HP OpenView Smart Plug-in for BEA WebLogic Server

For HP OpenView Operations for Windows

Software Version: 04.10.000

Configuration Guide

Document Release Date: June 2006
Software Release Date: June 2006

invent

Legal Notices

Warranty

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

The information contained herein is subject to change without notice.

Restricted Rights Legend

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

Copyright Notices

Trademark Notices

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.

Documentation Updates

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

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

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

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

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

Support

Please visit the HP OpenView support web site at:

http://www.hp.com/managementsoftware/support

This web site provides contact information and details about the products, services, and support that HP OpenView offers.

HP OpenView online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support site to:

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

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

To find more information about access levels, go to:

http://www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

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

Contents

1 The WLS-SPI

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. Read about WLS-SPI concepts in the online help. The following topics are covered in the online help:

- Concepts
- Configuration editor
- Tools
- Policies
- Reports and graphs
- Error messages

2 Installing, Upgrading, and Removing WLS-SPI

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

Installing WLS-SPI

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 Smart Plug-ins and Integration Modules, New and Upgraded, Volume 1 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 New and Upgraded Smart Plug-ins CD for OpenView Operations Installation / Upgrade Guide for Windows* for more information.

Upgrading WLS-SPI

Detailed information about supported software, enhancements, fixes, and known problems and workarounds is available in the *HP OpenView Smart Plug-in for BEA WebLogic Server Release Notes*, located on the Smart Plug-ins, New and Upgraded CD, Volume 1, in `\Documentation\Releasenotes\WebLogic_AppServer_Releasenotes.html`.

Policy Changes

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

The following policies are modified:

- In the Logfiles policy group the Weblogic Logs policy.

- In the Monitors group, the existing polices WLSSPI-81-1hr, WLSSPI-70-1hr, WLSSPI-61-1hr, WLSSPI-81-15min, WLSSPI-70-15min, WLSSPI-61-15min WLSSPI-81-05min, WLSSPI-70-05min, and WLSSPI-61-05min are replaced by three new policies:
  - WLSSPI-15min
  - WLSSPI-05min
  - WLSSPI-1h

Before Starting

Before upgrading to WLS-SPI version 04.10.000, you must:

1. Upgrade OVOW to version 7.5.
2. Take a backup of the existing WLS-SPI configuration file from the location:

   `\<%OvOWShareInstallDir%>\SPI-Share\wasspi\wls\conf\SiteConfig`

   For example: *`<%OvOWShareInstallDir%>`* can be `C:\Program Files\HP OpenView\Data\shared\`
3. Install the new version of the WLS-SPI by running the InstallShield Wizard (installer).

If the installer detects that an older version of the WLS-SPI is installed, it upgrades to the new version by doing the following:

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

Upgrading WLS-SPI

To upgrade to WLS-SPI version 04.10.000, do the following:

1. Install the WLS-SPI version 04.10.000 software:
   a. Insert the Smart Plug-ins and Integration Modules, New and Upgraded, Volume 1 CD into the CD drive of the management server. The InstallShield Wizard starts.
   b. From the first screen, select **Next**.
   c. In the Program Maintenance window, select **Install products**.
   d. In the Product Selection window, select the check box next to **BEA WebLogic** and select **Next**.
   e. 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.
2. Refresh the `SPI for WebLogic Server` node group:
   a. In the console tree, select **Tools** → **SPI for WebLogic Server** → **WebLogic Server Admin** → **Create WLSSPI Node Groups**.

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

3. Uninstall older versions of the modified WLS-SPI policies (see Policy Changes on page 11) from existing nodes (versions 4.0 and lower). This includes any customized policies that you may have moved outside of the old `SPI for WebLogic Server` policy group and does not reside in the `SPI for WebLogic Server - Saved Policies` policy group.
   a. In the console tree, select **Policy management** → **Policy groups**.
   b. Select the **SPI for WebLogic Server - Saved Policies** policy group.
   c. Right-click the policies that have been modified in this version. See Policy Changes on page 11.
   d. Select **All tasks - Uninstall from**.
   e. Select the node group **SPI for WebLogic Server**. Select **OK**.
   f. Repeat these steps for any customized WLS-SPI policies that do not reside in the `SPI for WebLogic Server - Saved Policies` policy group.
4. Deploy new instrumentation to the `SPI for WebLogic Server` node group:
   a. Select the **SPI for WebLogic Server** node group and right-click.
   b. Select **All Tasks** → **Deploy instrumentation**.
   c. Select **SPI for WebLogic Server**.
   d. Verify that the Remove existing instrumentation before deploying new instrumentation checkbox is clear.
   e. Select **OK**.
5. Customize the modified policies in the `WLSSPI` policy group to match old customized policies. Compare old and new policies by opening them side by side.
6. Deploy the modified policies.

