

HP OpenView Smart Plug-in for BEA WebLogic Server

Configuration Guide

Version: B.02.05

For Windows OpenView Operations Management Servers



August 2003

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

Legal Notices

Warranty

Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company
United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Copyright Notices

© Copyright 2003 Hewlett-Packard Development Company L.P., all rights reserved.

No part of this document may be copied, reproduced, or translated into another language without the prior written consent of Hewlett-Packard Company. The information contained in this material is subject to change without notice.

Trademark Notices

Windows® is a registered trademark of Microsoft Corporation.

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

UNIX® is a registered trademark of The Open Group.

All other product names are the property of their respective trademark or service mark holders and are hereby acknowledged.

Support

Please visit the HP OpenView website at:

<http://openview.hp.com/>

There you will find contact information and details about the products, services, and support that HP OpenView offers.

The support area of the HP OpenView website includes:

- Downloadable documentation
- Troubleshooting information
- Patches and updates
- Problem reporting
- Training information
- Support program information

contents

- Chapter 1 The Smart Plug-in for BEA WebLogic Server 9**
 - Introduction 9
 - Software Requirements 10

- Chapter 2 Installing and Removing WLS-SPI 13**
 - Installing WLS-SPI 13
 - Installing WLS-SPI and OVO 13
 - Installing WLS-SPI with OVO Installed 14
 - Removing WLS-SPI 14
 - Task 1: Uninstall All WLS-SPI Policies from the Managed Nodes 15
 - Task 2: Remove WLS-SPI Node Groups on the Management Server 15
 - Task 3: Remove the WLS-SPI Software from the Management Server . . . 16
 - Task 4: Remove WLS-SPI Policy Group from the Management Server . . 16

- Chapter 3 Configuring WLS-SPI 17**
 - Introduction 17
 - Configuration Prerequisites 17
 - Task 1: Add OVO Managed Nodes 17
 - Task 2: Verify the Application Server Status 18
 - Task 3: Collect Application Server Information 18
 - WebLogic Login and Password 19
 - OVO Managed Node Name(s) 20
 - Basic WLS-SPI Configuration 20
 - Task 1: Run WLSSPI Discover 20
 - Task 2: Verify the Discovery Process 23

Task 3: Set Additional Properties	24
Additional WLS-SPI Configuration	25
Chapter 4 Using and Customizing WLS-SPI	27
Introduction	27
Using WLS-SPI Policies	28
WLS-SPI Policy Groups	28
OVO Policy Types and WLS-SPI	29
Basic Policy Customizations	30
Modifying Metric Policies	30
Threshold Level and Actions	31
Message and Severity	35
Advanced Policy Customizations	36
Choosing Metrics To Monitor	36
Using the WebLogic Server Command Parameters	37
Changing the Collection Interval for All Scheduled Metrics	39
Changing the Collection Interval for Selected Metrics	40
Customize the Threshold for Different Servers	41
Creating Custom, Tagged Policies	42
Re-installing the WLS-SPI Policies	43
Using Policies/Tools to View Annotation Reports and Graphs	43
Automatic Command Reports	45
Reports	45
Sample Report	46
Monitoring WebLogic on Unsupported Platforms	47
Requirements for Monitoring Remote Nodes (running on Platforms not supported by WLS-SPI)	48
Overview	48
Configure Remote System Monitoring	50
Task 1: Configure the Remote WebLogic server	50
Task 2: (optional) Integrate OpenView Performance Agent (MeasureWare Agent)	51
Task 3: Assign Local Node to WLS-SPI node group	51
Configuring Remote Monitoring for Logfiles (Optional)	52
Configure the logfile policy for remote logfiles	52
Remote Monitoring Limitations	53

Chapter 5	Using HP OpenView Reporting and Graphing Features with WLS-SPI	55
	Introduction	55
	Using the OpenView Performance Agent	57
	Integrating WLS-SPI with Reporter to Generate Reports	58
	WLS-SPI Reports	60
	Removing the WLS-SPI Reporter Package	62
	Integrating WLS-SPI with HP OpenView Performance Manager	62
	Viewing Graphs that Show Alarm Conditions	63
	Viewing Graphs that Show Past/Current Conditions	63
	WLS-SPI Metrics Available for Add-On Graphs	63
	Specifying a Date Range	66
	Removing the WLS-SPI Grapher Package	67
Chapter 6	User Defined Metrics	69
	Introduction to User Defined Metrics	69
	Metric Definitions DTD	70
	The MetricDefinitions Element	70
	The Metric Element	71
	The MBean Element	72
	FromVersion and ToVersion Elements	74
	Calculation and Formula Elements	76
	Sample 1	77
	Sample 2	78
	Sample 3: Metric Definitions File	78
	Create User-defined Metrics	81
	Task 1: Disable graphing (if enabled)	81
	Task 2: Create a metric definitions file	81
	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: Restart the OVO subagent	84
	Task 7: Enable graphing	85

Chapter 7	Troubleshooting WLS-SPI	87
	Introduction	87
	Log and Trace Files	88
	Management Server	88
	UNIX Managed Nodes	88
	Windows Managed Nodes	90
	The WLSSPI Discovery Policies	92
	Manually Deploying the Discovery Policies	95
	Verifying Discovery Policy Deployment is Currently Active	95
	Verifying the Java Home Directory	96
	The Configuration	97
	Verifying the Node Name	97
	Tools	98

The Smart Plug-in for BEA WebLogic Server

Introduction

The Smart Plug-in for BEA WebLogic Server (WLS-SPI) is a full-featured SPI that allows you to manage WebLogic servers from an HP OpenView Operations console. It is suggested you read about the WLS-SPI concepts from the online help. The following topics are covered in the online help:

- Concepts
- How the WLS-SPI Works
- Components
- Configuration editor
- Tools
- Policies
- Reports and graphs
- Error messages

In addition to this configuration guide, a reference guide is also available on the *hp OpenView OV operations 7.20 for Windows start-up* CD in the `/Disk1/Documentation/SPI Guides/wlsspi_reference.pdf` file. The reference

guide contains the same information found in the online help (although some interactive pages are not included) and has been provided for your convenience.

Software Requirements

Table 1 Management Server

Component	Supported Version(s)
HP OpenView Operations for Windows	7.2

Table 2 Managed Nodes

Component	Supported Version(s)
HP OpenView Operations for Windows Agent	7.2
HP-UX	11.00 or 11.11
Solaris	2.6, 7, 8, or 9
Windows	NT 4.0 and SP5+ Windows 2000 Windows 2003 Windows XP
Windows Script Host (for NT 4.0 only) ^a	5.6
WebLogic Application Server	5.1 ^b 6.0 and sp1 or sp2 ^c 6.1 and sp1, sp2, sp3, or sp4 ^d 7.0, 7.0 and sp1 or sp2 8.1
HP OpenView Performance Agent UNIX®	C.02.00
HP OpenView Performance Agent NT	C.02.00 or higher
HP OpenView Performance Agent Windows 2000	C.03.00
HP OpenView Reporter	A.03 or higher
HP OpenView Performance Manager	4.0



- a. Go to <http://msdn.microsoft.com/library/default.asp?url=/downloads/list/webdev.asp/> to download this component. See <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/script56/html/wskonwhatiswsh.asp> for more information about Windows Script Host.
- b. WLS-SPI offers a reduced level of monitoring for WLS 5.1.
- c. Service pack 1 or 2 must be installed with WLS 6.0. WLS-SPI only supports WLS 6.0 with one of the listed service packs installed (if no service packs are installed, WLS-SPI does not support that server).
- d. Service pack 1, 2, 3, or 4 must be installed with WLS 6.1. WLS-SPI only supports WLS 6.1 with one of the listed service packs installed (if no service packs are installed, WLS-SPI does not support that server).

Installing and Removing WLS-SPI

This chapter covers installing and removing the Smart Plug-in for BEA WebLogic Server (WLS-SPI).

Installing WLS-SPI

The WLS-SPI can be installed in the following manner:

- Install WLS-SPI and OVO simultaneously
- Install WLS-SPI with OVO already installed

Installing WLS-SPI and OVO

If you are installing OVO for the first time, you can install WLS-SPI at the same time:

- 1 Follow the installation instructions in the *HP OpenView Operations/Performance for Windows Installation Guide*.
- 2 When the Product Selection window displays, be sure to select the checkbox next to BEA WebLogic listed under the Smart Plug-ins section.
- 3 Complete the installation as described in the *HP OpenView Operations/Performance for Windows Installation Guide*.

Installing WLS-SPI with OVO Installed

If OVO is already installed, it is not necessary to stop your OVO sessions before beginning the WLS-SPI installation. To install WLS-SPI, do the following:

- 1 Insert the *hp OpenView OV Operations 7.20 for Windows start-up* CD into the CD drive of the management server. The HP OpenView Operations InstallShield Wizard starts.
- 2 From the first screen, select **Next**.
- 3 In the Program Maintenance window, select **Install products**.
- 4 In the Product Selection window, select the check box next to **BEA WebLogic** and select **Next**.
- 5 Complete the installation by following the instructions in the windows that display. Refer to the *HP OpenView Operations/Performance for Windows Installation Guide* for more information.

Removing WLS-SPI

The complete removal of the WLS-SPI deletes all WLS-SPI program components as well as the WLS-SPI policies.

Complete the tasks in the order listed

- Task 1: Uninstall All WLS-SPI Policies from the Managed Nodes
- Task 2: Remove WLS-SPI Node Groups on the Management Server
- Task 3: Remove the WLS-SPI Software from the Management Server
- Task 4: Remove WLS-SPI Policy Group from the Management Server


Task 1: Uninstall All WLS-SPI Policies from the Managed Nodes

If you have customized policies (copies of WLS-SPI default policies) residing in other OVO policy groups, you should remove them as well.

- 1 In the console tree, select **Policy management** → **Policy groups**.
- 2 Right-click on **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 Select **OK**.
- 5 Verify the policies are uninstalled. Check the status of the job in **Deployment jobs** under **Policy groups**. All WLS-SPI policies must be uninstalled before you start the next task.

Task 2: Remove WLS-SPI Node Groups on the Management Server

If you ran the WLSSPI Create Node Groups tool, the SPI for WebLogic Server node group was created and must be removed:

- 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  on the Configuration toolbar to open the editor.
- 3 In the Nodes list, either select the name of the node group you want to delete and press the **Delete** key or right-click on the node group and select **Delete**.
- 4 Click **Yes** to continue the delete operation.

Task 3: Remove the WLS-SPI Software from the Management Server

- 1 Insert the *hp OpenView OV Operations 7.20 for Windows start-up* CD into the CD drive of the management server. The HP OpenView Operations InstallShield Wizard starts.
- 2 From the first screen, select **Next**.
- 3 In the Program Maintenance window, select **Remove products**.
- 4 In the Product Selection window, select the check box next to **BEA WebLogic** and select **Next**.
- 5 Complete the removal by following the instructions in the windows that display.

Task 4: Remove WLS-SPI Policy Group from the Management Server

If you have customized policies (copies of WLS-SPI default policies) residing in other OVO policy groups, you should remove them as well.

- 1 From the OVO console, select **Operations Manager** → **Policy management** → **Policy groups**.
- 2 Right-click on **SPI for WebLogic Server** and **Delete**.
- 3 Select **Yes** at the `Are you sure you want to delete this group?` prompt.

Configuring WLS-SPI

Introduction

This chapter covers configuring the Smart Plug-in for BEA WebLogic Server (WLS-SPI) for use with OpenView Operations (OVO). To successfully configure WLS-SPI, you must complete all configuration prerequisites, complete basic configuration, and complete additional configuration based on your environment.

Configuration Prerequisites

Complete the following tasks before configuring WLS-SPI:

- [Task 1: Add OVO Managed Nodes](#)
- [Task 2: Verify the Application Server Status](#)
- [Task 3: Collect Application Server Information](#)

Task 1: Add OVO Managed Nodes

For each WebLogic Administration Server and WebLogic managed server you want to manage from OVO, make sure each node on which the WebLogic servers are running is configured in OVO as a managed node.

