Shows the percent of time a request to access a bean from an EJB's cache succeeded for the server.</p> <p>The top 20 EJBs are selected based on the highest average cache hit percent over the reporting period. Stateful and entity EJBs are included in this data.</p>	7.0, 8.1, 9.x
Throughput vs. Utilization Of JMS Server by Message Size - Top 20	<p>Compares the throughput vs. utilization of the JMS Servers on the server based on the size of JMS messages.</p> <p>The throughput is the number of JMS messages processed by a JMS server per second. The utilization of the message queue is the total size of the messages being processed as a percent of the maximum size configured for the queue.</p> <p>The top 20 Servers are selected based on the highest average of throughput over the reporting period.</p>	6.1, 7.0, 8.1, 9.x
Throughput vs. Utilization Of JMS Server by Message Count - Top 20	<p>Compares the throughput vs. utilization of the JMS Servers on the server based on the number of JMS messages.</p> <p>The throughput is the number of JMS messages processed by a JMS server per second. The utilization of the message queue is the number of messages being processed as a percent of the maximum number of messages configured for the queue.</p> <p>The top 20 Servers are selected based on the highest average of throughput over the reporting period.</p>	6.1, 7.0, 8.1, 9.x

Table 3 Reports for Single System

Report Title	Description	WebLogic Version
Server Availability Details	<p>Contains spectrum graphs showing minutes of uptime by day and hour for the system. Uptime and downtime are measured by the WebLogic SPI. “No Data” may include system downtime or data collection not running. Graphs are based on measured uptime and downtime only (i.e. standby = down).</p> <p>The spectrum graphs use color to indicate the uptime percentage during each hour of each day.</p>	6.1, 7.0, 8.1, 9.x
Server Queue Throughput vs. Utilization - Top 20	<p>Shows the throughput vs. the utilization of the server execute queues on the server. The throughput is the number of requests processed by a server queue per second.</p> <p>The utilization of the server queue is the number of busy threads in the server queue thread pool as a percent of the number of threads configured for the pool.</p> <p>The top 20 Servers are selected based on the highest average of throughput over the reporting period</p>	6.1, 7.0, 8.1, 9.x
Transaction Throughput - Top 20	<p>Shows the average number of transactions processed per second for each server on the system.</p> <p>The top 20 servers are selected based on the highest average number of transactions processed per second over the reporting period.</p>	6.1, 7.0, 8.1, 9.x
Servlet Requests - Top 20	<p>Shows the number of requests per second for servlets on the server.</p> <p>The top 20 servlets are selected based on the highest average number of requests per second for the servlet over the reporting period.</p>	6.1, 7.0, 8.1, 9.x
Transaction Rollbacks	<p>Shows the percent of processed transactions that had to be rolled back for the server.</p> <p>Each color in the chart bar represents the amount of the rollback percent that is attributable to each error category.</p>	6.1, 7.0, 8.1, 9.x

Removing the WebLogic SPI Reporter Package

To remove the WebLogic SPI Reporter Package, follow these step:

- 1 Insert the HP Operations Smart Plug-ins DVD into the DVD drive of the Windows system running Reporter. The HP Operations Manager InstallShield Wizard opens.
- 2 Click **Next**. The Program Maintenance window opens.
- 3 Click **Remove Products**. The Product Selection window opens.
- 4 From the options listed select the **Reports** option of BEA WebLogic and click **Next** till the Remove the Selected Products window opens.
- 5 Click **Remove**.

Integrating the WebLogic SPI with HP Performance Manager

You must install HP Performance Manager on the HPOM management server to access WebLogic SPI graphs from the HPOM console.

To integrate the WebLogic SPI with HP Performance Manager, follow these steps:

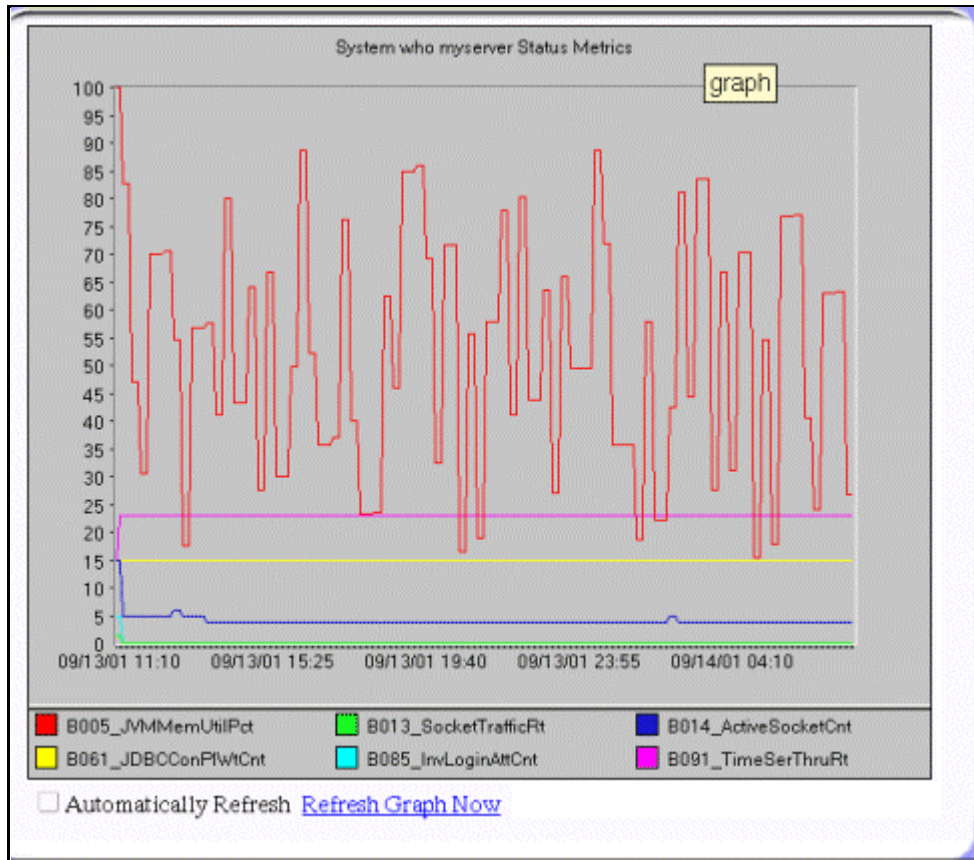
- 1 Install the WebLogic SPI graph package on the Windows system running HP Performance Manager:
 - a Insert the HP Operations Smart Plug-ins DVD into the DVD drive of the Windows system running Performance Manager. The HP Operations Manager InstallShield Wizard opens.
 - b Click **Next**. The Program Maintenance window opens. Click **Install Products**. The Product Selection window opens.
 - c From the options listed (there are three Product Selection windows) select the **Graph** option of BEA WebLogic and click **Next**.
 - d Complete the installation by following the instructions as you proceed.
- 2 To graph any WebLogic server metric use the data source name—`WLSspi_METRICS`

Viewing Graphs that Show Alarm Conditions

For graphing purposes, the WebLogic SPI organizes metrics into four groups, according to their type. When a message is generated for any metric (listed in the tables in the following section), you can view a graph of that metric along with other metric values.

To view a graph associated with an alarm condition (operator-initiated action must be defined with the WebLogic SPI policy), follow these steps:

- 1 In the HPOM Message Browser double-click the message for which you want to view the graph. The Message Properties window opens.
- 2 Click the **Commands** tab.
- 3 Click **Start** in the section Operator Initiated to start the operator-initiated command.
The operator action launches the web browser, where you can view the graph.