   To deploy the modified polices in the `WLSSPI` policy group to all WebLogic nodes:

   a. Drag and drop the modified policies on the SPI for WebLogic Server node group.

   To deploy the modified policies to selected WebLogic nodes:

   a. Right-click the modified policies.
   b. Select **All Tasks** → **Deploy on ...**
   c. Select the node(s) on which to deploy the policy group.
   d. Click **OK**.

After upgrading WLS-SPI, if you add an instance of BEA WebLogic Server 9.0 or 9.1 on a managed node, then you must run the Discover WebLogic tool on that node.

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: Remove All WLS-SPI Policies from the Managed Nodes
- Task 2: Remove WLS-SPI Node Groups on the Management Server
- Task 3: Remove Any SPI for WebLogic Server Groups
- Task 4: Remove the WLS-SPI Software from the Management Server

Task 1: Remove 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 **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 Create WLSSPI 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 the node group and select **Delete**.
4. Click **Yes** to continue the delete operation.

Task 3: Remove Any SPI for WebLogic Server Groups

1. Under **Tools**, **Policy management** → **Policy groups**, **Reports & Graphs** → **Reports**, and **Reports & Graphs** → **Graphs**, look for the **SPI for WebLogic Server** group.
2. If the **SPI for WebLogic Server** group exists, remove it and its contents.
3. If you upgraded the SPI, also remove the **SPI for WebLogic Server - Saved Policies** group from **Policy management** → **Policy groups**.

Task 4: Remove the WLS-SPI Software from the Management Server

1. Insert the HP OpenView Smart Plug-ins 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.

3 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:

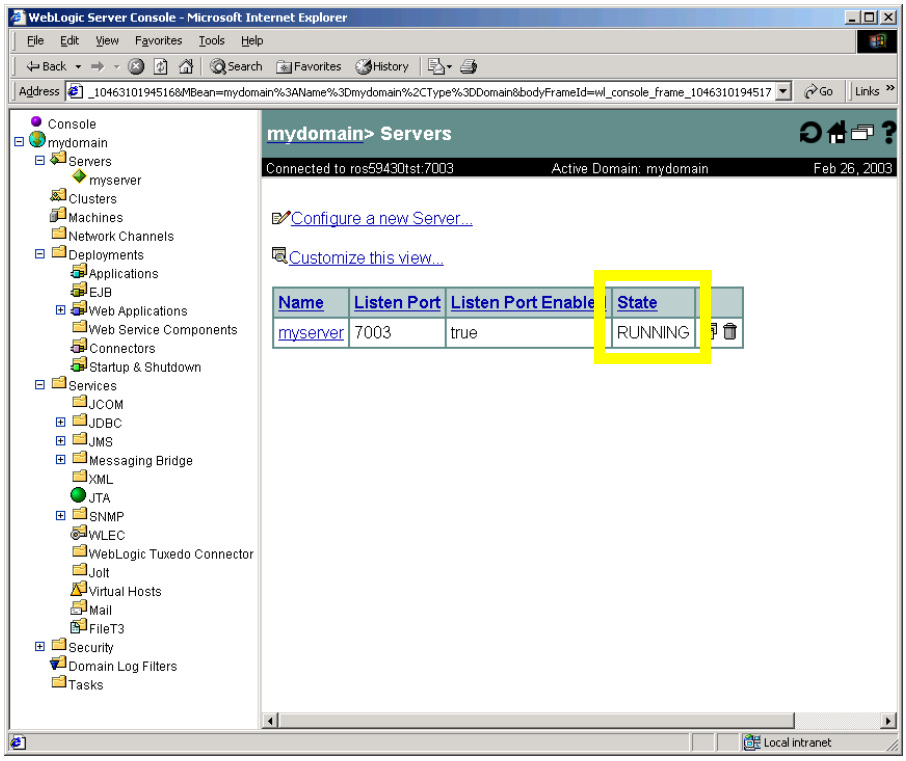
1. Install the OVO agent on the node. Refer to the OVO online help topic “Agent Installation on UNIX computers” for more information.
2. Specify each WebLogic Server node on UNIX to be managed. Refer to the OVO online help topic “Configure Managed Nodes” for more information.