To add a UNIX managed node, do the following:

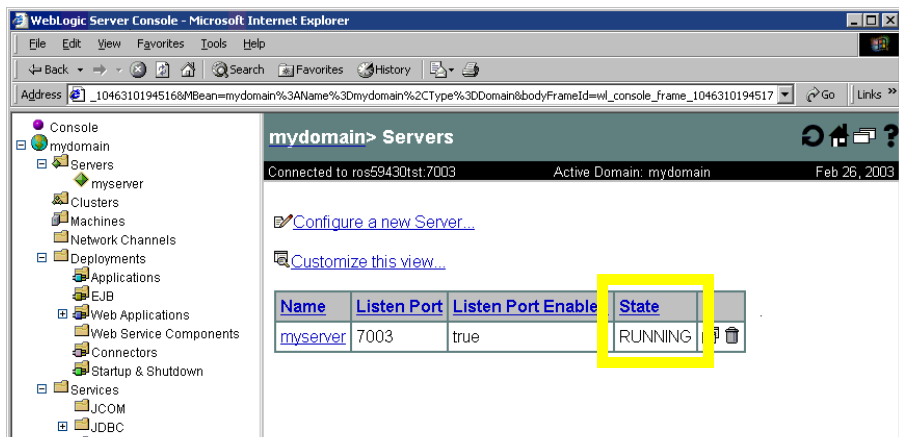
- 1 Install the OVO agent on the node. Refer to the OVO online help topic “Agent Installation on UNIX computers” for more information.
- 2 Specify each WebLogic Server node on UNIX to be managed. Refer to the OVO online help topic “Configure Managed Nodes” for more information.

To add a Windows managed node, do the following:

- 1 Specify each WebLogic Server node on Windows to be managed. Refer to the OVO online help topic “Configure Managed Nodes” for more information (the OVO agent is automatically installed when you complete this step).

Task 2: Verify the Application Server Status

Verify that your application servers are running by checking a server’s status from the WebLogic administration console. The WLS-SPI discovery policies only find application servers that are running.



Task 3: Collect Application Server Information

Basic information about your environment is needed to configure WLS-SPI:

- WebLogic login and password

- OVO managed node name(s) on which the WebLogic Administration Server(s) are running

WebLogic Login and Password

Collect the WebLogic login and password for each WebLogic Administration Server. Or, if you do not wish to use the existing login and password, create them. The login and password are needed by the WLS-SPI discovery process to gather basic configuration information and by the WLS-SPI data collector to collect metrics.

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

WebLogic Server Version 5.1 and 6.x

If you are running WebLogic Server version 5.1 or 6.x, you can use the default administration login “system.”

If you want to configure a different login, do the following:

- 1 In the WebLogic Server administration console, create the login and password that you plan to use for the WLS-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, you can use the administration login that was configured when you installed the WebLogic Server.

If you want to configure a different login, do the following:

- 1 In the WebLogic Server administration console, create a login.
- 2 Assign that login to the Administrators group.

For more information about creating a login and assigning a login to a group, refer to the “Users and Groups” section of the *Securing WebLogic Resources* manual (http://e-docs.bea.com/wls/docs70/secwres/usrs_grps.html#1150651).

OVO Managed Node Name(s)

Collect the names of the OVO managed node(s) on which the WebLogic Administration Server(s) are running. You must select these managed node(s) when you configure WLS-SPI.



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

Basic WLS-SPI Configuration

To complete basic WLS-SPI configuration, complete the following tasks:

- [Task 1: Run WLSSPI Discover](#)
- [Task 2: Verify the Discovery Process](#)
- [Task 3: Set Additional Properties](#)

Task 1: Run WLSSPI Discover

WLSSPI Discover sets basic configuration properties needed for discovery and deploys the WLS-SPI discovery policies.

To run WLSSPI Discover, do the following:

- 1 From the OVO console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **WLSSPI - SPI Admin**.
- 2 Double-click on **WLSSPI Discover**.
- 3 Select the managed nodes on which the WebLogic Administration Server(s) are running.

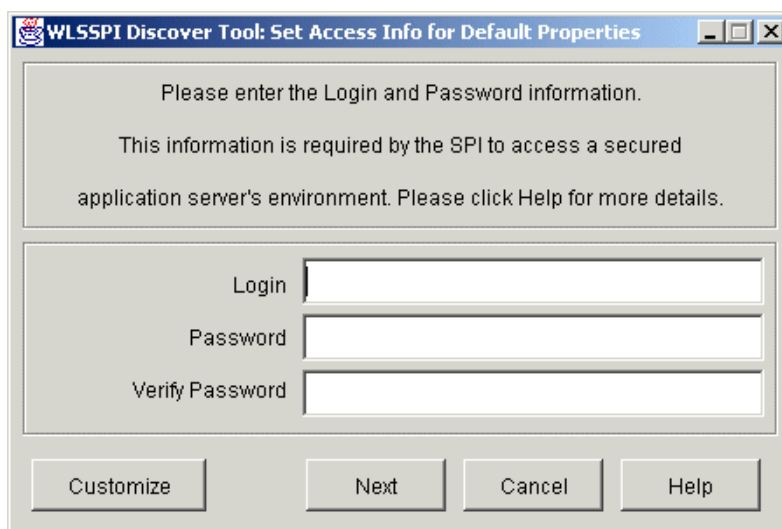
- 4 Click **Launch**.
- 5 The “Console Status” window displays. Wait a few seconds for the “Introduction” window to display. This window contains brief information about the WLSSPI Discover tool.

Select **Next**.

- 6 A second “Introduction” window displays. This window displays information on which properties may be required in order for the discovery process to work.

Read this information and select **Next**.

- 7 If you have not set the WLS-SPI `LOGIN` and `PASSWORD` properties, the “Set Access Info for Default Properties” window displays.



The screenshot shows a dialog box titled "WLSSPI Discover Tool: Set Access Info for Default Properties". The dialog contains the following text: "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".

If you have already set the `LOGIN` and `PASSWORD` properties, the configuration editor displays. Go to step 8.

Set the `LOGIN` and `PASSWORD` properties to the WebLogic login and password collected in "[Task 3: Collect Application Server Information](#)" on page 18. You must set the `LOGIN` and `PASSWORD` even if you are using the default login/password (WebLogic version 5.1 and 6.x) or the login/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). That is, if no `NODE` level or server-specific `LOGIN` and `PASSWORD` properties are set, this WebLogic login and password are used by WLS-SPI to access all WebLogic Administration Servers. For more information about the configuration structure, refer to the online help topic “The configuration.”

If the WebLogic Administration Server login and password are the same for all instances of WebLogic on all OVO managed nodes, do the following:

- a Set the `LOGIN` and `PASSWORD` in the “Set Access Info for Default Properties” window.
- b Select **Next**.
- c Go to step 10.

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 WLS-SPI configuration by setting the `LOGIN` and `PASSWORD` properties at the `NODE` level (for more information about the configuration structure, refer to the online help topic “Configuration editor operation.”):

- a Set `LOGIN` and `PASSWORD` to the most commonly used WebLogic login and password in the “Set Access Info for Default Properties” window.
- b Select **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 WLS-SPI configuration by setting the `LOGIN`, `PASSWORD`, `NAME`, and `PORT` properties at the server-specific level (for more information about the configuration structure, refer to the online help topic “The configuration.”):

- a Set `LOGIN` and `PASSWORD` to the most commonly used WebLogic login and password in the “Set Access Info for Default Properties” window.
 - b Select **Customize** to start the configuration editor and set the `LOGIN`, `PASSWORD`, `NAME`, and `PORT` properties at the server-specific level.
- 8 From the configuration editor, set the properties. Refer to the online help for more information about using the configuration editor.

- 9 Select **Next** to save any changes and exit the editor.
- 10 The “Confirm Operation” window displays. Select **OK**. The discovery policies are deployed to the selected managed nodes.



If you select **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 start the WLSSPI Discover tool, select those managed nodes, select **Next** from the configuration editor, and then select **OK**.

- 11 Scan the “Console Status” window for any error messages. If none display, click on **Close**.

If the window displays an error message, refer to "[The WLSSPI Discovery Policies](#)" on page 92 in the [Troubleshooting WLS-SPI](#) chapter to diagnose the problem.

Task 2: Verify the Discovery Process

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

- 1 Verify that the following message appears in the message browser of the managed node:

```
Updating WLS-SPI configuration in OVO server for <node>
```

and 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 OVO 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 display for all managed nodes.

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

If these messages are not present, go to "[Task 3: Set Additional Properties](#)" on page 24.

- 2 View the service map and verify that the WebLogic and WebLogic Server instances are correctly represented.
- 3 After the discovery process has completed, the appropriate WLSSPI group policies are deployed on the managed node(s). An automatic procedure to set up a managed node for WLSSPI operations starts about 10 minutes after the policies are deployed. Wait 10 minutes and run the **Verify** tool to verify the version of the policies installed on a managed node:
 - a From the OVO console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **WLSSPI - SPI Admin**.
 - b Double-click on **WLSSPI Verify**.
 - c Select the node(s) to verify (select all managed nodes running WebLogic Administration Servers and WebLogic managed servers).
 - d Click **Launch**.
 - e The WLS-SPI version is displayed. The version should be B.02.05 or higher.
 - f Click **Close**.
- 4 If verification is successful, go to "[Additional WLS-SPI Configuration](#)" on page 25. Otherwise, go to [Task 3: Set Additional Properties](#).

Task 3: Set Additional Properties

LOGIN and PASSWORD are the basic properties needed by the discover process. However, depending on your environment, additional configuration information may be needed by the discover process.

Determine if you need to set the following properties (refer to the online help for more information):

- ADDRESS - Required if the WebLogic Server is configured to a virtual IP address. ADDRESS is the domain name or IP address where the WebLogic Server is listening.
- ADMIN_PORTS - Required if the domain configuration file (config.xml) is not located in the default directory:
 - `<WebLogic_Install_Dir>/config/<WebLogic_Domain>/`
(WebLogic 6.x)

- `<BEA_Home_Dir>/user_projects/<WebLogic_Domain>/`
(WebLogic 7 or 8)

where `<WebLogic_Install_Dir>` is the directory where the WebLogic Server is installed, `<BEA_Home_Dir>` is the directory that contains the `registry.xml` file, and `<WebLogic_Domain>` is the WebLogic domain name.

`ADMIN_PORTS` is the port number(s) of WebLogic Admin server(s).

- `HOME_LIST` - Required if the BEA `registry.xml` file is not accurate or cannot be found (occurs if you did not use BEA's installation scripts to install the WebLogic Server software and Service Packs). `HOME_LIST` is a list of directories where the WebLogic Server is installed.
- `JAVA_HOME` - Required if there is more than one directory where Java is installed. `JAVA_HOME` is the default directory where Java is installed.

To set one or more of these properties, do the following:

- 1 Repeat "[Task 1: Run WLSSPI Discover](#)" on page 20. When you get to step 8, set one or more of the properties listed above.
- 2 Repeat "[Task 2: Verify the Discovery Process](#)" on page 23.
- 3 If verification is successful, go to "[Additional WLS-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.

Additional WLS-SPI Configuration

Once you have successfully completed basic WLS-SPI configuration, you must finish WLS-SPI configuration by setting additional configuration properties (these properties are not automatically discovered by the Discovery policies) and/or installing and configuring additional components. Setting some of these properties and configuring additional components depends on your environment.

Refer to the online help for a complete definition of the properties.

To run the WLSSPI Start and WLSSPI Stop tools from the OVO console, set the `START_CMD`, `STOP_CMD`, and `USER` properties.

If you are configuring user-defined metrics, set the `UDM_DEFINITIONS_FILE` property. Refer to ["User Defined Metrics"](#) on page 69 for additional configuration information.

If HP OpenView Performance Manager is installed (must be purchased separately) and you want to view graphs, set the `GRAPH_SERVER` property. Refer to ["Integrating WLS-SPI with HP OpenView Performance Manager"](#) on page 62 for additional installation and configuration information.

If HP OpenView Reporter is installed (must be purchased separately and is *not* the version of Reporter that is included with OVO), refer to ["Integrating WLS-SPI with Reporter to Generate Reports"](#) on page 58 for installation and configuration information.

If WebLogic clustering is implemented on a managed node (WebLogic 5 or 5.1 only), set the `CLUSTER` property.

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), set the `LAUNCH_DIR` property.

To update the configuration, do the following:

- 1 From the OVO console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **WLSSPI - SPI Admin**.
- 2 Double-click on **WLSSPI Configure**.
- 3 Select the managed node(s) to configure.
- 4 Click **Launch**.
- 5 Click **Next** in the "Introduction" window.
- 6 Set the properties. Refer to the online help for more information about using the configuration editor.
- 7 Select **Save**.
- 8 Select **Next**.
- 9 Select **OK**.

Using and Customizing WLS-SPI

Introduction

Like all OpenView plug-ins, the Smart Plug-in for BEA WebLogic Server (WLS-SPI) is easy to use. You should have installed the software, configured the WLS-SPI connections, and deployed the policies. In the OVO message browser, some messages may be appearing regarding the performance of WebLogic Server.

As you become familiar with the WLS-SPI, you can determine which policies are most useful to you and which you might want to change. This chapter assists you by providing further detail on the policies and how to make those changes.