Viewing Graphs that Show Past or Current Conditions

To generate an available graph manually, follow these steps:

- 1 From the console, select **Operations Manager** → **Graphs** → **SPI for WebLogic Server**.
- 2 Double-click the graph you want to generate. A new window opens.
- 3 Select the nodes from which you want to retrieve data. Select the date range and the granularity for the graph.
- 4 Click **Finish**.




Graphs appears in the HPOM console tree only if you install HP Performance Manager on the same system as the HPOM management server.

Viewing Graphs from the HP Performance Manager Console

If you have not installed HP performance Manager on the same system as HPOM management server, you can view the WebLogic SPI Graphs from the HP Performance Manager console. Follow these steps:

- 1 Click **Start** → **All Programs** → **HP** → **HP BTO Software** → **Performance Manager** → **Performance Manager**. The Performance Manager console opens.

- 2 From the Select Nodes pane, select the node for which you want to see graph. If the node is not listed in the list, add the node:
 - a Click **Admin** in the menu bar. The Manage Nodes window opens.
 - b Click the **Add a Node**  icon. The Add a Node Window opens.
 - c Enter the node name and click **Add**.
 - d Click **Home** on the menu bar.
- 3 From the Select a Graph pane, select **SPI for WebLogic Server**.
- 4 Select the graph you want to see and click **Draw**.

▶ If you have installed HP Performance Agent to collect performance data, you must select the **SPI for WebLogic Server - OVPA <version>** from the list in the Select a Graph pane.

WebLogic SPI Metrics Available for Graphs

The following tables show the graphs available for mapping collected metric values. If you want to view the graph for any of the metrics included in the following tables, you can use the View Graphs tool. A graph of the metric as well as other related metrics appears in your web browser.

Table 4 Cluster: 80, 81

Graph Label	Metric Name	Metric Description
Cluster Outgoing Message Failure Rate	B080_ClsOutMesFailRt	Number of multicast messages per minute resent to cluster.
Cluster Incoming Message Failure Rate	B081_ClsInMesFailRt	Number of multicast messages per minute from cluster lost by the server.

Table 5 Enterprise Java Beans (EJB): 25, 26, 35, 36

Graph Label	Metric Name	Metric Description
Aggregate EJB Free Pool Wait Rate	B025_EJBFreePoolWtRt	Number of times per minute that no EJB beans were available from the free pool.
EJB Timeout Rate	B026_EJBTimeoutRt	Number of times per minute a client timed out waiting for an EJB.
EJB Transaction Throughput Rate	B035_EJBTranThruRt	Number of EJB transactions per second.
EJB Transaction Rollback Rate	B036_EJBTranRbRt	Number of EJB transactions rolled back per second.

Table 6 Server Status (Serverstat): 5, 13, 14, 61, 85, 91

Graph Label	Metric Name	Metric Description
JVM Memory Utilization Percent	B005_JVMMemUtilPct	Percentage of heap space used in the JVM.
Socket Traffic Rate	B013_SocketTrafficRt	Number of socket connections opened per second.
Active Socket Count	B014_ActiveSocketCnt	Number of socket connections opened.
JDBC Connect Pool Wait Count	B061_JDBCConPIWtCnt	Number of clients waiting for a connection from the connections pools.
Invalid Login Attempts Count	B085_InvLoginAttCnt	Number of invalid login attempts.
Timer Services Throughput Rate	B091_TimeSerThruRt	Number of triggers executed per second.

Table 7 Transaction: 70, 71, 72, 73, 74, 75, 76, 77

Graph Label	Metric Name	Metric Description
Transaction Average Time	B070_TranAveTime	Average commit time for transactions.
Transaction Rollback Percent	B071_TranRollbackPct	Percentage of transactions rolled back, based on the total.
Transaction Resource Error Rollback Percent	B072_TranResErrRbPct	Percentage of the transactions rolled back due to resource error.
Transaction Application Error Rollback Percent	B073_TranAppErrRbPct	Percentage of transactions rolled back due to application error.
Transaction Time Error Rollback Percent	B074_TranTimErrRbPct	Percentage of transactions rolled back due to a timeout error.
Transaction System Error Rollback Percent	B075_TranSysErrRbPct	Percentage of the transactions rolled back, based on system error.
Transaction Throughput Rate	B076_TranThruRate	Number of transactions processed per second.
Transaction Heuristic Count	B077_TranHeurCnt	Percentage of transactions returning a heuristic decision.

Removing the WebLogic SPI Grapher Package

To remove the WebLogic SPI Grapher package, follow these steps:

- 1 Insert the HP Operations Smart Plug-ins DVD into the DVD drive of the Windows system running Performance Manager. The HP Operations Manager InstallShield Wizard opens.
- 2 Click **Next**. The Program Maintenance window opens.
- 3 Click **Remove Products**. The Product Selection window opens.
- 4 From the options listed select the **Graphs** option of BEA WebLogic and click **Next** till the Remove the Selected Products window opens.
- 5 Click **Remove**.

6 User-Defined Metrics

The WebLogic SPI can collect data on roughly 55 metrics. However, you can expand that number by adding your own. The advantage of defining your own metrics is that you can monitor your own applications.

You can register application MBeans with the WebLogic MBean server and create user-defined metrics (UDMs) that instruct the WebLogic SPI to gather data from these MBeans.



A custom MBean must expose a “Name” attribute. The WebLogic SPI uses this name as the identifying name for the MBean. If your custom MBean is a multi-instance MBean, each MBean instance must have a unique value in its “Name” attribute. For example, WebLogic's ServletRuntime MBeans are multi-instance because a ServletRuntime MBean is instantiated by WebLogic for each deployed servlet. The Name attribute of the MBean identifies the servlet that the MBean is monitoring.

See the JMX documentation for more information about creating MBeans. Also see the WebLogic documentation for more information about registering MBeans.

You must understand the metric definitions DTD before creating UDMs. The sections that follow assume you are familiar with XML (extensible markup language) and DTDs (Document Type Definitions).

Metric Definitions DTD

The `MetricDefinitions.dtd` file provides the structure and syntax for the XML file that you create. The WebLogic SPI uses this DTD file to parse and validate the XML file you create. Following sections describe the `MetricDefinitions.dtd` file and provide an example XML file.

On a managed node, the `MetricDefinitions.dtd` file is located in the following directory:

Operating System Directory

UNIX /*<AgentDir>*/wasspi/wls/conf/

Windows *<AgentDir>*\wasspi\wls\conf\

For HPOM for Windows 8.00, *<AgentDir>* is typically `/var/opt/OV/` or `/var/lpp/OV/` for UNIX and `\Program Files\HP OpenView\data\` for Windows.



Because the `MetricDefinitions.dtd` file is used at runtime, you should not edit, rename, or move it.

`MetricDefinitions.dtd` consists of the following elements:

- `MetricDefinitions`
- `Metric`
- `MBean`
- `FromVersion/ToVersion`
- `Calculation/Formula`

The MetricDefinitions Element

The `MetricDefinitions` element is the top-level element within the `MetricDefinitions.dtd` file. It contains one collection of metrics, consisting of one or more metric definitions.

```
<!ELEMENT MetricDefinitions (Metrics)>
<!ELEMENT Metrics (Metric+)>
```

Example

```
<MetricDefinitions>
  <Metrics>
    .
    .
    .
  </Metrics>
</MetricDefinitions>
```


The Metric Element

The Metric element represents one metric. Each metric has a unique ID (for example, WLSSPI_0701). If a user-defined metric is an alarming, graphing, or reporting metric, the metric ID must be “WLSSPI_0XXX” where XXX must be a number from 700 through 799. Otherwise, if the metric is used only within the calculation of another metric, the metric ID must begin with a letter (case-sensitive) and can be followed by any combination of letters, numbers, and underscores (for example, “mbean1”).