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 names on which the WebLogic Administration Servers 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.

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

WebLogic Server Version 6.x

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

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

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

A user that belongs to the Monitors group cannot use the Start/Stop WebLogic applications (to start or stop WebLogic Servers from the OVO console) nor perform the JMX call "set" when implementing JMX actions (to assign a value to a specifed attribute if you are creating UDMs).

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 for WebLogic servers running on HTTP mode, complete the following tasks. For information about configuring WLS-SPI for WebLogic servers running on HTTPS mode see Configuring WLS-SPI for WebLogic Servers running on HTTPS on page 24.

- Configuration Prerequisite
- Task 1: Run Discover WebLogic

- Task 2: Verify the Discovery Process
- Task 3: Set Additional Properties

Configuration Prerequisite

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

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

To deploy these instrumentation files:

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

To verify if these files have been deployed successfully check Deployment Jobs under Policy management. There must be no error messages.

Task 1: Run Discover WebLogic

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

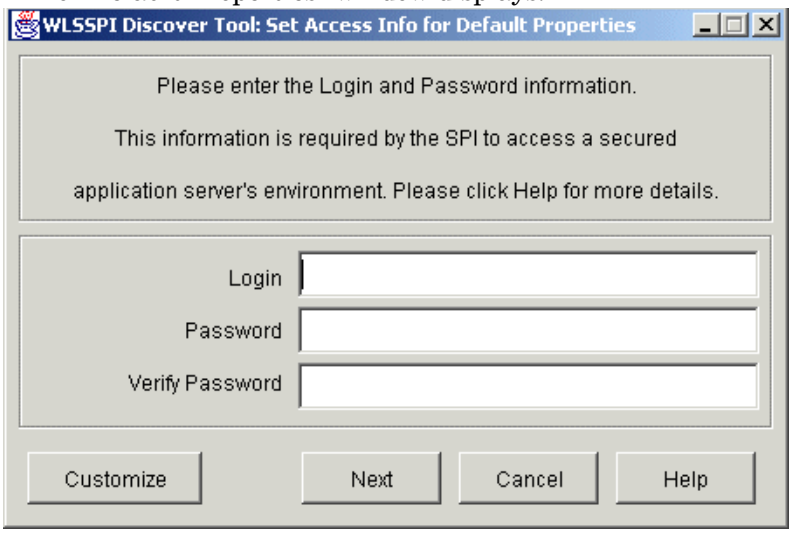
To run Discover WebLogic, do the following:

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



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 Discover WebLogic 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 appear click **Close**.

If the window displays an error message, refer to The WLSSPI Discovery Policies on page 79 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 23.

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** → **SPI Admin**.

b Double-click **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 26. 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 properties may be needed by the discover process. Refer to the online help for a complete definitions of the properties.

If multiple versions of the WebLogic Server are installed on a node, set:

- JAVA_HOME - The default directory where Java is installed. Use the highest version of Java available. If you are running WebLogic Server version 9.0, you must use Java version 1.4.1 or higher.

If more than one version of Java is installed, set

- JAVA_HOME - Use the highest version of Java available. If you are running WebLogic Server version 9.0, you must use Java version 1.4.1 or higher.

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

If you did not use BEA's installation scripts to install the WebLogic Server and service packs, set:

- HOME_LIST - A list of directories where the WebLogic Server is installed.
- JAVA_HOME - The default directory where Java is installed. If you are running WebLogic Server version 9.0, you must use Java version 1.4.1 or higher.

If WebLogic Server is configured to a virtual IP address, set:

- ADDRESS - The domain name or IP address where the WebLogic Server is listening.