In general the following topics are covered:

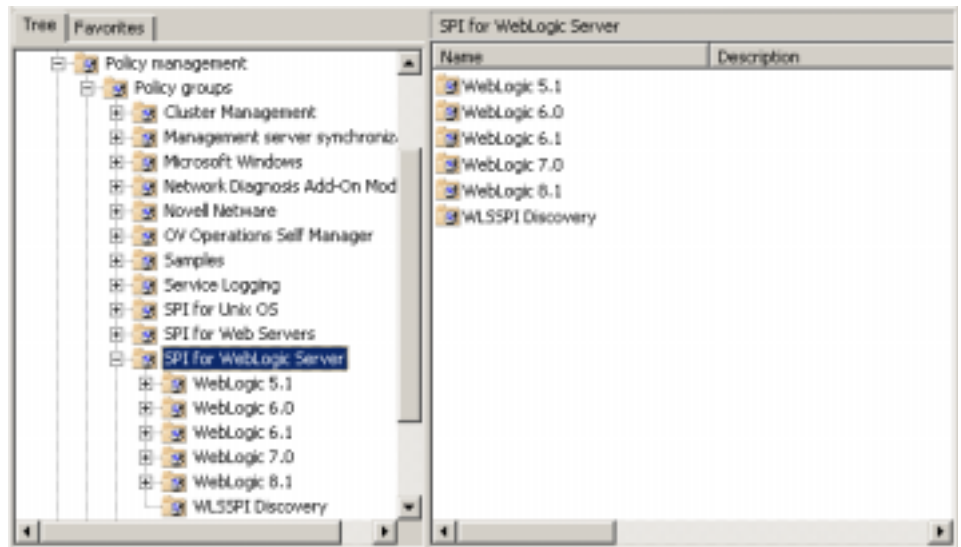
- Using the WLS-SPI Policies
- Basic Policy Customizations
- Advanced Policy Customizations
- Reinstalling WLS-SPI Policies
- Using Policies/Tools to Generate Reports
- Monitoring WebLogic on Unsupported Platforms

Using WLS-SPI Policies

You can customize WLS-SPI policies, although they work without any modifications. To help you in understanding how you might customize the WLS-SPI policies, the following sections cover the OpenView Operations policies in general and WLS-SPI policy groups in particular.

WLS-SPI Policy Groups

WLS-SPI policy groups provide a means of organizing various OVO policies. There are four primary WLS-SPI policy groups (6.0, 6.1, 7.0, and 8.1).



These groups contain metric and logfile policies as follows:

- **Metric (monitor) policies:** generate messages according to threshold settings monitoring WebLogic availability and performance metrics grouped according to collection intervals. The Metrics group contains a list of policies that fall into two categories according to their function. *Metric policies* pertain to individual metrics. *Collector policies* pertain to all metrics scheduled to be collected in the specified collection interval.
- **Logfile policies:** generate messages according to logfile and error text detected in both the WebLogic Server logfiles and in the WLS-SPI logfiles.

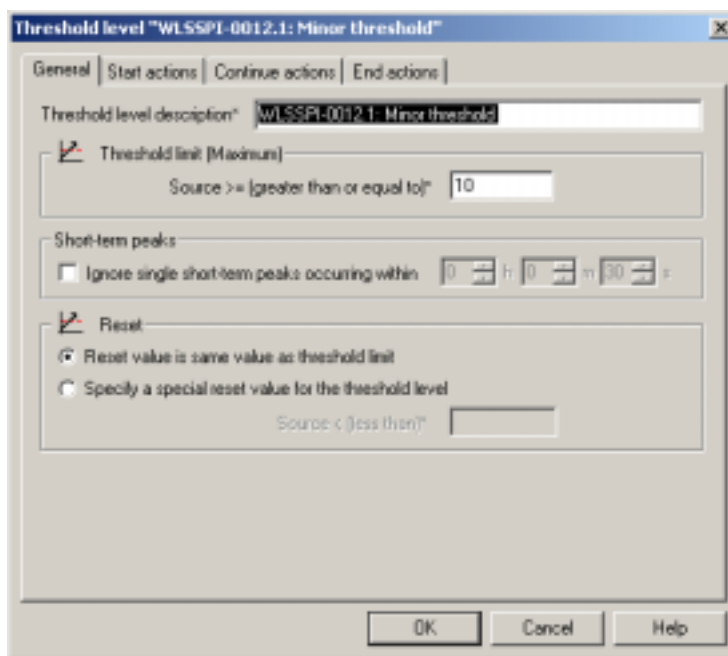
The **WLSSPI Discovery** policy automatically updates the WebLogic configuration information in the service map. Uncheck the `AUTO_DISCOVER` check box in the configuration editor if you do not want the discovery policy to automatically overwrite this configuration information. Refer to the online help for information on how to use the configuration editor.

OVO Policy Types and WLS-SPI

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 on 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 figure below, the default threshold is set at 10.

Figure 1 Threshold Level Window

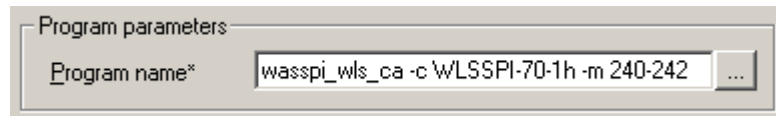


Collector policies define all metrics for the WebLogic Server application that are scheduled for collection at the specified interval. Though still identified as a "metric policy" in the OVO lists of policies, you will notice that these policies

have names different from the individual metric policies. Within the name of each collector policy is its collection interval (for example, `WLSspi-70-1h`). 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`.

The figure below shows the `Program name` text box which contains the collector/analyzer command (`wasspi_wls_ca`), followed by the collection parameter and collection name (`- c WLSspi-70-1h`), followed by the metric parameter and each metric (`- m 240-242`)

Figure 2 Program Name Text Box.



Basic Policy Customizations

After you begin using the WLS-SPI, you may decide that specific WLS-SPI policies need some modification. Descriptions contained in the previous section of metric policies (rules for interpreting metric data, for example, thresholds,) and collector policies (rules for the scheduled metric collection) show you where to go to make the various changes. For example, to change a threshold, you would open a metric policy. To schedule or delete a metric from data collection, you open the collector policy. Those kinds of basic customizations are covered in this section.



In most cases, it is advisable to make copies of the original policies so that the default policies remain intact.

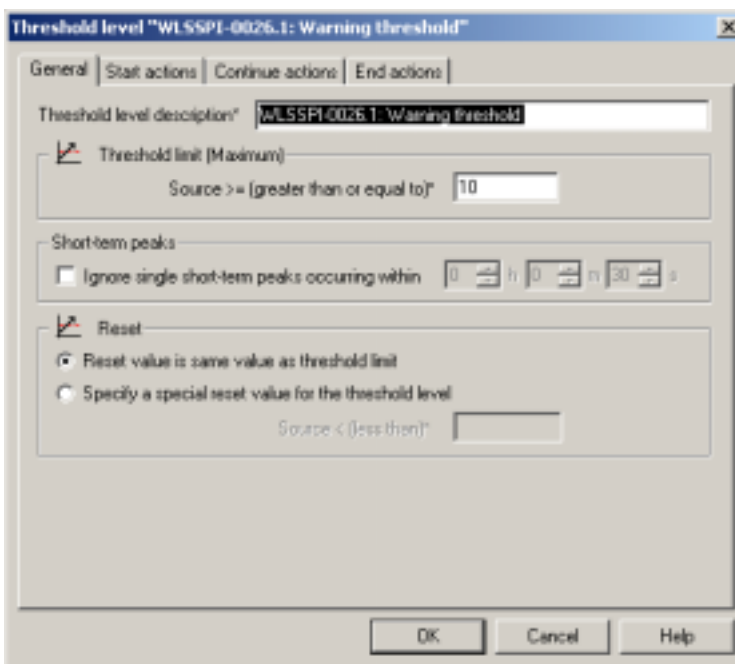
Modifying Metric Policies

Many metric attributes can be modified for all monitored instances of WebLogic Server. Attributes not mentioned here are defined in the online help.

Threshold Level and Actions

To modify the threshold level and actions of a policy, do the following

- 1 From the OVO console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WebLogic 7.0** → **Metrics**.
- 2 Double-click on a policy.
- 3 Select the **Threshold levels** tab.
- 4 From the **Level summary** pane, click on **Threshold level**. The **Threshold level** window displays.



In the figure above, 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. A value of more than 10 would start to impact the server response time the client experiences, generating a Warning message.

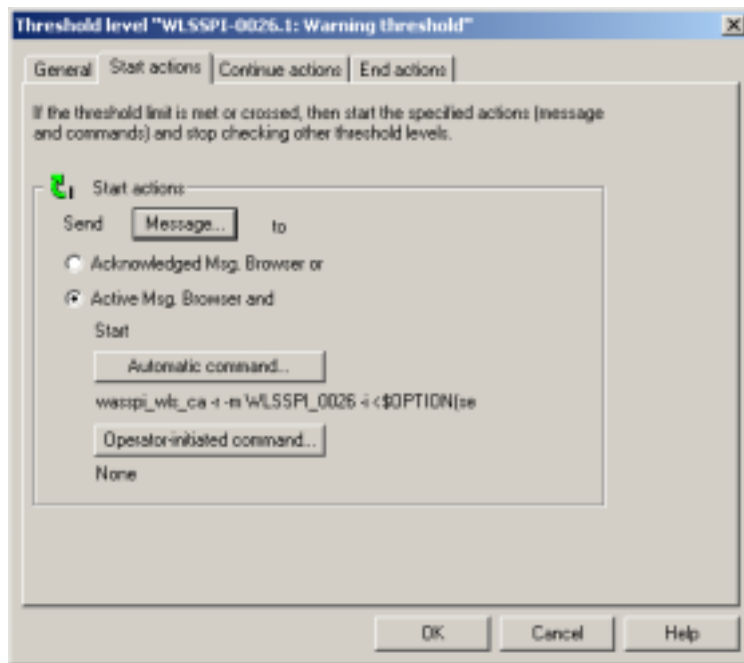
The following attributes can be modified:

- **Threshold limit.** The value that triggers a message if it is met or crossed.

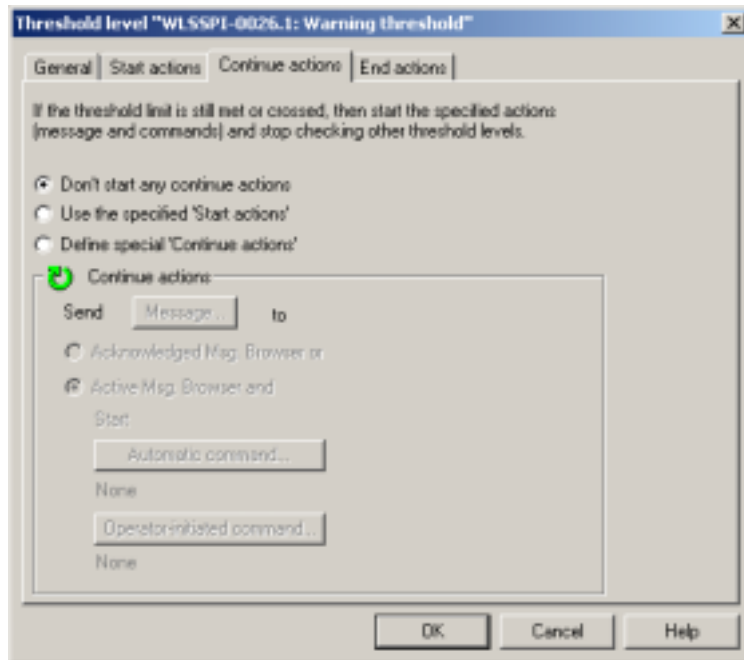
- **Short-term peaks.** A minimum time period over which the monitored value must exceed the threshold before generating a message. For a message to be sent, the value must be greater than the threshold each time the value is measured during a duration that you select. If the duration is set to 0 or the box is left empty, an alarm is generated as soon as OVO 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.

Click on one of the actions tabs to set the following:

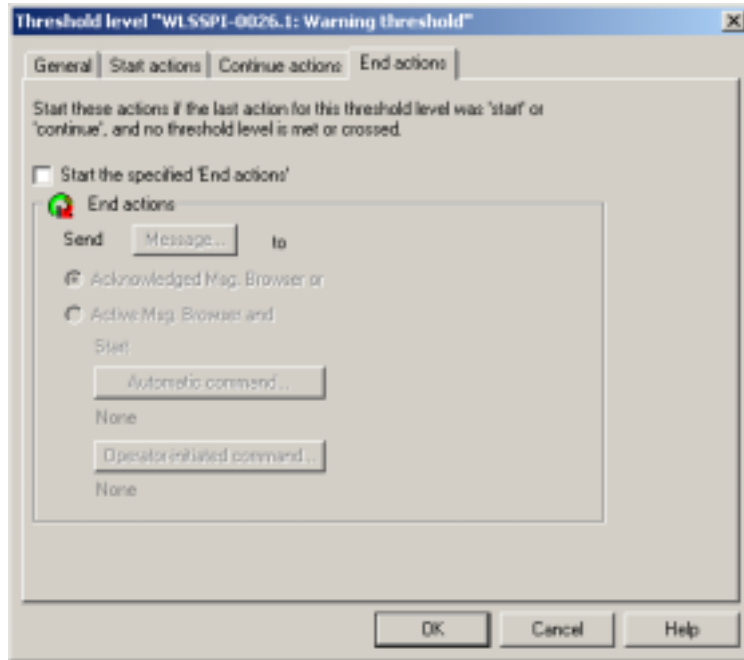
- **Start actions.** Actions carried out the first time that the threshold is crossed.