A Metric element contains one or more elements that represent the metric data source. Two data sources are supported: Mbeans and calculations. Each metric data source element is scanned for a FromVersion or ToVersion child element to determine which metric data source element to use for the version of the application server being monitored.

```
<!ELEMENT Metric (MBean+ | Calculation+)>
<!ATTLIST Metric id          ID          #REQUIRED
                  name        CDATA      ""
                  alarm       (yes | no)  "no"
                  report      (yes | no)  "no"
                  graph       (yes | no)  "no"
                  previous    (yes | no)  "yes"
                  description  CDATA      #IMPLIED >
```

The following table lists metric element attributes.

Attribute	Type	Required	Default	Description
id	ID	yes	--	The metric ID.
name	text	no	“no”	The metric name, used for graphing and reporting. The name can be up to 20 characters in length.
alarm	“yes” “no”	no	"no"	If yes, the metric value is sent to the agent through <code>opcmon</code> .
report	“yes” “no”	no	“no”	If yes, the metric value is logged for reporting.
previous	“yes” “no”	no	“yes”	If yes, the metric value is saved in a history file so that deltas can be calculated. If you are not calculating deltas on a metric, set this to "no" for better performance.
graph	“yes” “no”	no	“no”	If yes, the user-defined metric is graphed.
description	text	no	“”	A description of the metric.

Example

```
<Metric id="WLSSPI_0700" name="UDM_700" alarm="yes">
.
.
.
</Metric>
```

The MBean Element

The MBean element is used when the data source of the metric is an attribute of a JMX MBean. The MBean element contains the following elements:

- **ObjectName** - the JMX-compliant object name of the MBean. The object name can include JMX-compliant pattern matching.
- **Attribute** - the MBean attribute name.
- **AttributeValueMapping** (optional) - numeric values that should be substituted for the values returned by the MBean attribute. This can be used to convert string attributes to numbers so they can be compared to a threshold. Each AttributeValueMapping contains one or more **Map** elements. Each Map element specifies one value to be mapped.
- **AttributeFilter** (optional) - provides basic filtering of MBeans based on MBean attribute values.
- **FromVersion/ToVersion** (optional) - the versions of the WebLogic Server for which the MBean element is valid. See [FromVersion and ToVersion Elements](#) on page 76 for more information.

```
<!ELEMENT MBean (FromVersion?, ToVersion?, ObjectName,
                 Attribute,AttributeValueMapping?,
                 AttributeFilter*)>
<!ATTLIST MBean instanceType (single | multi) "single"
                 dataType      (numeric | string) "numeric" >

<!ELEMENT ObjectName (#PCDATA)>

<!ELEMENT Attribute (#PCDATA)>

<!ELEMENT AttributeValueMapping (Map+)>
<!ELEMENT Map EMPTY>
<!ATTLIST Map from CDATA #REQUIRED
               to   CDATA #REQUIRED >

<!ELEMENT AttributeFilter EMPTY>
<!ATTLIST AttributeFilter type (include | exclude) "include"
                           name      CDATA #REQUIRED
                           operator (initialSubString |
                                   finalSubString |
                                   anySubString | match |
                                   gt | geq | lt | leq | eq)
                           #REQUIRED
                           value     CDATA #REQUIRED >
```

The following table lists MBean element attributes.

Attribute	Type	Required	Default	Description
instanceType	"single" "multi"	No	"single"	Indicates if there are multiple instances of this MBean.
dataType	"numeric" "string"	no	"numeric"	Indicates if the value returned from the MBean attribute is a string or a numeric value.

The following table lists Map element attributes.

Attribute	Type	Required	Default	Description
from	text	yes	no default	The value that is to be mapped.
to	text	yes	no default	The new metric value to be returned in place of the mapped value.

The following table lists AttributeFilter element attributes.

Attribute	Type	Required	Default	Description
type	“include” “exclude”	no	“include”	Specifies if an MBean that matches this filter should be included or excluded from consideration by the data collector.
name	text	yes	no default	Specifies the MBean attribute on which to apply the filter.
operator	“initialSubString” “finalSubString” “anySubString” “match” “gt” “geq” “lt” “leq” “eq”	yes	no default	Specifies the filter to apply. “initialSubString”, “finalSubString”, “anySubString”, and “match” can be used with MBean attributes that return text values. “gt”, “geq”, “lt”, “leq”, “eq” can be used for MBean attributes that return numeric values. See the JMX documentation for more information about filtering MBeans.
value	text or number	yes	no default	Specifies the value to compare. The metric definition creator is responsible for making sure the value data type matches the data type of the corresponding MBean attribute.

Example

```
<MBean instanceType="multi">
  <FromVersion server="6.0" update="1"/>
  <ObjectName>*:*,Type=ExecuteQueueRuntime</ObjectName>
  <Attribute>PendingRequestCurrentCount</Attribute>
</MBean>
```

The above example indicates that the collector collects metric data about the attribute `PendingRequestCurrentCount` of the Mbean `*:*,Type=ExecuteQueueRuntime`. This data is collected only if the server version is 6.0 or above.

FromVersion and ToVersion Elements

The FromVersion and ToVersion elements are used to specify the version of the WebLogic Server for which the data source element is valid.

The following algorithm is used for determining which application server version is supported by each metric data source element within the Metric element.

- If a FromVersion element is not present, no lower limit exists to the server versions supported by this metric.
- If a FromVersion element is present, the server attribute indicates the lowest server version supported by this metric. If an update attribute exists, it qualifies the lowest server version supported by specifying the lowest service pack or patch supported for that version.
- If a ToVersion element is not present, no upper limit exists to the server versions supported by this metric.
- If a ToVersion tag is present, the server attribute indicates the highest server version supported by this metric. If an update attribute exists, it qualifies the server version supported by specifying the highest service pack or patch supported for that version.

```
<!ELEMENT FromVersion (EMPTY)>
<!ELEMENT ToVersion (EMPTY)>

<!ATTLIST FromVersion  server CDATA #REQUIRED
                        update CDATA  "*">
<!ATTLIST ToVersion   server CDATA #REQUIRED
                        update CDATA  "*">
```

The following table lists FromVersion and ToVersion element attributes.

Attribute	Type	Required	Default	Description
server	numeric string	yes	none	Specifies a primary server version; for example, <code><FromVersion server="6.0"/></code>
update	numeric string	no	"*"	Specifies a secondary server version, such as "1" for service pack 1. A "*" indicates that the metric is valid for all secondary server versions.

Example

```
<FromVersion server="6.0"/>
<ToVersion server="6.999"/>
```

Calculation and Formula Elements

The Calculation element is used when the data source of the metric is a calculation using other defined metrics. The Calculation element contains a Formula element whose content is a string that specifies the mathematical manipulation of other metric values to obtain the final metric value. The metrics are referred to in the calculation expression by their metric ID. The result of the calculation is the metric value.

```
<!ELEMENT Calculation (FromVersion?, ToVersion?,Formula)>
<!ELEMENT Formula (#PCDATA)>
```

Syntax

Calculations must use the following syntax:

- Operators supported are +, -, /, *, and unary minus.
- Operator precedence and associativity follows the Java model.
- Parentheses can be used to override the default operator precedence.
- Allowable operands are metric IDs and literal doubles.

A metric ID can refer to either an MBean metric or another calculated metric. Literal doubles can be specified with or without the decimal notation. The metric ID refers to the `id` attribute of the Metric element in the metric definitions document.

Functions

The calculation parser also supports the following functions. All function names are lowercase and take a single parameter which must be a metric ID.