If a remote WebLogic Server is listening on a virtual IP, set:

- NODE_NAMES - The virtual IP address where the server is listening. If not set, the remote WebLogic Server is not discovered

If the WebLogic Server's domain configuration file is not located in the default directory, set:

- ADMIN_PORTS - The port number(s) of WebLogic Admin server(s). The domain configuration file (`config.xml`) may be located in the default directory:

— <*WebLogic_Install_Dir*>/`config`/<*WebLogic_Domain*>/ (WebLogic 6.x)

— <*BEA_Home_Dir*>/`user_projects/domains`/<*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.

If the BEA registry.xml file is not accurate or cannot be found, set:

- HOME_LIST - A list of directories where the WebLogic Server is installed.

To reduce the amount of time for the discover process to run, set:

- EXCLUDE_SAMPLES - By setting this property to "true," the WebLogic Server sample programs are not included in the discover process. It is recommended that you set this to "true" at the default properties level to reduce the amount of time it takes for the discover process to run. The discover process can take several minutes to complete.

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

1. Repeat Task 1: Run Discover WebLogic 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 22.
3. If verification is successful, go to Additional WLS-SPI Configuration on page 26.

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

Configuring WLS-SPI for WebLogic Servers running on HTTPS

When configuring WLS-SPI for WebLogic servers running on HTTPS mode, there can be two scenarios:

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

  To configure:

  a. From the OVO console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **SPI Admin**.
  b. Double-click **Discover WebLogic**.
  c. Select the managed nodes on which the WebLogic Administration Servers are running.
  d. Click **Launch**. The Console Status window opens. After a few seconds the Introduction window opens. This window contains brief information about the Discover WebLogic tool.
  e. Click **Next**. A second Introduction window opens. This window conatins information on the properties that may be required for the discovery process to function.
  f. Click **Next**. The Set Access Info for Default Properties window opens.
  g. Set the BEA WebLogic LOGIN and PASSWORD properties.
  h. Set the ADMIN_PORTS and PROTOCOL property. The default value for PROTOCOL is **t3s**.

ADMIN_PORTS is the SSL port on which the application server is listening. The PROTOCOL property specifies if the application server port is using SSL or non-SSL.

i If required, set the PASSPHRASE and KEYSTORE properties. Click **Next** to run discovery on the selected nodes.

KEYSTORE is the path to the SSL trust keystore file.

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

For more information about setting the properties refer to the WLS-SPI online help.

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

After Discovery is successful:

a From the OVO console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **SPI Admin**.

a Double-click **Configure WebLogic**.

b Select the new servers that are discovered.

c Click **Launch. The Console Status window opens. After a few seconds the WLSSPI Configure Tool:Introduction window opens. This window contains information about the Configure tool.**

d Click **Next**. The Configuration Editor opens.

e Set the value of the PROTOCOL property to t3 for the WebLogic servers. If you do not change the value of PROTOCOL property for the servers it will take up the default value (t3s) set for the Admin Server. For more information about configuration property precedence refer to the section Configuring OAS SPI in the online help.

f Create a MONITOR USER in BEA under the active Security Realm

g Use the newly created user credentials in the SERVER properties (set the LOGIN and PASSWORD properties of the server similar to the values set for Monitor User).

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

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

  You can configure the admin server and the WebLogic servers as mentioned in the previous scenario. But, in this case you must not set the PROTOCOL property to t3 as the servers are running on the same mode as the admin server. The PROTOCOL property must have the default value t3s only.

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 Start WebLogic and Stop WebLogic tools from the OVO console, set the START_CMD and STOP_CMD properties.

If you are configuring user-defined metrics, refer to the JMX Metric Builder release notes for additional installation information and the JMX Metric Builder online help for additional configuration information.

If HP OpenView Performance Manager is installed (must be purchased separately and is *not* the version of OVPM that is included with OVO), refer to Integrating WLS-SPI with HP OpenView Performance Manager on page 56 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 51 for installation and configuration information.

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** → **SPI Admin**.
2. Double-click **Configure WLSSPI**.
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**.