- **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.



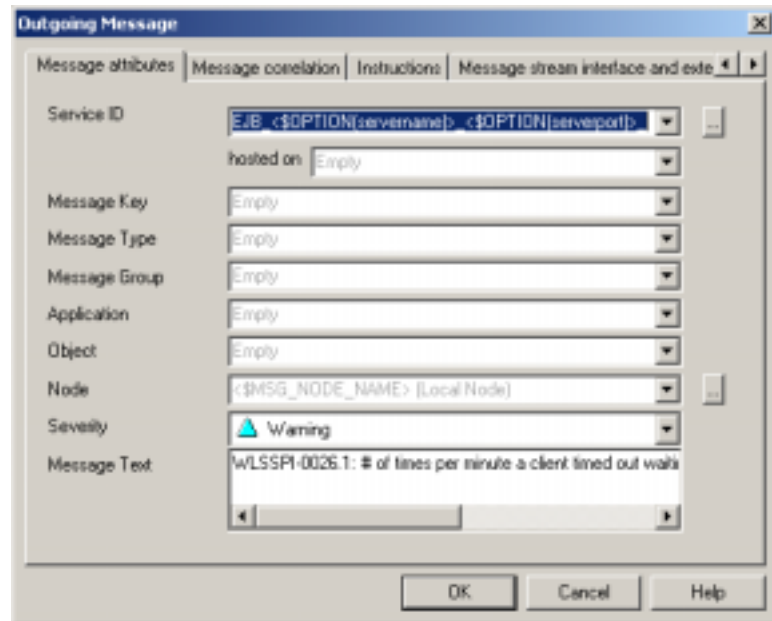
In each of the actions tabs, you can set the type of actions to perform. The WLS-SPI provides the ability to generate Performance Manager graphs or reports, or to add custom programs. The reports or graphs are accessible to the user from:

- **Automatic command.** A command run when the rule is matched. The automatic command delivered with the WLS-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. This command can be started by the operator from the message browser. The operator-initiated command delivered with the WLS-SPI allows the operator to press 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, do the following

- 1 Double-click on a policy.
- 2 Select the **Threshold levels** tab.
- 3 From the Level summary pane, click on **Message**. The **Outgoing Message** window displays.



The following attributes can be modified:

- **Severity.** Indicates to the operator the importance of the event which triggers this message.
- **Message Text.** Be careful not to modify any of the parameters—surrounded by <> brackets, beginning with \$—in a message.

Advanced Policy Customizations

The policy changes suggested here range from making copies of default policy groups in order to customize a few settings, to deleting whole groups of metrics within a policy's command line. This section is considered *advanced* because all changes described here, whether simple or complex, require some advanced knowledge of the WLS-SPI metrics.

Choosing Metrics To Monitor

As a start, determine which metrics you want to change and what policies within the group you want to use. Then proceed as follows:

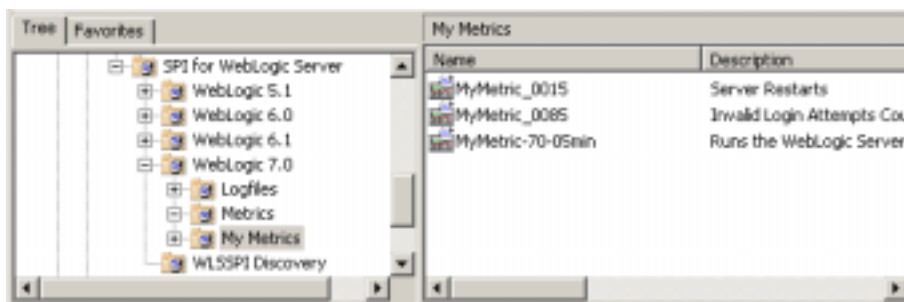
- 1 Create a new policy group:
 - a Right-click on the policy group you want to copy and select **Copy**.

For example, right-click on the Metrics policy group under WebLogic 7.0 and select **Copy**.
 - b Right-click on the group under which this policy group is located and select **Paste**.

For example, right-click on WebLogic 7.0 and select **Paste**.
 - c Rename the new group.

For example, right-click on `Copy of Metrics` and select **Rename**. Type in a new name.
- 2 Rename the original policies within the new policy group:
 - a Right-click on the policy and select **All Tasks** → **Edit**.
 - b Click on **File** → **Save As**.
 - c Enter a new policy name and select **OK**.
- 3 Delete all original policies within the new policy group:
 - a Highlight the policies and hit **Delete**.
- 4 Alter the renamed policies within the new group as necessary.

Creating a new policy group allows you to keep custom policies separate from the original default policies, which you copy and place within the new group.



Using the WLS-SPI Collector/Analyzer Command with Parameters

The `wasspi_wls_ca` command is used in every collector policy, named according to its collection interval. You can view the default command line parameters within each collector policy in the **Program name** text box in OVO.

Using the WebLogic Server Command Parameters

WLS-SPI data collections are started with the `wasspi_wls_ca` command, to which you can add other parameters, as identified in the following table.

Parameter	Function	Syntax with Example
-c	(collector) Specifies collector policy name. (required)	<code>-c <collector_policy_name></code> Note: Must match the collector policy name in which it appears. Example: <code>-c WLSSPI-61-05min</code>
-m	(metric) Specifies the metric numbers or number ranges on which to collect data.	<code>-m <metric_number,metric_number_range></code> Example: <code>-m 1,3-5,9-11,15</code>

Parameter	Function	Syntax with Example
-t	(tag) Allows you to create a new policy group by adding a prefix to an existing collector policy along with the metric number(s).	<pre>wasspi_wls_ca <prefix>-<collector_policy> -m <metric_number> -t <prefix>-</pre> <p>Example:</p> <pre>wasspi_wls_ca -c WLSSPI-61-05min -m 220-223 -t DEV-</pre>
-i	(include) Allows you to list specific servers to monitor. This option may not be used with <code>-e</code> option.	<pre>-i <server_name></pre> <p>Example: <code>-i server1,server3</code></p>
-e	(exclude) Allows you to exclude specific servers; may not be used with <code>-i</code> option.	<pre>-e <server_name></pre> <p>Example: <code>-e server2,server4</code></p>
-x	Allows you to specify a property/value as follows: alarm: when <code>off</code> , overrides any alarming condition as set up in the metric policy. print: when <code>on</code> , prints the metric name, instance name, and metric value to STDOUT in addition to any configured alarming or logging. log: when <code>off</code> , prevents graphing or reporting functions.	<pre>-x <property>=<property_value></pre> <pre>-x alarm=off</pre> <pre>-x print=on</pre> <pre>-x log=off</pre>

Syntax Examples

- To specify metrics to collect:

```
wasspi_wls_ca -c WLSSPI-61-05min -m 10-14,25,26
wasspi_wls_ca -c <measurement_threshold_policy_name> -m
<metric_number_range>, <metric_number>
```

- To differentiate server instances:

```
wasspi_wls_ca -c STAGE-WLSSPI-61-05min -m 245,246,260 -i
server1,server3 -t STAGE-
```

(Inserting “server1” and “server3” in the Program name text box of the collector policy results in collecting data for the specified metrics from these servers only.)

Changing the Collection Interval for All Scheduled Metrics

To change the metric collection interval, simply change the Polling Interval in the appropriate collector policy. For example, to change the collection of default metrics from 5 minutes to 10 minutes for the WebLogic 7.0 policy group WLSSPI-70-05min collector policy, follow these steps:

- 1 From the OVO console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WebLogic 7.0**.
- 2 Select **Metrics**.
- 3 Right-click on the collector policy **WLSSPI-70-05min** and select **All Tasks** → **Edit**.
- 4 Click on **File** → **Save As** and change the Name to **WLSSPI-70-10min**.
- 5 Change the **Polling Interval** from 5m to 10m.
- 6 Modify the command line **-c** parameter to reflect the new policy name (WLSSPI-70-10min) as follows:

```
wasspi_wls_ca -c WLSSPI-70-10min....
```
- 7 Deploy the new policy:
 - a Right-click on **WLSSPI-70-10min**.
 - b Select **All Tasks** → **Deploy on ...**
 - c Select the node(s) on which to deploy the policy.
 - d Click **OK**.

Changing the Collection Interval for Selected Metrics

To change the collection interval for selected metrics, copy the appropriate collector policy and rename with a name reflecting the new interval, deleting all but the metrics you are changing. Set the new interval. Edit the original policy to remove the changing metrics. For example, to change the collection interval to 10 minutes for metrics 70-81, follow these steps:



- 1 From the OVO console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WebLogic 7.0**.
- 2 Select **Metrics**.
- 3 Right-click on the collector policy **WLSSPI-70-05min** and select **All Tasks** → **Edit**.
- 4 Click on **File** → **Save As** and change the Name to **WLSSPI-70-10min**.
- 5 In the Program name text box, delete all metrics after the **-m** except **70-81**.
- 6 Change the **Polling Interval** from 5m to 10m.
- 7 Modify the command line **-c** parameter to reflect the new policy name (WLSSPI-70-10min) as follows:
`wasspi_wls_ca -c WLSSPI-70-10min . . .`
- 8 Select **Save and Close**.
- 9 Right-click on the collector policy **WLSSPI-70-05min** and select **All Tasks** → **Edit**.
- 10 In the Program name text box, delete metrics **70-81** after the **-m**.
- 11 Select **Save and Close**.
- 12 Deploy the new policies:
 - a Right-click on **WLSSPI-70-10min**.
 - b Select **All Tasks** → **Deploy on ...**
 - c Select the node(s) on which to deploy the policy.
 - d Click **OK**.
 - e Right-click on **WLSSPI-70-05min** and repeat steps b-d.

Customize the Threshold for Different Servers

Customize the threshold as needed. For example, you may want to set the threshold to 20 for `SERVER_1` for metric 0012 and leave the threshold at 10 for all other servers. To do so, copy the existing condition and modify it to serve as the exception. Follow these steps:

- 1 Double-click on the metric to open it for customization (for example, double-click on **WLSSPI-0012**).

The Measurement Threshold window displays.
- 2 Select the `Threshold levels` tab.
- 3 Press the **Specify instance filters** button.
- 4 Select the `Condition` tab.
- 5 Enter a Rule description (for example, "Rule for all servers except `SERVER_1`").
- 6 Press **OK**.
- 7 In the Measurement Threshold window, press the **Copy** button to make a copy of the rule.
- 8 Double-click on the copy of the rule. Enter a new Rule description (for example, "Rule for `SERVER_1`").
- 9 In the `Object name matches` field, enter the desired characters to use for pattern matching (in this example `SERVER_1`).
- 10 Select the `Actions` tab.
- 11 Double-click on the condition.
- 12 Change the name the condition to `WLSSPI-0012.2`.
- 13 Change the Threshold limit to 20.
- 14 Press **OK**.
- 15 Press **OK**.
- 16 Verify the order of your rules (for example, make "Rule for `SERVER_1`" the first rule).
- 17 Click the **Matching test** button to test the pattern and verify pattern matching (you must set up a match file first).

Creating Custom, Tagged Policies

Another advanced customization option is to use the tag option (`-t` on the command line), which allows the collector/analyzer to recognize customized policies that have a tag attached to the name. This option provides you with the flexibility of using more than a single set of policies to define conditions pertaining to specific installations of WebLogic Server.

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

For example, you might create a group of policies and change each policy name to include `CLIENT01` in it. A metric policy might be named `CLIENT01-WLSSPI_0012` (retaining the metric number, which must be used). The collector policy name would be named `FIRST_CLIENT-61-05min`. You could then set up another group for `SECOND_CLIENT` and change all those policies to include the `CLIENT02` in the name.

To create the new policy group:

1 Copy the original policy group:

- a Right-click on the policy group you want to copy and select **Copy**.

For example, right-click on the Metrics policy group under WebLogic 7.0 and select **Copy**.

- b Right-click on the group under which this policy group is located and select **Paste**.

For example, right-click on WebLogic 7.0 and select **Paste**.

- c Right-click on `Copy of Metrics` and select **Rename**. Rename the new group according to how you plan to identify the new metric and collector policies.