- `delta` returns the result of subtracting the previous value of the metric from the current value.
- `interval` returns the time in milliseconds that has elapsed since the last time the metric was collected.
- `sum` returns the summation of the values of all the instances of a multi-instance metric.
- `count` returns the number of instances of a multi-instance metric.

Examples

In the following example the value of the metric is the ratio (expressed as a percent) of Metric_1 to Metric_3.

```
<Formula>(Metric_1 / Metric_3) *100</Formula>
```

The following example shows how to define a metric that is a rate (number of times per second) for Metric_1.

```
<Formula>(delta (Metric_1) /interval (Metric_1)) *1000</Formula>
```

Sample 1

Metric 10 uses metric “mbean1” in its calculation. This calculated metric applies to all WebLogic Server versions. However, the MBean metric on which it is based has changed. Originally the MBean for metric 10 was introduced on server version 6.0, service pack 1. However in version 6.1, the attribute name changed, and this change remains the same up to the current server version.

```
<Metric id="mbean1" alarm="no">
  <MBean>
    <FromVersion server="6.0" update="1"/>
    <ToVersion server="6.099"/>
    <ObjectName>*:*,Type=ExecuteQueue</ObjectName>
```

```

        <Attribute>ServicedRequestTotalCount</Attribute>
    </MBean>
    <MBean >
        <FromVersion server="6.1"/>
        <ObjectName>*:*,Type=ExecuteQueue</ObjectName>
        <Attribute>ServicedRequestCount</Attribute>
    </MBean>
</Metric>
<Metric id="WLSSPI_0710" alarm="yes">
    <Calculation>
        <Formula>
            (delta(mbean1) / interval(mbean1))*1000
        </Formula>
    </Calculation>
</Metric>

```

Sample 2

Metric 10 should have a per-minute rate instead of a per-second rate as of server version 7.0 (based on Sample 1). Note that the versions supported by the base metrics and calculated metrics are not necessarily in sync.

```

<Metric id="mbean1" alarm="no">
    <MBean>
        <FromVersion server="6.0" update="1"/>
        <ToVersion server="6.099"/>
        <ObjectName>*:*,Type=ExecuteQueue</ObjectName>
        <Attribute>ServicedRequestTotalCount</Attribute>
    </MBean>
    <MBean>
        <FromVersion server="6.1"/>
        <ObjectName>*:*,Type=ExecuteQueue</ObjectName>
        <Attribute>ServicedRequestCount</Attribute>
    </MBean>
</Metric>
<Metric id="WLSSPI_0710" alarm="yes">
    <Calculation>
        <FromVersion server="6.0"/>
        <ToVersion server="6.999"/>
        <Formula>
            (delta(mbean1) / interval(mbean1))*1000
        </Formula>
    </Calculation>
    <Calculation>
        <FromVersion server="7.0"/>
        <Formula>
            (delta(mbean1) / interval(mbean1))*1000 * 60
        </Formula>
    </Calculation>
</Metric>

```

Sample 3: Metric Definitions File

The following sample metric definitions file illustrates how you may create your own user-defined metrics. This sample file also contains examples of calculated metrics.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE MetricDefinitions SYSTEM "MetricDefinitions.dtd">
<!-- sample UDM metrics configuration File -->
<MetricDefinitions>
  <Metrics>
<!-- The following metrics illustrate some of the options
    available when creating user-defined metrics.
-->
    <!-- The following metric uses an MBean that can have
        multiple instances in the MBean server. Note that
        JMX-compliant pattern-matching can be used in the
        MBean ObjectName tag.
-->
    <Metric id="WLSSPI_0700" name="UDM_700" alarm="yes">
      <MBean instanceType="multi">
        <FromVersion server="6.0" update="1"/>
        <ObjectName>*:*,Type=ExecuteQueueRuntime</ObjectName>
        <Attribute>PendingRequestCurrentCount</Attribute>
      </MBean>
    </Metric>
    <!-- The following 2 metrics are "base" metrics.
        They are used in the calculation of a "final"
        metric and are not alarmed, reported, or graphed
        themselves. Base metrics may have an 'id' that
        begins with a letter (case-sensitive) followed by
        any combination of letters, numbers, and underscore.
        Base metrics normally have alarm="no".
-->
    <Metric id="JVM_HeapFreeCurrent" alarm="no" >
      <MBean instanceType="single">
        <FromVersion server="6.0" update="1"/>
        <ObjectName>*:*,Type=JVMRuntime</ObjectName>
        <Attribute>HeapFreeCurrent</Attribute>
      </MBean>
    </Metric>
    <Metric id="JVM_HeapSizeCurrent" alarm="no">
      <MBean>
```

```

    <FromVersion server="6.0" update="1"/>
    <ObjectName>*:*,Type=JVMRuntime</ObjectName>
    <Attribute>HeapSizeCurrent</Attribute>
  </MBean>
</Metric>
<!-- The following metric illustrates a calculated metric.
      The calculation is based on the previous 2 "base"
      metrics.
-->
<Metric id="WLSSPI_0705" name="B705_JVMMemUtilPct"
alarm="yes" graph="yes">
  <Calculation>
    <FromVersion server="6.0" update="1"/>
    <Formula>((JVM_HeapSizeCurrent-JVM_HeapFreeCurrent)
      /JVM_HeapSize Current)*100</Formula>
  </Calculation>
</Metric>
<!-- The following metric illustrates a mapping from the
      actual string value returned by the MBean attribute to
      a numeric value so that an alarming threshold can be
      specified in a metric policy. Note that the 'datatype'
      must be specified as 'string'.
-->
<Metric id="WLSSPI_0701" alarm="yes" report="no">
  <MBean dataType="string">
    <ObjectName>*:*,Type=ServerRuntime</ObjectName>
    <Attribute>State</Attribute>
    <AttributeValueMapping>
      <Map from="Running" to="1"/>
      <Map from="Shutdown Pending" to="2"/>
      <Map from="Shutdown In Progress" to="3"/>
      <Map from="Suspended" to="4"/>
      <Map from="Unknown" to="5"/>
    </AttributeValueMapping>
  </MBean>
</Metric>
<!-- Metric IDs that are referenced from the collector
      command line must have a namespace prefix followed by
      4 digits. The default namespace prefix is 'WLSSPI_'.

```


The 'namespace' option must be used on the command line for the following metric since this metric has a different prefix other than 'WLSSPI_'.

Example:

```
wasspi_wls_ca -c FIRST_CLIENT_60-5MIN  
-x namespace=Testing_ -m 992 ...
```

-->

```
<Metric id="Testing_0992" name="Testing_Metric" alarm="yes">  
  <MBean>  
    <ObjectName>*:*,Type=ServerRuntime</ObjectName>  
    <Attribute>OpenSocketsCurrentCount</Attribute>  
  </MBean>  
</Metric>  
</Metrics>  
</MetricDefinitions>
```

Creating User-Defined Metrics

To create UDMs, complete the following tasks in the specified order.

Task 1: Disable Graphing (if Enabled)

If graphing is enabled, disable it:

- 1 From the HPOM console, select **Operations Manager** → **Nodes**.
- 2 Right-click the node on which you want to disable UDM graphing and select **All Tasks** → **Launch Tool** → **UDM Graph Disable**.

Task 2: Create a Metric Definitions File

The metrics definition file you create must be an XML file that follows the format defined by the metric definitions DTD file described in [Metric Definitions DTD](#) on page 72.



Do not edit, rename, or move the `MetricDefinitions.dtd` file installed with the WebLogic SPI.

A sample metric definitions file is installed on the managed node:

UNIX	<code>/<AgentDir>/wasspi/wls/conf/UDMMetrics-sample.xml</code>
Windows	<code>\<AgentDir>\wasspi\wls\conf\UDMMetrics-sample.xml</code>

For HPOM for Windows 8.00, `<AgentDir>` is typically `/var/opt/OV/` or `/var/lpp/OV/` for UNIX and `\Program Files\HP Openview\data\` for Windows.

Task 3: Configure the Metric Definitions File Name and Location

For the UDM data collection to occur, the WebLogic SPI configuration must include the name and location of the metric definitions file, as shown below:

```
UDM_DEFINITIONS_FILE = <full path of user metric definitions file>
```

The path name should use only forward slashes (“/”).

To add the UDM file name and its location to the WebLogic SPI configuration, follow these steps:

- 1 From the HPOM console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **SPI Admin**.
- 2 Double-click **Configure WLSSPI**.
- 3 Select the managed nodes on which the metrics definition file exists and click **Launch**. The Console Status window opens.

After some time the Configure WLSSPI Tool Introduction window opens. Read the information and click **Next**. The configuration editor opens.

- 4 If the metrics definition file uses the same name and location on all managed nodes, configure the `UDM_DEFINITIONS_FILE` property at the Defaults (global properties) level. Otherwise, set the property for each managed node selected in step 3:
 - a Click **Default Properties** at the Defaults level or for a node.

- b Click the **Set Configuration Properties** tab.
- c From the Select a Property to Add dropdown menu, select **UDM_DEFINITONS_FILE** and click **Add Property**.
- d Type the value (metric definitions file name and its absolute path name, using forward slashes in only the path name).
- e Click **Save** to save the changes.
- f Click **Next**. The Confirm Operation window opens.
- g Click **OK** to save changes and exit the configuration editor.

The changes you made to managed nodes that were not selected are saved to the configuration on the management server. However, to configure those managed nodes, you must deploy the WLSSPI Service Discovery policy on these nodes.

Task 4: Create a UDM Policy Group and Policies

To run the UDM data collection and establish thresholds for alarming, create a UDM policy group and policies:

- 1 Copy an existing WebLogic SPI policy group:
 - a From the HPOM console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WLSSPI**.
 - b Right-click the policy group you want to use as a starting point, and select **Copy**.
 - c Right-click **WLSSPI** and select **Paste**.
- 2 Rename the new policy group depending on how you want to identify the new metric and collector policies. For example, you could include UDM in the name to clearly indicate that the group consists of custom metric monitors.
 - a Right-click the policy group and select **Rename**.
 - b Type the new name.
- 3 Edit and rename each policy in the new group:
 - a Double-click the policy you want to use.
 - b Configure the collector policy command line (in the Command text box) to include the policy name and UDM metric number. For more information, see [Advanced Policy Customizations](#) on page 37.
 - c Configure thresholds in the policy, as appropriate. For more information, see [Advanced Policy Customizations](#) on page 37.
 - d Click **File** → **Save As**, and rename the policy according to the naming scheme.

The name you assign to the new metric policy in the group may contain each new UDM number. For example, a copy of WLSSPI_0001 can be called WLSSPI_0701.

The name you assign to the new collector policy must also contain the identifying name.
- 4 Select all the original policies from the new group and press the **Delete** key.

Task 5: Deploy the Policy Group

- 1 Right-click the new policy group and select **All Tasks** → **Deploy on**.
- 2 Select the nodes on which you want to deploy the policy group.
- 3 Click **OK**.

Task 6: Enable Graphing

If you are using the graphing product HP Performance Manager, enable data collecting for UDM graphing:

- 1 From the HPOM console, select **Operations Manager** → **Nodes**.
- 2 Right-click the node on which you want to enable UDM graphing and select **All Tasks** → **Launch Tool** → **UDM Graph Enable**.

Allow sufficient collection intervals before attempting to view graphs.

7 Troubleshooting the WebLogic SPI

This chapter covers basic troubleshooting for the WebLogic SPI. The Error messages section of the WebLogic SPI online help lists error messages by number.

The Self-Healing Info Tool

The Self-Healing Info tool gathers troubleshooting information about the SPI and stores it in a file that you can submit to HP support for assistance. For more information about this tool, see the WLSSPI Admin tools section under Tools in the WebLogic SPI online help.



The file created by the Self-Healing Info tool may be hidden on some Windows managed nodes. If you do not see the file, open Windows Explorer and from the **Tools** menu, select **Folder Options**. Click the **View** tab. Under Hidden files and folders, select **Show hidden files and folders**.

Log and Trace Files

Log and trace files are maintained on the managed nodes. You can gather troubleshooting information about the WebLogic SPI from the data logged in these log and trace files.

UNIX Managed Nodes

The following log and trace files are found on the managed nodes running on UNIX (typically, *<AgentDir>* is `/var/opt/OV/` or `/var/lpp/OV/`):

File Type	Log
Directory	<i><AgentDir></i> /wasspi/wls/log/config.log
Description	Output from the WebLogic SPI configuration scripts is recorded in this log file.

File Type	Log
Directory	<i><AgentDir></i> /wasspi/wls/log/errorlog
Description	WebLogic SPI logs the error messages in this file. This log file is monitored by WebLogic SPI policies.

File Type	Log
Directory	<i><AgentDir></i> /wasspi/wls/log/wasspi_wls_discovery.log
Description	Output from the WebLogic SPI discovery process is recorded in this log file.

File Type	Trace
Filename	<i><AgentDir></i> /wasspi/wls/log/wasspi_wls_discovery.trc (archived files have a three digit number appended to the filename)
Description	Discovery binary trace file used by your HP support representative. By default, tracing to this file is enabled. To disable tracing, in <i><AgentDir></i> /bin/instrumentation/wasspi_wls_discovery.pl, set the <code>\$trace_on</code> variable to 0. To enable this trace, set the <code>\$trace_on</code> to 1. When instrumentation is deployed, the <code>wasspi_wls_discovery.pl</code> file is overwritten (therefore, if you disable tracing, it becomes enabled when instrumentation is deployed). Five archived versions are kept. A new trace file is created when the discovery policy is run.

File Type Trace

Directory /<AgentDir>/wasspi/wls/log/trace.log (archived files have a three digit number appended to the filename)

Description The HP support representative uses this trace file. This file gives information about the CollectorServer, regardless of whether the Collector is set to PERSISTANT or TRANSIENT mode in the SPIConfig file. The default value of the Collector Mode is 'PERSISTENT'.

By default, tracing to this file is disabled. To enable this tracing, run the Start Tracing tool.

File Type Trace

Directory /<AgentDir>/wasspi/wls/log/traceCollectorClient.log (archived files have a three digit number appended to the filename)

Description Trace file used by your HP support representative. This file gives information about the CollectorClient when the Collector is set to 'PERSISTENT' mode in the SPIConfig file. The default value of the Collector Mode is 'PERSISTENT'.

By default, tracing to this file is disabled. To enable this tracing, use the Start Tracing tool.

Windows Managed Nodes

The following log and trace files are maintained on the Windows managed nodes (for HPOM 8.00, typically, <AgentDir> is \Program Files\HP OpenView\Data:

File Type Log

Directory \<AgentDir> \wasspi\wls\log\config.log

Description Records output from configuration scripts.

File Type Log

Directory \<AgentDir> \wasspi\wls\log\errorlog

Description Records WebLogic SPI error messages. This log file is monitored by WebLogic SPI policies.

File Type Log

Directory \<AgentDir> \wasspi\wls\log\wasspi_wls_discovery.log

Description Records output from the WebLogic SPI discovery process.

File Type Trace

Filename *<AgentDir>*\wasspi\wls\log\wasspi_wls_discovery.trc
(archived files have a three digit number appended to the filename)

Description Discovery binary trace file used by your HP support representative. By default, tracing to this file is enabled. To disable tracing, in *<InstallDir>*\bin\instrumentation\wasspi_wls_discovery.pl, set the \$trace_on variable to 0. To enable this trace, set the \$trace_on to 1. When instrumentation is deployed, the wasspi_wls_discovery.pl file is overwritten (therefore, if you disable tracing, it becomes enabled when instrumentation is deployed). Five archived versions are kept. A new trace file is created when the discovery policy is run.

File Type Trace

Directory *<AgentDir>* \wasspi\wls\log\trace.log (archived files have a three digit number appended to the filename)

Description Trace file used by the HP support representative. This file gives information about the CollectorServer, regardless of whether the Collector is set to PERSISTANT or TRANSIENT mode in the SPIConfig file. The default value of the Collector Mode is 'PERSISTENT'.

By default, tracing to this file is disabled. To enable this tracing, run the Start Tracing tool.

File Type Trace

Directory *<AgentDir>*\wasspi\wls\log\traceCollectorClient.log
(archived files have a three digit number appended to the filename)

Description Trace file used by your HP support representative. This file gives information about the CollectorClient when the Collector is set to 'PERSISTENT' mode in the SPIConfig file. The default value of the Collector Mode is 'PERSISTENT'

By default, tracing to this file is disabled. To enable this tracing, use the Start Tracing tool.

Troubleshooting the Discovery Process

Problem

The WLSSPI Discovery policies do not automatically discover and update the WebLogic SPI configuration.

Solutions

To troubleshoot the discovery process, do one or more of the following (as applicable):

- Check for errors in the message browser of the managed nodes not being discovered. Follow the instruction text of any error messages displayed.
- Check for errors in the `<AgentDir>/wasspi/wls/log/wasspi_wls_discovery.log` file on the managed node.
- If you have more than one version of the WebLogic Server installed on a managed node, set the `JAVA_HOME` property to the directory where the highest version of Java is installed. If you are running WebLogic Server version 8.1, you must use Java version 1.4.1 or higher.
- If you have both WebLogic Server version 6.1 and 8.1 installed on the same system, set the `HOME` property and run the Discover WebLogic tool
- Check if the WLSSPI Discovery policies are being deployed:

From the HPOM console, select **Operations Manager** → **Policy management** → **Deployment jobs**.

- If the state of a WLSSPI Discovery policy is `Active`, the policy is still being deployed. Wait for the deployment of the policy to complete.
- If the state of a WLSSPI Discovery policy is `Suspended` or `Error`, then check for any error messages in the message browser and continue to troubleshoot the problem by reading the rest of this section.
- If the WLSSPI Discovery policies are not listed, check the message browser for the following messages:

```
WASSPI-302: Updating WLS SPI configuration in HPOM server for <node>  
WASSPI-303: The SPI configuration for <node> was updated by discovery  
in the HPOM server. The updated configuration is as shown below
```

If these messages are present, the WLSSPI Discovery policies are successfully deployed. If these messages are not present, either the policies were not successfully deployed or the `AUTO_DISCOVER` check box has not been selected in the configuration editor.

Continue to troubleshoot the problem by reading the rest of this section.

- Verify that a WebLogic application server is installed on the managed node. If an application server is not installed, uninstall the WLSSPI Discovery policy group from the managed node, install an application server, and complete the configuration tasks listed in [Chapter 3, Configuring the WebLogic SPI](#).
- Verify the WebLogic application server status. The application server must be running. See [Task 2: Verify the Application Server Status](#) on page 15 for more information.
- Verify that the `LOGIN/PASSWORD` properties are set (see the WebLogic SPI online help) and that the WebLogic user configured has the correct permissions. See [Task 3: Collect WebLogic Login Information](#) on page 16.

- On a Windows managed node, if the `HKEY_LOCAL_MACHINE\\Software\\BEA Systems\\BEAHOMELIST` registry key does not exist, either configure it, create the file `SystemDrive\BEA\beahomelist`, or configure the `BEA_HOME_LIST` property for that managed node.
- Verify the Java home directory. See [Verifying the Java Home Directory](#) on page 92.
- Verify that the discovery agent is running on the managed node:
 - a Run the command `opcagt -status`
 - b Look for the following:


```
Service Discovery Agent OvSvcDiscAgent.cmd (1084) is running
```

If the agent is not running, start it by running the command `opcagt -start -id 13`
- If you are running WebLogic Server 7.0 or higher and did not save the domain configuration file (for example, `config.xml`) in the default directory (`<BEA_Home_Dir>/user_projects/<WebLogic_Domain_X>/`, where `<BEA_Home_Dir>` is the directory that contains the `registry.xml` file), do one of the following:
 - Manually set the server using the Configure WLSSPI tool.
 - Manually configure `ADMIN_PORTS`, the port number(s) of the WebLogic Admin server(s) listed in the domain configuration file, using the Configure WLSSPI tool. The global `LOGIN` and `PASSWORD` must be configured for the node on which these WebLogic Admin servers are running.
- On a UNIX managed node, verify that `BEA_HOME_LIST` and `HOME_LIST` directory path names do not include spaces. The discover process currently does not support spaces in directory names.
- If you deployed the Discovery policies at the WLSSPI Discovery level or not in the order shown in [Manually Deploying the Discovery Policies](#) on page 92, uninstall and redeploy the Discovery policies:
 - a From the HPOM console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server**.
 - b Right-click **WLSSPI Discovery** and select **All Tasks** → **Uninstall all**. The Uninstall policies on... window opens.
 - c Select the nodes from which to uninstall the Discovery policies and click **OK**.
 - d Follow the steps given in the section [Manually Deploying the Discovery Policies](#) on page 92 to redeploy the discovery policies. Deploy the policies in the order mentioned and not as a group. If you deploy the policies as a group, the policies may not be deployed in the correct order.
- Verify that the Configure WLSSPI tool is not running or a configuration is not open in an editor. Only one process can access a configuration at a time. If a configuration is open, other processes that must access that file (like the discovery policy) hang until the file becomes available.

Other Discovery Related Problems

Problem	The WLSSPI Discovery policies are adding inaccurate information to the configuration.
Solution	<ul style="list-style-type: none">• Verify LOGIN and PASSORD are correct. For more information, see Task 3: Collect WebLogic Login Information on page 16.• Verify the Java home directory. For more information, see Verifying the Java Home Directory on page 92. <p>Update the configuration and clear the AUTO_DISCOVER check box in the configuration editor to prevent the WLSSPI Discovery policies from overwriting the configuration information.</p>
Problem	Two or more WebLogic domains have managed WebLogic Servers on the same managed node.
Solution	<ol style="list-style-type: none">1 From the HPOM console, select Operations Manager → Policy management → Deployment jobs.2 Find the jobs in an <code>Error</code> state.3 For each job you want to restart, right-click it and select All Tasks → Restart job
Problem	The following error message appears: <pre>PMD51) Error: Unable to deploy instrumentation files from directory <directory_name>: (NUL16389E) Unspecified error (0x80004005). Please check the error log on the managed node.</pre>
Solution	<ol style="list-style-type: none">1 From the HPOM console, select Operations Manager → Policy management → Deployment jobs.2 Find the jobs in an <code>Error</code> state. <p>For each job you want to restart, right-click it and select All Tasks → Restart job.</p>
Problem	The property of critical error messages in the HPOM console is: Errors occurred during the distribution of the monitors. Solve the problems and distribute the monitors again. (OpC30-1030).
Solution	<ol style="list-style-type: none">1 From the HPOM console, select Operations Manager → Policy management → Deployment jobs.2 Find the jobs in an <code>Error</code> state.3 Right-click each job you want to restart and select All Tasks → Restart job.

Manually Deploying the Discovery Policies

If the WLSSPI Discovery policies are not deployed successfully when you run the Discover WebLogic tool, you can manually deploy the policies on the managed nodes on which the WebLogic Admin Servers are running (you *must* deploy the policies in the given order only):

- 1 From the HPOM console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WLSSPI Discovery**.
- 2 Right-click **WLSSPI-Messages** and select **All Tasks** → **Deploy on**. The Deploy Policies on... window opens.
- 3 Select the nodes on which you want to deploy the auto-discovery policies and click **OK**.
- 4 Right-click **WLSSPI Service Discovery** and select **All Tasks** → **Deploy on**. The Deploy Policies On... window opens.
- 5 Select the nodes on which you want to deploy the auto-discovery policies and click **OK**.

Verifying the Java Home Directory

The WebLogic SPI Collector is dependant on the Java home directory information. Collector will not work if the Java Home directory information is inaccurate or not available. The Java Home directory must, therefore, be configured properly on both Windows and UNIX managed nodes.

In order to successfully use the WLSSPI Discovery policies, the Java home directory (on both a Windows and UNIX managed node) *must* be configured correctly.

Although the discovery policies search for this information, if they cannot find this information or the information is not accurate, the discovery policies do not function completely.

On each managed node on which you want to run the discovery policies, verify one of the following (listed in the order of precedence used by the discovery policies):

For the Collector to work properly, ensure one of the following:

- **JAVA_HOME** is correctly defined in the configuration. To edit or view the configuration, run the Configure WLSSPI tool:
 - a From the HPOM console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **SPI Admin**.
 - b Double-click **Configure WLSSPI**. The Edit Parameters window opens.
 - c Select the nodes to configure and click **Launch**. The Console Status window and then the configuration editor opens.
 - d In the configuration editor, set the **JAVA_HOME** property. For more information about setting the property, see the Configuring WLSSPI section in the WebLogic SPI online help.
 - e Run the Discover WebLogic tool on the managed nodes on which the **JAVA_HOME** property was added or edited. Running the Discover WebLogic tool updates the service map.
- Java is installed in each of the BEA home directories (each directory listed in the file `beahomelist`).
- The **JAVA_HOME** system variable is correctly defined.

On a Windows managed node, follow these steps:

- a Select **Start** → **Settings** → **Control Panel**.
- b Double-click **System**.
- c Select the **Advanced** tab.
- d Select **Environment Variables...**
- e Scroll through the System variables list. Verify the `JAVA_HOME` value. If `JAVA_HOME` does not exist, it is not defined.

On a UNIX managed node, type:

```
echo $JAVA_HOME
```

Verify the output. If no output is returned, `JAVA_HOME` is not defined.

Troubleshooting the Configuration

Problem	The WebLogic SPI configuration does not have complete or accurate information for a WebLogic managed server.
Solution	Verify LOGIN and PASSWORD are correct. For more information see, Task 3: Collect WebLogic Login Information on page 16 and the WebLogic SPI online help. This is the most common reason for incorrect information for a WebLogic managed server running on a remote node (not running on the HPOM managed node).
Problem	The WLSSPI Discovery policies overwrite the configuration with inaccurate information.
Solution	Update the configuration and clear the AUTO_DISCOVER check box in the configuration editor to prevent the WLSSPI Discovery policies from overwriting the configuration information.
Problem	The <code>Server status is unknown (down)</code> message appears in the message browser but the server is running
Solution	Check that you have correctly set the PORT, PROTOCOL, and/or PASSPHRASE properties: <ul style="list-style-type: none">• Verify that PROTOCOL is set to one of two values: t3 (for non-SSL) or t3s (for SSL).• If the application server is using SSL, verify that the PORT is set to a valid SSL port number and that PROTOCOL is set to t3s.• If the application server is not using SSL, verify that the PORT is set to a valid non-SSL port number and that PROTOCOL is set to t3.• If the keystore has a password defined, re-set the PASSPHRASE in case it has been mis-typed.

Verifying the Node Name

Verify that the node name specified in a node or group block matches the primary node name configured in HPOM. To display the primary node name, follow these steps:

- 1 From the HPOM console, select **Operations Manager** → **Nodes**.
- 2 Right-click the node and select **Properties**.
- 3 Select the **Network** tab.

Troubleshooting the Tools

Message	Configuration variable SERVER<n>_START_CMD missing for server "Default Server"
Solution	To successfully run the Start WebLogic tool, you must set the START_CMD property. Set this property using the Configure WLSSPI tool. For more information about this tool, see the WebLogic SPI online help.
Message	Configuration variable SERVER<n>_STOP_CMD missing for server "Default Server"
Solution	To successfully run the Stop WLSSPI tool, you must set the STOP_CMD property. Set this property using the Configure WLSSPI tool. For more information about this tool, see the WebLogic SPI online help.
Problem	Check WebLogic tool showing a wrong status for a server instance.
Solution	<p>If a server is up and running but Check WebLogic tool returns the server status as NOT_RUNNING, then turn ON the monitoring for that particular server by following these steps:</p> <ol style="list-style-type: none">1 Select Tools → SPI for WebLogic Server → SPI Admin → Start Monitoring.2 Select the server instance for which the status is shown as NOT_RUNNING. Ensure that Monitoring is ON.3 Relaunch the Check WebLogic tool and verify the server status.
Problem	When launching the tools, the tools hang or there is no output.
Solution	The tools will not work if the memory is low. Check the performance of the node and the management server. The physical memory available must be more than 500 MB.

Problem	Verify tool lists files and directories related to the management server as missing. For example:
	<pre> / MGMT_SERVER/SPI-Share/wasspi/wls/bin/parseDefs.pl / MGMT_SERVER/SPI-Share/wasspi/wls/bin/ processWASSPIDiscovMsg.pl / MGMT_SERVER/SPI-Share/wasspi/wls/conf </pre>
Solution	This is a known problem. The verify tool lists management server related files if you install the WebLogic Server on the management server itself. This problem occurs if both the managed node and the management server are the same.
Problem	Check WebLogic tool does not give any output.
Solution	Ensure that the Collector is running for the WebLogic Server instance on that node.
Problem	The “All” option of the Start WebLogic tool does not start all the WebLogic servers.
Solution	This is a known problem. The “All” option of the Start WebLogic tool does not function. You must start each server individually.
Problem	Even after the Start WebLogic tool has started the WebLogic Servers, the tool output does not show the status as “finished” (the output status is “running”).
Solution	NA
Problem	When launched, the Verify tool gives improper output.
Solution	Before you launch the Verify tool ensure that you have installed the latest version of Self-Healing Service (SHS) component from the SPI DVD.
Problem	When launched, the Self-Healing Info application gives improper output.
Solution	Ensure that you have installed the latest version of Self-Healing Service (SHS) component from the SPI DVD.

Glossary

agent

A program or process running on a remote device or computer system that responds to management requests, performs management operations, or sends performance and event notification. An agent can provide access to managed objects and MIB variables, interpret policy for resources and configure resources.

application

Packaged software that provides functionality designed to accomplish a set of related tasks. An application is generally more complex than a tool.

ASCII

American Standard Code for Information Interchange.

assigned policy

A policy assigned to one or more resources in the computing environment but not yet deployed or installed on those resources.

automatic action

A pre-configured program or script executed in response to an event, message, or a change in information in the management database. without operator intervention.

client

When the context is network systems, a computer system on a network that accesses a service from another computer (server). When the context is software, a program or executable process that requests a service from a server.

client console

An instance of the user interface that appears on the client system while the application runs on a server.

command

An instruction to a computer program that causes a specified operation to be carried out. Commands are typically typed by users on a command line.

configuration

In a network context, the complete set of inter-related systems, devices and programs that make up the network. For example the components of a network may include computer systems, routers, switches, hubs, operating systems and network software. The configuration of the network determines the way that it works and the way that it is used. In a software context, the combination of settings of software parameters and attributes that determine the way the software works, the way it is used, and how it appears.

configuration file

A file that contains specifications or information that can be used for determining how a software program should look and operate.

connection

A representation of a logical or physical relationship between objects.

console

An instance of the user interface from which the user can control an application or set of applications.

customization

The process of designing, constructing or modifying software to meet the needs and preferences of a particular customer or user.

data type

A particular kind of data; for example database A repository of data that is electronically stored. Typically databases are organized so that data can be retrieved and updated.

deploy

To install and start software, hardware, capabilities, or services so that they work in the business environment.

deployed application

An application and its components that have been installed and started to work in the business environment.

deployed policy

A policy that is deployed on one or more resources in the computing environment.

deployment

The process of installing and activating software, hardware, capabilities or services so that they work in the business environment.

deployment package

A software package that can be deployed automatically and installed on a managed node.

error log

An output file containing error messages.