4 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:

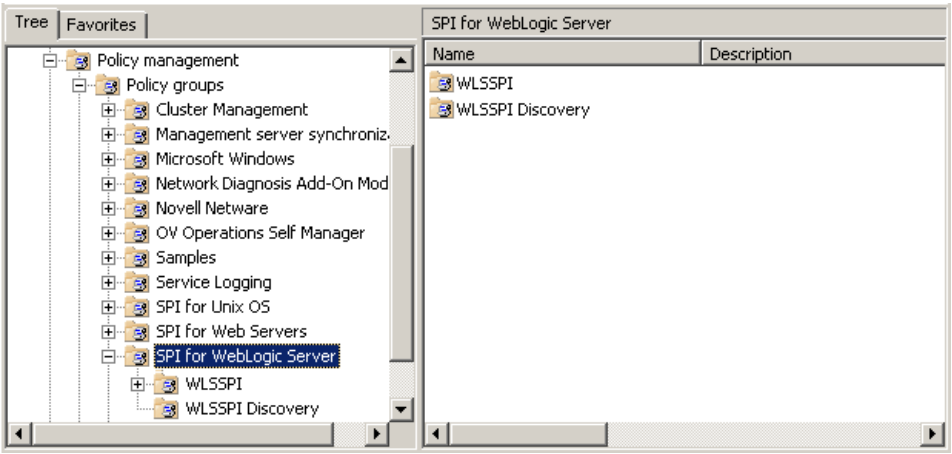
- Using the WLS-SPI Policies
- Basic Policy Customizations
- Advanced Policy Customizations
- Reinstalling WLS-SPI Policies
- Using Policies/Tools to Generate Reports
- Monitoring WebLogic on Unsupported Platforms

Using WLS-SPI Policies

You can customize WLS-SPI policies (although they work without any modifications). To help you understand how you might customize the WLS-SPI policies, the following sections cover WLS-SPI policy groups and general information about the OpenView Operations policies.

WLS-SPI Policy Groups

WLS-SPI policy groups provide a means of organizing various OVO policies.



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

- **Metric (monitor) policies**: generate messages according to threshold settings that monitor WebLogic availability and performance metrics.
- **Collector policies**: pertain to all metrics scheduled to be collected in the specified collection interval (grouped according to collection intervals).
- **Logfile policies**: generate messages according to logfile and error text detected in both the WebLogic Server logfiles and in the WLS-SPI logfiles.

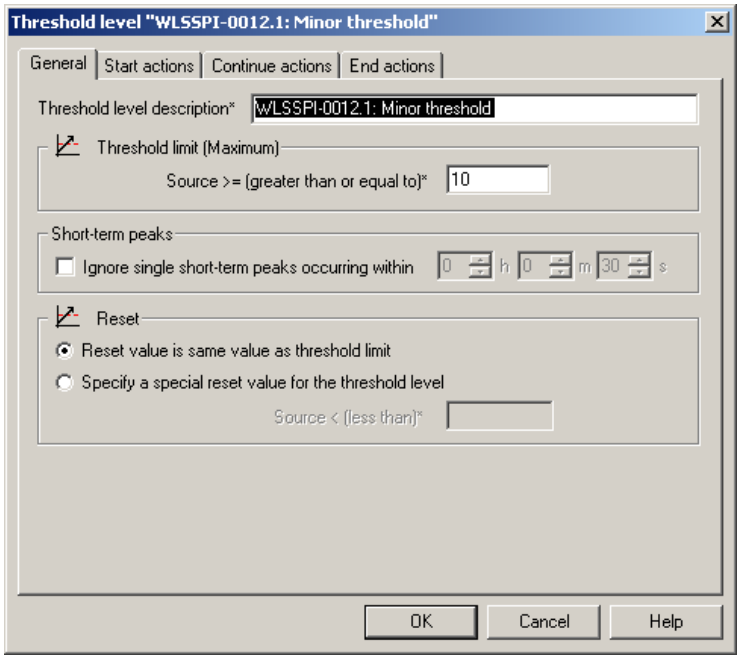
The WLSSPI Discovery policy automatically updates the WebLogic configuration information in the service map. Uncheck the AUTO_DISCOVER check box in the configuration editor if you do not want the discovery policy to automatically overwrite this configuration information. Refer to the online help for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

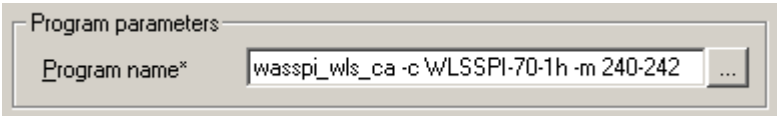
Figure 1 Threshold Level Window



Collector policies define all metrics for the WebLogic Server application that are scheduled for collection at the specified interval. Identified as a "metric policy" in the OVO lists of policies, 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, where the collection interval is 1h or one hour). When you open any collector policy, you see all metrics (by number) collected within the interval following the **-m** option of the collector/analyzer command `wasspi_wls_ca`.