For example, rename the group to `CLIENT01Metrics`.

2 Rename the original policies within the new policy group.

The names you give the new *metric policies* in the group would contain the new name followed by the original metric number. For example, a copy of

WLSSPI-0001 could be called CLIENT01-WLSSPI_0001.

The name you give the new *collector policy* would also contain the identifying name. You would also modify the scheduled collection for the new group by inserting the `-t` property in the Program name text box. For example:

```
wasspi_wls_ca -c FIRST_CLIENT-61-10min -m 16 -t CLIENT01-
```

In this case the copied collector policy has been renamed:
FIRST_CLIENT-61-10min

- a Right-click on the policy and select **All Tasks** → **Edit**.
 - b Click on **File** → **Save As**.
 - c Enter a new policy name and select **OK**.
- 3 Delete all original policies within the new policy group:

Highlight the original policies and hit **Delete**.

Re-installing the WLS-SPI Policies

To restore the default WLS-SPI policy groups on your OVO management server, you must remove and then reinstall the WLS-SPI. Refer to "[Removing WLS-SPI](#)" on page 14 and "[Installing WLS-SPI](#)" on page 13 for more information.

Using Policies/Tools to View Annotation Reports and Graphs

Some policies have actions defined with threshold violations or error conditions that automatically cause reports to appear in the message Annotations. These reports are snapshots of data values collected from the server around the time that the alarm occurred.

Other policies have operator actions associated with them that allow you to generate a graph.



The reports discussed in this section should not be confused with those generated by OpenView Reporter, which show more consolidated, historical data generated as Web pages in management-ready presentation format.

You can access the data as follows:

- **To view the Message Properties.** Double-click on a message in the OVO message browser. Reports can be viewed from the `Annotations` tab area, showing data values on a single server.
- **To view reports.** From the OVO console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **WLSSPI - Metric Reports**. Double-click on the report. A report is generated for all WebLogic Server instances on the selected managed node.
- **To view graphs.** Double-click on a message in the OVO message browser. Graphs can be generated in the `Commands` tab area, if an operator-initiated command has been configured and data has been collected. Select **Start** to generate the graph.

The View Graphs application launches your Web browser, which displays the graphing feature available in HP OpenView Performance Manager (which must be purchased separately).

Automatic Command Reports

Many metrics generate Automatic Command Reports. These reports are generated as soon as an alarm is triggered in OVO.

How you know a report has been generated

When an Automatic Command Report is executed from OVO, the server is queried for additional data. If your message browser is set to display the SUIAON column, you will see an “S” under the “A” column (see illustration), 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	Applicatio	Obj
Warning	-	-	X	-	-	-	1/14/2003 10:06:34 AM	OpC	HP OpenView Operations	opon
Warning	-	-	-	-	-	-	1/16/2003 12:57:15 PM	WINOSPI-NO...	Norton AntiVirus	Nort
Normal	-	-	-	-	-	-	1/17/2003 9:34:23 AM	VP_SM	HP OpenView Operations	OvD
Critical	0	-	X	S	S	X	1/17/2003 9:34:27 AM	WebLogic	WebLogic	opoc
Warning	-	-	-	-	-	-	1/17/2003 9:39:24 AM	OpC	HP OpenView Operations	opon
Critical	-	-	-	-	-	-	1/23/2003 9:05:22 AM	VP_SM	HP OpenView Operations	Alert

How to view the report

To view the report, double-click on the message and select the *Annotations* tab. Column descriptions provide further clarification.

How Automatically Generated and Manually Generated Reports Differ

Automatic Command reports show the state of the system moments after the alarm occurred. You can manually generate a report that shows the current state of the system by navigating down the console tree to the individual SPI for WebLogic reports and double-clicking on the report.

Reports

Reports run for all WebLogic Server instances configured on the managed node, in contrast to Automatic Command Reports which are generated for a single WebLogic Server instance. The reports generated reflect the current state of WebLogic Server on the managed node.

To generate a report in OVO, simply navigate down the console tree to the individual SPI for WebLogic reports, double-click on the report, and select a managed node.

Use these tools to manually generate a report similar to an Automatic Command report. These reports differ by showing data on ALL WebLogic Server instances on the managed node. The Automatic Command report shows data on the single WebLogic Server instance with the exceeded threshold

Sample Report

The following example shows the format of a WLS-SPI report. This report is automatically generated (automatic command report).

Figure 3 Report on Metric B-011

```

Report for Application Server_01
Oct 16, 2001 3:22:20 PM
Metric B011_ExQThrdUtilPet

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

```

Monitoring WebLogic on Unsupported Platforms

The WLS-SPI supports monitoring WebLogic Server-installed systems running on HP-UX, Solaris, and Windows NT/2000. However, it is possible to configure the WLS-SPI to monitor a WebLogic Server installed on systems running on unsupported platforms—systems we refer to as “remote systems.”

The intent of this section is to help you determine if your environment is conducive to setting up remote monitoring. If you determine that your environment meets the criteria described below, and you have some expertise in using the WLS-SPI, this section offers an example to get you started.

Requirements for Monitoring Remote Nodes (running on Platforms not supported by WLS-SPI)

For a WebLogic Server installation on a system running on a platform other than Solaris, HP-UX, or Windows NT/2000, you can use WLS-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 0 pricing).
- The WLS-SPI runs on at least one managed node on a supported platform: HP-UX, Solaris, Windows NT or Windows 2000.
- (Optional, for logfile monitoring) The remote system runs on a platform supported by the OVO agent software.

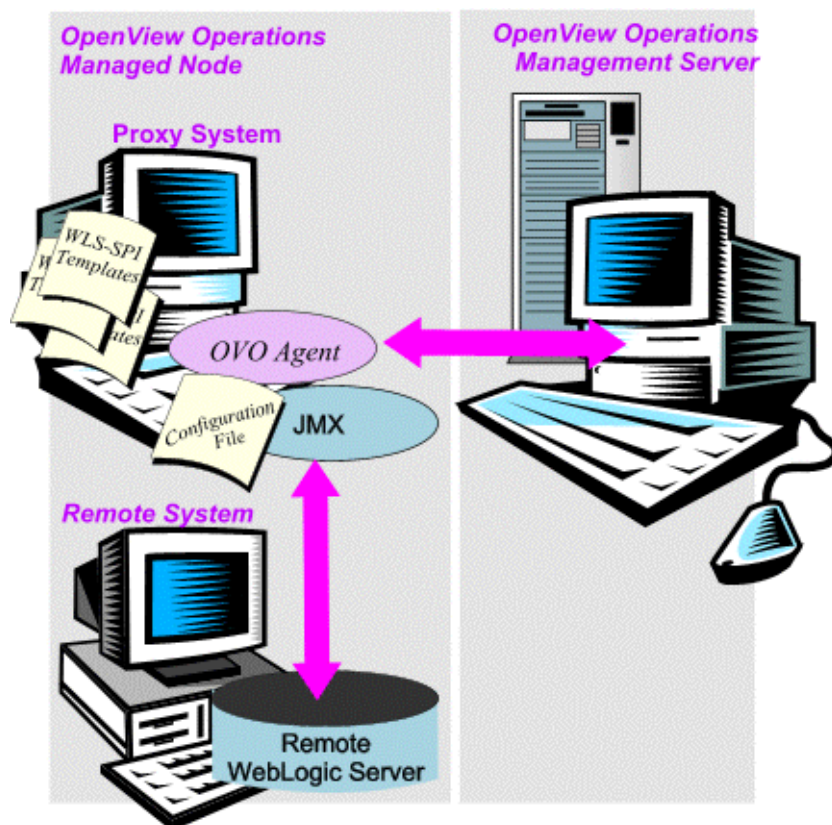
Overview

The following section provides an overview of remote monitoring and shows how it is implemented. Also included are details on how to set up the WLS-SPI to access WebLogic Server metrics and logfiles on unsupported platforms by using both the WLS-SPI and OVO agent software.

Remote Monitoring: How It Works

In a standard configuration, WLS-SPI programs/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/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. (Please refer to the example on page 51, showing how the remote entry appears (with system IP address).

Policy deployment requirement: Policies for the correct WebLogic Server version must be deployed on the local node. If you need a separate policy group (for example 6.0 or 6.1) to cover a different WebLogic Server version, you can copy and rename the existing policies and specify the WebLogic Server name on the command line using the `-i` or `-e` options. Refer to a previous section in this chapter "Using WLS-SPI Policies, page 28" for details on using these command line parameters.

OVO agent deployment requirement (optional logfile monitoring): To access remote WebLogic logfiles, the OVO agent software must be installed on the remote system. Using standard OVO processes, you can modify the standard logfile policies included with the WLS-SPI to specify the correct logfile names, then deploy them to the remote system.



Monitoring remote systems using logfile versioning is not supported.

Configure Remote System Monitoring

You can monitor WebLogic servers remotely on systems other than those running on HP-UX, Solaris, or Windows NT/2000 platforms by completing the following tasks.

Task 1: Configure the Remote WebLogic server

Using the **WLSSPI Configure** tool in the WLSSPI - 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 Start the **WLSSPI Configure** tool.
- 2 Choose a WebLogic managed node from which to monitor the remote WebLogic Server.
- 3 In the configuration, include an entry for each remote WebLogic server:
ADDRESS=<DNS server name or IP address>.