event

An unsolicited notification such as an SNMP trap or WMI notification generated by an agent or process in a managed object or by a user action. An event usually indicates a change in the state of a managed object or cause an action to occur.

Hypertext Transfer Protocol (HTTP).

The protocol that World Wide Web clients and servers use to communicate.

HTTPS

Hypertext Transfer Protocol Secure.

icon

An on-screen image that represents objects that can be monitored or manipulated by the user or actions that can be executed by the user.

managed object

A network, system, software or service object that is both monitored for performance, status and messages and is manipulated by means of actions in the management software.

management console

An instance of the user interface from which the user can control the management application or set of management applications. The console may be on the system that contains the management software or it may be on another system in the management domain.

management server

A server that provides management services, processes, or a management user interface to clients. A management server is a type of management station.

message

A structured, readable notification that is generated as a result of an event, the evaluation of one or more events relative to specified conditions, or a change in application, system, network, or service status.

message browser

A graphical user interface that presents notifications that are generated as a result of an event, the evaluation of one or more events relative to specified conditions or a change in application, system, network, or service status.

message description

Detailed information about an event or message.

message key

A message attribute that is a string used to identify messages that were triggered from particular events. The string summarizes the important characteristics of the event. Message keys can be used to allow messages to acknowledge other messages, and allows for the identification of duplicate messages.

message severity level

A property of a message indicating the level of impact of the event or notification that initiated the message. See also severity level.

metadata

Data that defines data.

metric

A measurement that defines a specific operational or performance characteristic.

module

A self-contained software component that performs a specific type of task or provides for the presentation of a specific type of data. Modules can interact with one another and with other software.

node

When the context is network, a computer system or device (for example, printer, router, bridge) in a network. When the context is a graphical point to point layout, a graphical element in a drawing that acts as a junction or connection point for other graphical elements.

HPOM

HP Operations Manager

parameter

A variable or attribute that may be given an arbitrary value for use during an execution of either a computer program or a procedure within a program.

parameter type

An abstraction or categorization of a parameter that determines the particular kind of data that is valid for the parameter. For example a parameter type could be IP Address which indicates that parameter values must have four numbers separated by decimals with the value for each number being in the range of 0 to 255.

parameter value

A value given to a variable.

policy

A set of one or more specifications rules and other information that help automate network, system, service, and process management. Policies can be deployed to various targets (for example, managed systems, devices, network interfaces) providing consistent, automated administration across the network.

policy management

The process of controlling policies (for example, creating, editing, tracking, deploying, deleting) for the purposes of network, system or service management.

policy type

An abstraction or categorization of policies based on the function of the policy or the services that the policy supports.

port

If the context is hardware, a location for passing information into and out of a network device. If the context is ECS, a location for passing information into and out of a correlation node.

server

If the context is hardware plus software, a computer system that provides a service (for example, management capabilities, file storage capabilities) to other computer systems

(clients) on the network. If the context is a software component, a program or executable process that responds to and services requests issued by clients.

severity level

A property of an object indicating the status of the object. Severity level is based on the impact of events or messages associated with the object.

Smart Plug-in (SPI)

Prepackaged software that installs into a management console and provides management capabilities specific to a given type of business application, database, operating system, or service.

trace log

An output file containing records of the execution of application software

Index

A

Automatic Command reports, 46

B

basic policy customizations, 33

C

CODA, 55

collection intervals

 changing for all servers, 42

 changing for selected metrics, 43

Collector, 92

collector policies

 description of, 32

create new tagged policy group, 45

customizations

 creating new policies, 45

D

deinstallation

 removing WLS-SPI, 14

directories

 locations for trace file/error logs, 87

F

files, locations on management server/managed nodes, 86, 87

G

graphs

 for UDMs, 84

 HP Performance Manager, 55

 instructions for manually generating, 66

 list of metrics for server status graph, 67

 policies available for, 67

 showing alarm conditions, 65

H

HP Performance Manager, using WebLogic SPI with, 55

HP Reporter, integrating WebLogic SPI with, 55

L

Linux, monitoring WebLogic Server installed on, 49

Log, 86

Log and Trace files, 86

log file policies

 description of, 32

M

managed nodes, 9

Manually Generated Reports, 47

messages

 policy configuration for, 36

Message Source policy groups, description of WebLogic SPI groups, 31

metric element attributes for creating user defined metrics (UDMs), 73

metric policies

 description of, 32

metrics

 modifying collections in the collector policy, 38

metrics policies

 description of, 31

O

operator actions

 graphs generated from, 65

P

Performance Manager, using WebLogic SPI with, 55

policies

 customizing message displayed for alerts, 36

 customizing message text, 36

 customizing thresholds, 34

 customizing with the tag option, 45

 description of, 31, 32

 modifying, 33

 re-installing defaults, 46

- policy groups
 - changing collection intervals, 43
 - creating custom with the tag parameter, 45
 - description, 31
- Polling Interval, 42
- proxy configured monitoring, 49

R

- remote monitoring
 - requirements for, 50
- remote systems, monitoring, 49
- remove WebLogic SPI Grapher package, 69
- remove WebLogic SPI Reporter Package, 65
- Reporter
 - integrating WebLogic SPI to work with, 55
 - setting up WLS-SPI to work with, 58, 65
- reports
 - Automatic Command, 46
 - generated from HP Reporter, 61
 - Performance Insight, 55
 - Reporter, 55
 - sample automatic action, 47
 - using HP Reporter to generate, 55
- restoring default WLS SPI policy groups, 46

S

- scheduled metrics, 42
- Self-Healing Info tool, 85

T

- tag option
 - creating custom policy groups with, 45
- Text-Based Reports, 46
- thresholds
 - customizing, 34
 - exceeded
 - viewing graphs resulting from, 65
 - settings for different servers, 43
- Troubleshooting, 85
 - Self-Healing Info tool, 85

U

- UDMs, *please see user defined metrics*
- unsupported platforms, monitoring WebLogic on, 50
- upgrading WLS-SPI, 11

- user defined metrics
 - graphing, 84
 - MBean Element, description of, 74
 - metric definitions element, description of, 72
 - metric element, description of, 73
 - metric element attributes, description of, 73
 - sample XML file for, 79

V

- verify discovery process, 20

W

- WLS-SPI
 - upgrading, 11