The example configuration below shows how local and remote WebLogic servers are configured in the same file. Notice, however, that 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/wlserver6.0spi
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
```

There are two WebLogic servers configured in the example configuration. `SERVER1` is the local server, running on an HP-UX managed node. `SERVER2` is running on an OVO-managed node, that is a Linux system (a platform unsupported by WLS-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 OpenView Performance Agent (MeasureWare Agent)

Since the OpenView Performance Agent collection occurs on the managed node, not the remote system, if you use PerfView and would like to graph the remote system data, you must ensure that MeasureWare integration is enabled on the (local) managed node.

Task 3: Assign Local Node to WLS-SPI node group

Assign the local managed node to the appropriate node group. For example, you would assign the local node to the WLS 6.1 node group if the local and remote managed nodes are both WebLogic 6.1 servers. Make sure you are deploying the configuration on the same WebLogic Server version node as the remote system.

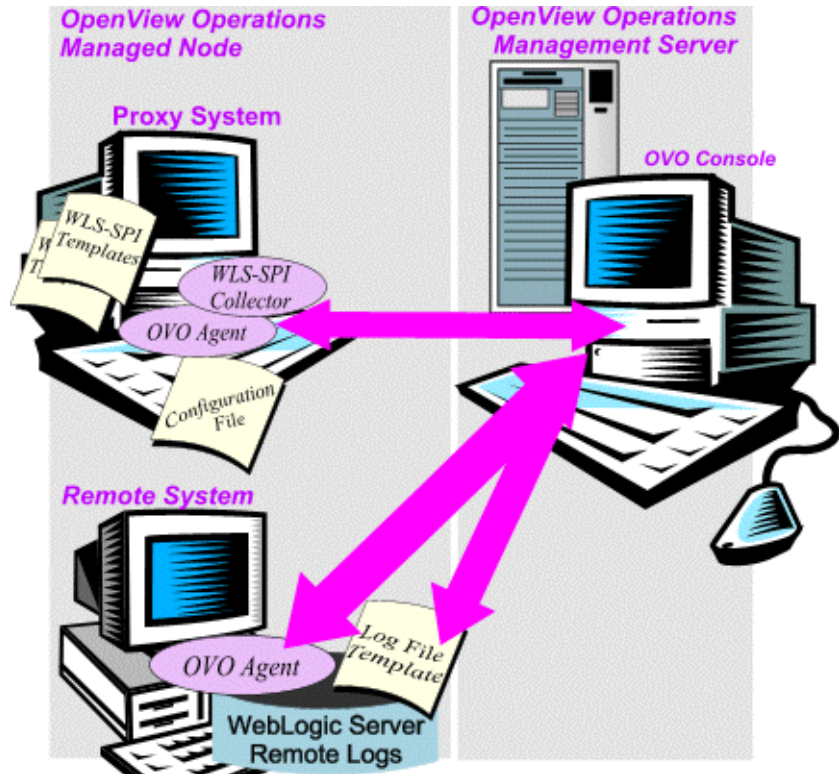
Configuring Remote Monitoring for Logfiles (Optional)

Monitoring remote system logfiles is supported if the following is true: (1) The remote system has an OVO agent running on it and (2) the system does not reversion logfiles when they roll. To set up logfile monitoring, at the OVO console, copy the WLS-SPI logfile policy and then configure, assign, and deploy the copied logfile policy to the remote system.

Configure the logfile policy for remote logfiles

- 1 Open a copy of the WebLogic Log Policy located under the WebLogic version Group.
 - a From the OVO console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WebLogic 7.0**.
 - b Select **Logfiles**.
 - c Double-click on the log policy
- 2 In the Logfile pathname text box, enter the location of the logfile on the remote system: `/<path>/<filename>`.
- 3 Assign and deploy the logfile policy to the remote OVO-managed node.

The Log File Policy and the OVO Agent, both present on the remote system, make WebLogic Server logfile monitoring possible.



Remote Monitoring Limitations

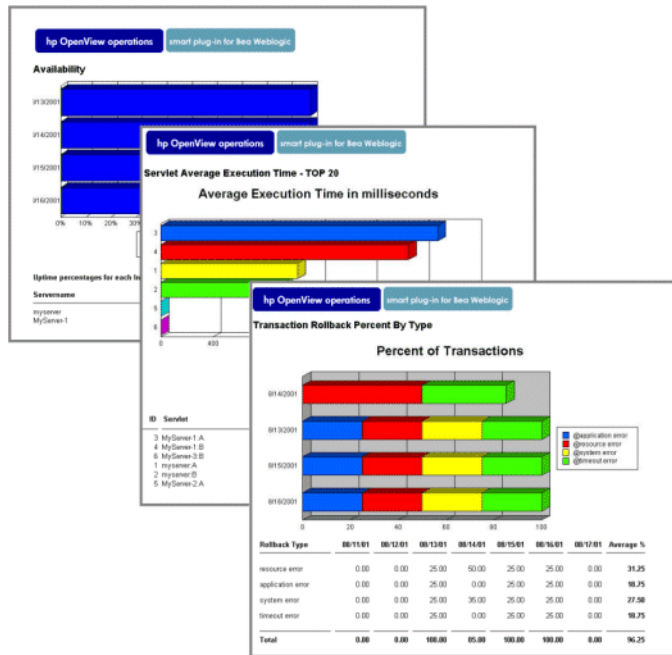
- The WLS-SPI and the OVO agent do not support access to logfiles that are re-versioned each time the logs are rolled.
- When no OVO agent is present on the remote system, monitoring of WebLogic logfiles on the remote system cannot occur.
- WLS-SPI tools cannot be executed on remote systems.
- WLS-SPI does not support application servers with the same name.

Using HP OpenView Reporting and Graphing Features with WLS-SPI

Introduction

HP OpenView Reporter, which must be purchased separately (this is *not* the version of Reporter that is included with HP OpenView Operations for Windows), integrates fully with the Smart Plug-in for BEA WebLogic Server (WLS-SPI). Reporter produces management-ready, Web page reports, showing historical and trending information.

The WLS-SPI, working in conjunction with Reporter, produces a variety of reports, showing consolidated information on WebLogic Server.



Another OpenView product, Performance Manager, provides graphing capability. With this added capability you can use the WLS-SPI Graph tools to generate graphs of current conditions. In addition, Performance Manager also works to generate graphs associated with Operator-initiated actions (wherever included in a WLS-SPI policy).

This section provides information on how to fully integrate WLS-SPI with HP OpenView Reporter and Performance Manager. After you complete the instructions in this chapter, every night Reporter generates reports that show the performance and availability of WebLogic Server on configured managed nodes. You can also take advantage of the built-in graphing capabilities of the WLS-SPI, coupled with Performance Manager.

Using the OpenView Performance Agent

The OpenView performance subagent (CODA) is automatically deployed on all managed nodes. WLS-SPI relies on this default performance subagent to collect and store performance data used with OVO graphing and reporting. These reporting and graphing features only work with CODA.

However, you may want to use the OpenView Performance Agent (OVPA or MeasureWare Agent) which is supported by HP OpenView Reporter and HP OpenView Performance Manager (these products must be purchased separately).

You can configure the WLS-SPI data collector to use OVPA to collect and store performance data, but you cannot use the graphing and reporting features in OVO with this data.

To configure the WLS-SPI data collector to use OVPA, do the following:

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

Operating System	File Location
UNIX	<code>/var/opt/OV/conf/dsi2ddf/</code>
Windows	<code>\Program Files\HP OpenView\Installed Packages\{790C06B4 ...}\conf\dsi2ddf\</code>

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

- 2 Edit the `nocoda.opt` file to contain the following single line:
ALL
- 3 Save the file.

Integrating WLS-SPI with Reporter to Generate Reports

Prerequisite: Configuration of the WLS-SPI, which includes software deployment, server connection configuration, and assignment/deployment of policies to targeted nodes.

- 1 Install the WLS-SPI report package on the Windows system running Reporter:
 - a Insert the *hp OpenView OV Operations 7.20 for Windows start-up* or *hp OpenView OV Operations 7.20 for Windows smart plug-ins* CD into the CD drive of the Windows system running Reporter.
 - b On the *start-up* CD, go to the /Disk1/WebLogic SPI Reporter Package directory.

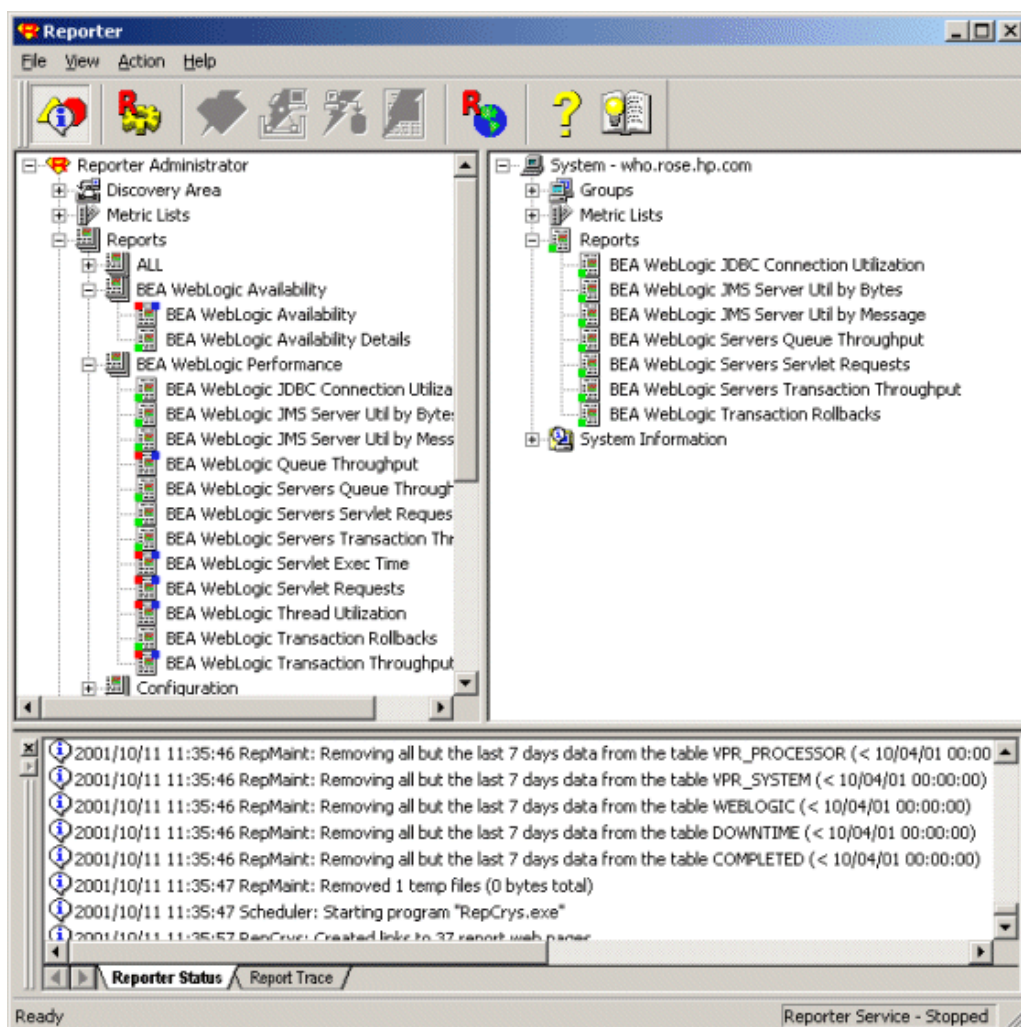
On the *smart plug-ins* CD, go to the /Disk2/WebLogic SPI Reporter Package directory.
 - c Double-click on **WLSSPI-Reporter.msi**
 - d Complete the installation by following the instructions in the windows that display.
- 2 Check the Reporter status pane (see the illustration that follows) to note changes to the Reporter configuration.



For Windows 2000 managed nodes, during the installation an error message may appear that indicates the installer has detected an older version of the installer on your system. You can safely ignore the message and continue.

For NT 4.0 managed nodes, during the installation you may get the error: 1604: This setup does not contain the Windows Installer engine [INSTMSIW.EXE] required to run the installation on this operating system. In this case you must install the Microsoft Windows Installer from the Microsoft Web site.

The status pane (at the bottom of the Reporter window) shows you information on the programs running and any errors occurring. You can check the status pane to see that Reporter has been updated with the WLS-SPI reports.



You can find instructions in the Reporter Help for assigning WLS-SPI reports to the targeted nodes. To access Help, select **Reports** or **Discovered Systems** in the left panel of the Reporter main window and right-click it. Select **Report Help** or **Discovered Systems Help** from the submenu that appears. See the topic “To assign a report definition to a Discovered Systems Group.”

- 3 Add group and single system reports by assigning reports as desired. (See the Reporter Help and the online *Concepts Guide* for complete information.)



Group and single system WLS-SPI reports require that you identify systems by their full name (for example, **abc.xyz.com** is acceptable while **abc** is not).

WLS-SPI Reports

The reports available through the integration of HP OpenView Reporter and WLS-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 the day following your installation of the WLS-SPI report package on the Reporter Windows system. (Please refer to page 58 if you have not yet completed the report package installation.)

The following tables show pre-defined reports

Table 1 Performance

Report Title	Description	WebLogic Version	Metric
DB Connection Utilization	Shows DB connection pools throughput against the connection utilization	6.x, 7.0, 8.1	260
EJB Cache Hits - Top 20	Shows stateful and entity EJB cache hits for the top 20 servers.	7.0, 8.1	238
EJB Free Pool Wait Rate - Top 20	Shows the number of times per minute that a request had to wait for an EJB to become available for the top 20 servers.	7.0, 8.1	25
EJB Pool Utilization	Shows the EJB pool utilization for the top 20 servers.	7.0, 8.1	235

Table 1 Performance

Report Title	Description	WebLogic Version	Metric
EJB System Transaction Tput. - Top 20	Shows the throughput (transactions per second) for the top 20 EJBs for each WebLogic Server.	6.1	227-230
EJB Timeout Rate - Top 20	Shows the number of times per minute that a request timed out while waiting for an EJB to become available for the top 20 servers.	7.0, 8.1	26
EJB Transaction Rollback Rate - Top 20	EJB transaction rollback rate by EJB for the top 20 servers.	6.1	36
EJB Transaction Throughput - Top 20	Shows the average EJB transaction throughput for the top 20 servers.	6.1	35
JMS Server Util by Bytes	Charts the throughput against the queue utilization by byte count for destinations on the JMS servers for each WebLogic Server.	6.x, 7.0, 8.1	252, 256
JMS Server Util by Message	Charts the throughput against queue utilization by message count on the JMS servers for each WebLogic server.	6.x, 7.0, 8.1	251, 255
Server Queue Throughput - Top 20	Shows the average throughput for the top 20 execute queues of all servers.	6.x, 7.0, 8.1	10
Server Queue Utilization	Shows the throughput against the thread utilization for each WebLogic server.	6.x, 7.0, 8.1	10,11
Server Queue Utilization - Top 20	Shows the throughput against the thread utilization for the top 20 servers.	6.x, 7.0, 8.1	10,11
Servlet Avg. Response Time - Top 20	Shows the average response time for the top 20 servlets.	6.x, 7.0, 8.1	240
Servlet Request Rates - Top 20	Shows the total servlet request rate being received by the top 20 servers.	6.x, 7.0, 8.1	242
Servlet Requests - Top 20	Shows the total servlet requests being received by the top 20 servers.	6.x, 7.0, 8.1	242
Transaction Rollbacks	Shows the average transaction rollback by day.	6.x, 7.0, 8.1	71
Transaction Throughput - Top 20	Shows the average transaction throughput for the top 20 servers by day.	6.x, 7.0, 8.1	76

Table 2 Availability

Report Title	Description	WebLogic Version	Metric
Server Availability	Contains a daily histogram showing the percentages of uptime and downtime for all servers.	6.x	2
Server Availability Details	Contains spectrum graphs showing minutes of uptime by day and hour for each WebLogic Server.	6.x	2

Removing the WLS-SPI Reporter Package

- 1 From the Control Panel, double-click on **Add/Remove Programs**.
- 2 From the Add/Remove Programs window, select **WLSSPI-Reporter**.
- 3 Click on **Remove**.
- 4 Complete the removal by following the instructions in the windows that display.

Integrating WLS-SPI with HP OpenView Performance Manager

To use Performance Manager, you must purchase and install it separately. To integrate WLS-SPI with OVP, do the following:

- 1 Configure the location of the graphing system within the WLS-SPI configuration. Refer to the online help for more information about setting the `GRAPH_SERVER` property.
- 2 Install the WLS-SPI graph package on the Windows system running Performance Manager:
 - a Insert the *hp OpenView OV Operations 7.20 for Windows smart plugins* CD into the CD drive of the Windows system running Performance Manager.

of these tables, you can use the View Graphs tool to launch the graph, which appears in your Web browser.

Table 3 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 4 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 5 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.

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

Graph Label	Metric Name	Metric Description
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.

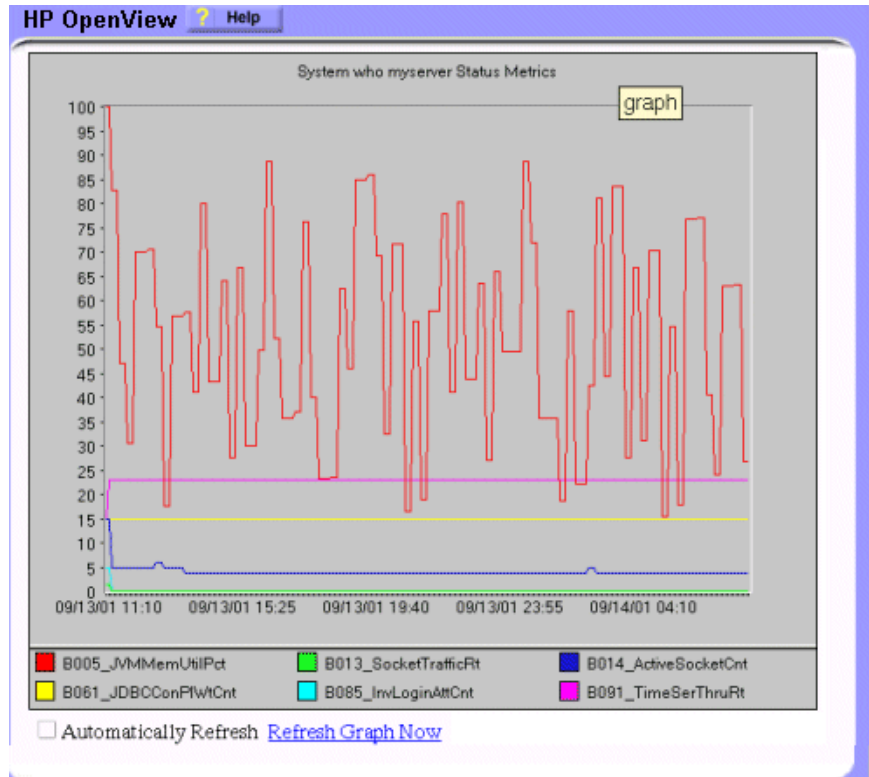
Table 6 Transaction: 70, 71, 72, 73, 74, 75, 76, 77

Graph Label	Metric Name	Metric Description
Transaction Average Time	B070_TrانAveTime	Average commit time for transactions.
Transaction Rollback Percent	B071_TrانRollbackPct	Percentage of transactions rolled back, based on the total.
Transaction Resource Error Rollback Percent	B072_TrانResErrRbPct	Percentage of the transactions rolled back due to resource error.
Transaction Application Error Rollback Percent	B073_TrانAppErrRbPct	Percentage of transactions rolled back due to application error.
Transaction Time Error Rollback Percent	B074_TrانTimErrRbPct	Percentage of transactions rolled back due to a timeout error.
Transaction System Error Rollback Percent	B075_TrانSysErrRbPct	Percentage of the transactions rolled back, based on system error.
Transaction Throughput Rate	B076_TrانThruRate	Number of transactions processed per second.
Transaction Heuristic Count	B077_TrانHeurCnt	Percentage of transactions returning a heuristic decision.

Launching the Web Page display with an Operator Action that Generate Graphs

Performance Manager graphs can be generated from most WLS-SPI alarm messages by double-clicking on the message, selecting the `Commands` tab, and selecting `Start` in the `Operator Initiated` section. The operator action launches your Web browser, which can then display a graph of the metric that generated the message as well as other related metrics.

Figure 1 A graph generated from an Operator Action



Specifying a Date Range

Within the Web page display, you can specify a date range of one day, one week, one month, or one year. Please see the online Help for instructions on changing display settings.

Removing the WLS-SPI Grapher Package

- 1 From the Control Panel, double-click on **Add/Remove Programs**.
- 2 From the Add/Remove Programs window, select **WLSSPI-Grapher**.
- 3 Click on **Remove**.
- 4 Complete the removal by following the instructions in the windows that display.

User Defined Metrics

Introduction to User Defined Metrics

The Smart Plug-in for BEA WebLogic Server (WLS-SPI) can collect data on roughly 55 metrics. However, you can add your own metrics. In defining your own metrics, you can monitor your own tools by registering application MBeans with the WebLogic MBean server and creating user-defined metrics (UDMs) that instruct the WLS-SPI to gather data from these MBeans.



A custom MBean must expose a “Name” attribute. The WLS-SPI uses this name as the identifying name for the MBean. If your custom MBean is a multi-instance MBean, then 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.

Please see the JMX documentation for more information about creating MBeans. Also refer to the WebLogic documentation for more information about registering MBeans.

You must understand the metric definitions DTD before creating your 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 WLS-SPI uses this file to parse and validate the XML file you create. The `MetricDefinitions.dtd` file content is described and a sample XML shown in the sections that follow.

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

Operating System	Directory
UNIX	<code>/var/opt/OV/wasspi/wls/conf/</code>
Windows	<code>%OvAgentDir%\wasspi\wls\conf\</code>



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+)>
```

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 >
```

Metric element attributes are described in the following table.

Attribute	Type	Required	Default	Description
id	ID	yes	--	The metric ID.
name	text	no	"no"	The metric name, used for graphing and reporting.
alarm	"yes" "no"	no	"no"	If yes, the metric value is sent to the agent via <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.

Attribute	Type	Required	Default	Description
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 74 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 >

```

MBean element attributes are described in the following table.

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.

Map element attributes are described in the following table.

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.

AttributeFilter element attributes are described in the following table.

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>
```

FromVersion and ToVersion Elements

The *FromVersion* and *ToVersion* elements are used to specify the versions 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.

- 1 If a *FromVersion* element is not present, no lower limit exists to the server versions supported by this metric.
- 2 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.
- 3 If a *ToVersion* element is not present, no upper limit exists to the *server* versions supported by this metric.
- 4 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  "*" >
```

FromVersion and *ToVersion* element attributes are described in the following table.

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

The following example defines a metric whose value is the ratio (expressed as a percent) of `Metric_1` to `Metric_3`.

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

The following example could be used 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, 6.5.

```
<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

Using the example above, a decision was made to make metric 10 a per-minute rate instead of a per-second rate as of server version 7.0. The changes that had to be made to the metric definitions are in bold type. 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 is a sample metric definitions file to illustrate how you might 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>

```


Create User-defined Metrics

Now that you have reviewed the structure for creating UDMs, do the following:

- Task 1: Disable graphing (if enabled)
- Task 2: Create a metric definitions file
- Task 3: Configure the metric definitions file name and location
- Task 4: Create a UDM policy group and policies
- Task 5: Deploy the policy group
- Task 6: Restart the OVO subagent
- Task 7: Enable graphing

Task 1: Disable graphing (if enabled)

If graphing has been enabled, disable it:

- 1 From the OVO console, select **Operations Manager** → **Nodes**.
- 2 Right-click on 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 70.



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

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

```
/var/opt/OV/wasspi/wls/conf/UDMMetrics-sample.xml (UNIX), or  
<%OvAgentDir%>\wasspi\wls\conf\UDMMetrics-sample.xml (Windows).
```

Task 3: Configure the metric definitions file name and location

In order for the UDM data collection to occur, the WLS-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>
```

where the path name should use only forward slashes ("/").

To add the UDM file name and its location to the WLS-SPI configuration:

- 1 From the OVO console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **WLSSPI - SPI Admin**.
- 2 Double-click on the **WLSSPI Configure** tool.
- 3 Select the managed node(s) on which the metrics definition file exists and select **Launch**.
- 4 A Console Status window appears.
- 5 The WLSSPI Configure Tool Introduction window appears. Read the information and select **Next**.
- 6 The WLSSPI Configure Tool window appears. 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, configure the property for each managed node selected in step 3:
 - a Single-click on **Default Properties** at the Defaults level or for a node.
 - b Select the **Set Configuration Properties** tab.
 - c From the **Select a Property to add** pulldown menu, select `UDM_DEFINITIONS_FILE`.
 - d Select **Add Property**.
 - e Enter the value (metric definitions file name and its fully-qualified path name, using forward slashes in the path name only).
 - f Select **Save**.
 - g Select **Next**.
- 7 The WLSSPI Configure Tool: Confirm Operation window displays. Select **OK** to configure the selected managed nodes.

Any 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 to 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 WLS-SPI policy group:
 - a From the OVO console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WebLogic <version>**.
 - b Right-click on the WLS-SPI group you want to use as a starting point, and select **Copy**.
 - c Right-click on **WebLogic <version>** and select **Paste**.
- 2 Rename the policy group:

Rename the new policy group according to how you plan to identify the new metric and collector policies. For example, you might include UDM in the name to clearly indicate that the group is made up of custom metric monitors.

 - a Right-click on the policy group and select **Rename**.
 - b Type in the new name.
- 3 Edit and rename each policy:
 - a Double-click on the policy you plan to use.
 - b Configure the collector policy command line (in the Program text box) to include the policy name and UDM metric number. Refer to "[Advanced Policy Customizations](#)" on page 36 for more information.
 - c Configure thresholds in the policy, as appropriate. Refer to "[Advanced Policy Customizations](#)" on page 36 for more information.
 - d Select **File** → **Save As**, and rename the policy according to your naming scheme.

The name you give the new *metric policy* in the group would contain each new UDM number. For example, a copy of WLSSPI-0001 could be called WLSSPI-0701.

The name you give the new *collector policy* would also contain the identifying name. You also include the policy name after the `-c` parameter on the command line as in the example:

```
wasspi_wls_ca -c UDM-61-15m -m 701
```

In this case the copied collector policy has been renamed:
UDM-61-15min

- 4 Delete all original policies from the new group:
 - a Right-click on the policy and select **Delete**.

Task 5: Deploy the policy group

- 1 Right-click on the policy group and select **All Tasks → Deploy on**.
- 2 Select the node(s) on which to deploy the policy group.
- 3 Click **OK**.

Task 6: Restart the OVO subagent

If you are using graphing (HP OpenView Performance Manager, also known as HP PerfView or MeasureWare, must be purchased and installed), you must restart the OVO subagent.



For the OVO subagent to collect data according to modified UDM definitions, you must first remove all OVO data from the database, including all SPI reporting, graphic, and system performance data. Please be certain you have no need of the data in the subagent's database before completing this task.

On the managed node running HP OpenView Performance Manager, to restart the OVO subagent, do the following:

- 1 Shutdown the OVO performance subagent by typing the command:
`opcagt -stop -id 12`

- 2 Delete all OVO performance subagent database files by typing:

AIX: `rm /var/lpp/OV/databases/coda*`

HP-UX and Solaris: `rm /var/opt/OV/databases/coda*`

Windows: `del \usr\ov\databases\coda*`

- 3 Restart the OVO performance subagent by typing:

```
opcagt -start -id 12
```

Task 7: Enable graphing

If you are using graphing (HP OpenView Performance Manager, also known as HP PerfView or MeasureWare, must be purchased and installed), enable data collecting for UDM graphing:

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

Allow sufficient collection intervals to occur before attempting to view graphs.

Troubleshooting WLS-SPI

Introduction

This chapter covers troubleshooting the Smart Plug-in for BEA WebLogic Server (WLS-SPI).

Error messages (listed by number) are available in the online help.

Log and Trace Files

Management Server

The following log file is found on the management server (typically, `\<OvInstallDir%\ is \Program Files\HP OpenView\`):

File Type	Log
Filename	<code>\<OvInstallDir%\install\WASSPI\WLSSPI\English\Discovery\log\ <managed_node>_disc_server.log</code>
Description	Records the updates done by the WLSSPI Discovery policy to the management server's configuration for each managed node. Log files are overwritten each time the discovery policy is run on the managed node. Logging to this file is always enabled.

UNIX Managed Nodes

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

File Type	Log
Filename	<code>/<OvAgentDir>/log/javaagent.log</code>
Description	OVO discovery agent log file containing the status of the OVO discovery agent. By default, logging to this file is enabled at <code>LOG_LEVEL 3</code> . Set the <code>LOG_LEVEL</code> variable in <code><OvAgentDir>/conf/svcDisc/OvJavaAgent.cfg</code> to 6 or higher (up to 9) to capture troubleshooting information (the higher the number, the more information is collected). To disable this log, set the <code>LOG_LEVEL</code> to 0 . Additional information can be configured in this file to define log file size and the number of archived files kept. By default, the log file size is 1MB and five archived versions are kept.

File Type Log

Directory /<OvAgentDir>/wasspi/wls/log/config.log

Description Records output from the WLS-SPI configuration scripts.

File Type Log

Directory /<OvAgentDir>/wasspi/wls/log/errorlog

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

File Type Trace

Filename /tmp/wasspi_wls_disc.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 <OvAgentDir>/bin/instrumentation/wasspi_wls_discoveryUnix.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_discoveryUnix.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** /<OvAgentDir>/wasspi/wls/log/trace.log (archived files have a three digit number appended to the filename)
- Description** Trace file used by your HP support representative. By default, tracing to this file is disabled. To enable this tracing, use the WLSPI Trace - Start tool.

Windows Managed Nodes

The following log and trace files are found on the managed nodes running on Windows (typically, <OvAgentDir%> is
\\Program Files\\HP OpenView\\Installed Packages\\{790 ...}\\):

- File Type** Log
- Filename** \\<OvAgentDir%>\\log\\javaagent.log
- Description** OVO discovery agent log file containing the status of the OVO discovery agent. By default, logging to this file is enabled at LOG_LEVEL 3. Set the LOG_LEVEL variable in <OvInstallDir%>\\conf\\svcDisc\\OvJavaAgent.cfg to 6 or higher (up to 9) to capture troubleshooting information (the higher the number, the more information is collected). To disable this log, set the LOG_LEVEL to 0. Additional information can be configured in this file to define log file size and the number of archived files kept. By default, the log file size is 1MB and five archived versions are kept.
- File Type** Log
- Directory** \\<OvAgentDir%> \\wasspi\\wls\\log\\config.log
- Description** Records output from configuration scripts.

File Type	Log
Directory	\<%OvAgentDir%> \wasspi\wls\log\errorlog
Description	Records WLS-SPI error messages. This log file is monitored by WLS-SPI policies.

File Type	Trace
Filename	C:\temp\wasspi_wls_disc.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 \<%OvInstallDir%>\bin\instrumentation\wasspi_wls_discoveryWin.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_discoveryWin.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	\<%OvAgentDir%> \wasspi\wls\log\trace.log (archived files have a three digit number appended to the filename)
Description	Trace file used by your HP support representative. By default, tracing to this file is disabled. To enable this tracing, use the WLSSPI Trace - Start tool.

The WLSSPI Discovery Policies

Problem	The WLSSPI Discovery policies do not automatically discover and update the WLS-SPI configuration.
Solution	<ol style="list-style-type: none"> 1 Check for errors in the message browser of the managed nodes not being discovered. Follow the instruction text of any error messages displayed. 2 Check if the WLSSPI Discovery policies are still being deployed: <ul style="list-style-type: none"> From the OVO console, select Operations Manager → Policy management → Deployment jobs. • If the state of a WLSSPI Discovery policy is <i>Active</i>, then the policy is still being deployed. Wait for the deployment of the policy to complete. • If the state of a WLSSPI Discovery policy is <i>Suspended</i> or <i>Error</i>, 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 <p>WASSPI-302: Updating WLS-SPI configuration in OVO server for <node></p> <p>WASSPI-303: The SPI configuration for <node> was updated by discovery in the OVO server. The updated configuration is as shown below</p> <p>If these messages are present, the WLSSPI Discovery policies have been successfully deployed. If these messages are not present, either the policies were not successfully deployed or the <code>AUTO_DISCOVER</code> check box has not been selected in the configuration editor.</p> <p>Continue to troubleshoot the problem by reading the rest of this section.</p> 3 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 WLS-SPI. 4 Verify the WebLogic application server status. The application server must be running. See "Task 2: Verify the Application Server Status" on page 18 for more information.

Solution (cont.)	<p>5 Verify that the LOGIN/PASSWORD properties are set (refer to the online help) and that the WebLogic user configured has the correct permissions (see "Task 3: Collect Application Server Information" on page 18).</p> <p>6 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.</p> <p>7 Verify the Java home directory (see "Verifying the Java Home Directory" on page 96).</p> <p>8 Verify that the discovery agent is running on the managed node:</p> <ul style="list-style-type: none"> a Run the command <code>opcagt -status</code> 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 <code>opcagt -start -id 13</code> <p>9 If you are running WebLogic Server 7.0 or higher and did not save the domain configuration file (for example, <code>config.xml</code>) in the default directory (<code><BEA_Home_Dir>/user_projects/<WebLogic_Domain_X>/</code>, where <code><BEA_Home_Dir></code> is the directory that contains the <code>registry.xml</code> file), then do one of the following:</p> <ul style="list-style-type: none"> • Manually set the server using the WLSSPI Configure tool, OR • Manually configure ADMIN_PORTS, the port number(s) of the WebLogic Admin server(s) listed in the domain configuration file, using the WLSSPI Configure tool. The global LOGIN and PASSWORD must be configured for the node on which these WebLogic Admin servers are running.
-------------------------	--

<p>Solution (cont.)</p>	<p>10 On a UNIX managed node, verify that <code>BEA_HOME_LIST</code> and <code>HOME_LIST</code> directory path names do not include spaces. The discover process currently does not support spaces in directory names.</p> <p>11 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 95, uninstall and redeploy the Discovery policies:</p> <ul style="list-style-type: none"> a From the OVO console, select Operations Manager → Policy management → Policy groups → SPI for WebLogic Server. b Right-click on WLSSPI Discovery and select All Tasks → Uninstall all. c Select the node(s) from which to uninstall the Discovery policies. d Click OK. e Redeploy the Discovery policies in the order shown in "Manually Deploying the Discovery Policies" on page 95. The policies must be deployed in order. If you deployed the policies as a group, the policies may not be deployed in the correct order. <p>12 Verify that the WLSSPI Configure tool is not running and/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.</p>
--------------------------------	--

<p>Problem</p>	<p>The WLSSPI Discovery policies are adding inaccurate information to the configuration</p>
<p>Solution</p>	<ul style="list-style-type: none"> 1 Verify <code>LOGIN</code> and <code>PASSORD</code> are correct. Refer to "Task 3: Collect Application Server Information" on page 18 for more information. 2 Verify the Java home directory. Refer to "Verifying the Java Home Directory" on page 96 for more information. 3 Update the configuration and uncheck the <code>AUTO_DISCOVER</code> check box in the configuration editor to prevent the WLSSPI Discovery policies from overwriting the configuration information.

Manually Deploying the Discovery Policies

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

- 1 From the OVO console, select **Operations Manager** → **Policy management** → **Policy groups** → **SPI for WebLogic Server** → **WLSSPI Discovery**.
- 2 Right-click on **WLSSPI-Messages** and select **All Tasks** → **Deploy on**.
- 3 Select the node(s) on which to deploy the auto-discovery policies.
- 4 Click **OK**.
- 5 Right-click on **WLSSPI Service Discovery** and select **All Tasks** → **Deploy on**.
- 6 Select the node(s) on which to deploy the auto-discovery policies.
- 7 Click **OK**.

Verifying Discovery Policy Deployment is Currently Active

To check if the WLSSPI Discovery policies are still being deployed, from the OVO console, select **Operations Manager** → **Policy management** → **Deployment jobs**.

- If the state of a WLSSPI Discovery policy is *Active*, then 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. Refer to "[The WLSSPI Discovery Policies](#)" on page 92 to diagnose the problem.
- If the WLSSPI Discovery policies are not listed, check the message browser for the following messages

WASSPI-302: Updating WLS-SPI configuration in OVO server for *<node>*

WASSPI-303: The SPI configuration for *<node>* was updated by discovery in the OVO server. The updated configuration is as shown below

If these messages are present, the WLSSPI Discovery policies have been 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.

Verifying the Java Home Directory

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):

- `JAVA_HOME` is correctly defined in the configuration. To edit/view the configuration, run the WLSSPI Configure tool:
 - a From the OVO console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **WLSSPI - SPI Admin**.
 - b Double-click on **WLSSPI Configure**.
 - c Select the node(s) to configure.
 - d Click **Launch**.
- 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, do the following:

- a Select **Start** → **Settings** → **Control Panel**.
- b Double-click on **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, do the following:

- a Type `echo $JAVA_HOME`

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

The Configuration

Problem	The WLS-SPI configuration does not have complete or accurate information for a WebLogic managed server.
Solution	Verify <code>LOGIN</code> and <code>PASSWORD</code> are correct. Refer to " Task 3: Collect Application Server Information " on page 18 and the online help for more information. This is the most common reason for incorrect information for a WebLogic managed server running on a remote node (not running on the OVO managed node).

Problem	The WLSSPI Discovery policies overwrite the configuration with inaccurate information.
Solution	Update the configuration and uncheck the <code>AUTO_DISCOVER</code> check box in the configuration editor to prevent the WLSSPI Discovery policies from overwriting the configuration information.

Verifying the Node Name

Verify that the node name specified in a node or group block matches the primary node name configured in OVO. To display the primary node name, do the following:

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

Tools

Message	Configuration variable SERVER<n>_START_CMD missing for server "Default Server"
Solution	Before you can successfully run the WLSSPI Start tool, you must set the START_CMD and USER properties. Set these properties using the WLSSPI Configure tool. Refer to the online help for more information about this tool.

Message	Configuration variable SERVER<n>_STOP_CMD missing for server "Default Server"
Solution	Before you can successfully run the WLSSPI Stop tool, you must set the STOP_CMD and USER properties. Set these properties using the WLSSPI Configure tool. Refer to the online help for more information about this tool.

C

Coda, overriding with MeasureWare Agent **57**

collection intervals
 changing for all servers **39**
 changing for selected metrics **40**

collector/analyzer
 description of **30**

command
 wasspi._wls_ca, description of **30**

customizations
 creating new policies **42**

D

deinstallation
 removing WLS-SPI **16**

directories
 locations for trace file/error logs **90**

F

files, locations on management server/
 managed nodes **88, 90**

G

graphs
 for UDMs **85**
 generating with Reporter's graphing capabilities **62**
 generating with the HP OpenView graphing tool **55**
 instructions for manually generating **63**
 launching display within Web browser **66**
 list of metrics for server status graph **63**
 Performance Manager requirement for **56**
 policies available for **63**
 showing alarm conditions **63**
 specifying date range within **66**

H

HP OpenView Reporter, integrating WLS-SPI with **55**
HP Performance Manager, using WLS-SPI with **55**

I

instances
 differentiating server instances within a collector policy **39**

L

Linux, monitoring WebLogic Server installed on **47**

log file policies
description of **28**

M

messages
policy configuration for **35**

Message Source policy groups, description of WLS-SPI groups **28**

metric element attributes for creating user defined metrics (UDMs) **71**

metric policies
description of **29**

metrics
modifying collections in the collector policy **37**

metrics policies
description of **28**

O

OpenView Performance Agent (MeasureWare Agent), using **57**

operator actions
graphs generated from **63**

P

Performance Manager, using WLS-SPI with **55**

policies

customizing message displayed for alerts **35**

customizing message text **35**

customizing thresholds **31**

customizing with the tag option **42**

description of **28**

modifying **30**

re-installing defaults **43**

policy groups

changing collection intervals **40**

creating custom with the tag parameter **42**

description **28**

proxy configured monitoring **47**

R

remote monitoring
requirements for **50**

remote systems, monitoring **47**

Reporter

integrating WLS-SPI to work with **55**

setting up WLS-SPI to work with **58, 63**

reports

annotations **44**

generated from HP OpenView Reporter **60**

sample automatic action **46**

showing WebLogic Server performance **60**

tool-generated **45**

using HP OpenView Reporter to generate **55**

reports (automatic action)

how they are generated **43**

S

servers
 setting thresholds for different **41**

T

tag option
 creating custom policy groups with **42**

thresholds
 customizing **31**
 exceeded
 viewing graphs resulting from **63**
 settings for different servers **41**

tools
 reports generated from **45**

U

UDMs, *please see user defined metrics*

unsupported platforms, monitoring
 WebLogic on **50**

user defined metrics
 graphing **85**
 MBean Element, description of **72**
 metric definitions element, description of **70**
 metric element attributes, description of **71**
 metric element, description of **71**
 sample XML file for **78**

W

wasspi_wls_ca, description of **30**

WebLogic Server
 instances, differentiating within the
 collector policy **39**