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 metric(s) collected (`- 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, such as 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, open a metric policy. To schedule or delete a metric from data collection, open the collector policy. These basic customizations are covered in this section.

➤ Make copies of the original policies so that the default policies remain intact

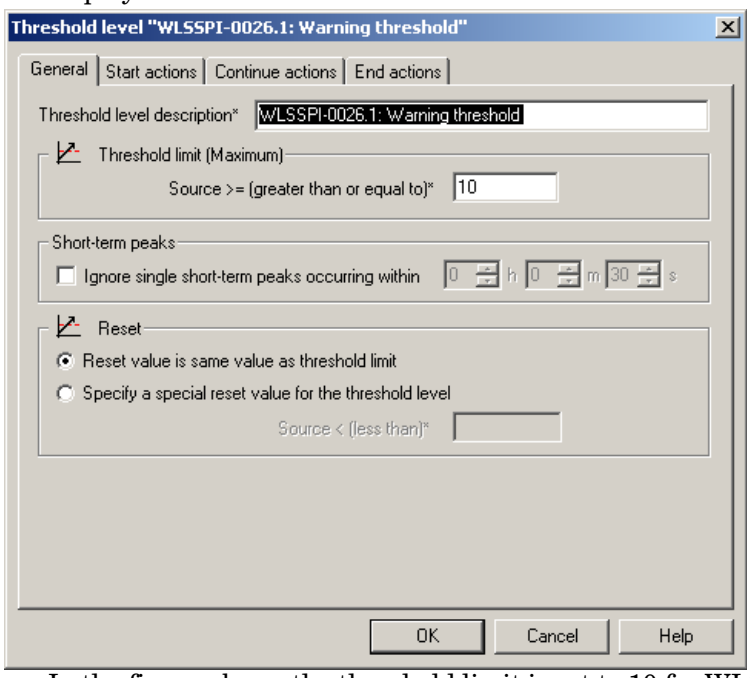
Modifying Metric Policies

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** → **WLSSPI** → **Metrics**.
2. Double-click a policy.
3. Select the Threshold levels tab.
4. From the Level summary pane, click Threshold level. The Threshold level window displays.



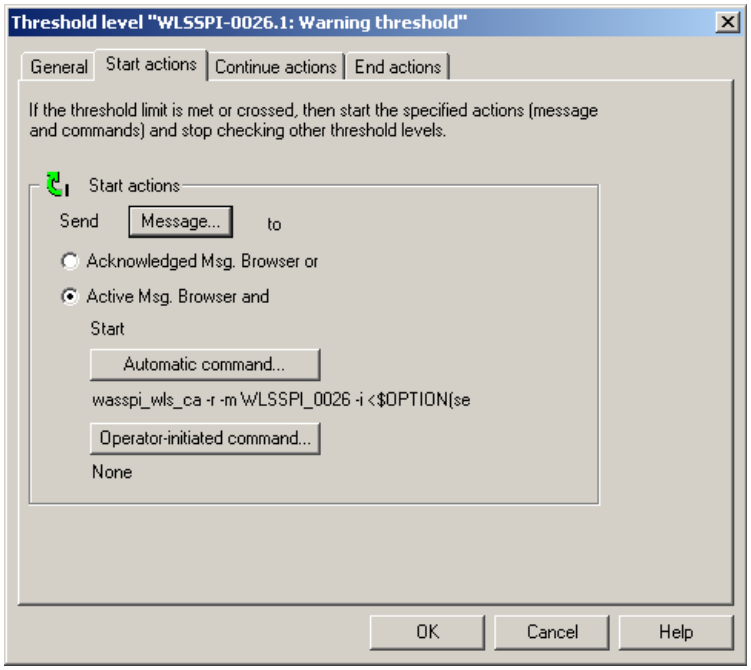
In the figure above, the threshold limit is set to 10 for WLSSPI-0026. The incoming values for this metric show the total number of times per minute clients must wait for an available (Enterprise Java) bean. A value of more than 10 would start to impact the server response time the client experiences, generating a warning message.

The following attributes can be modified:

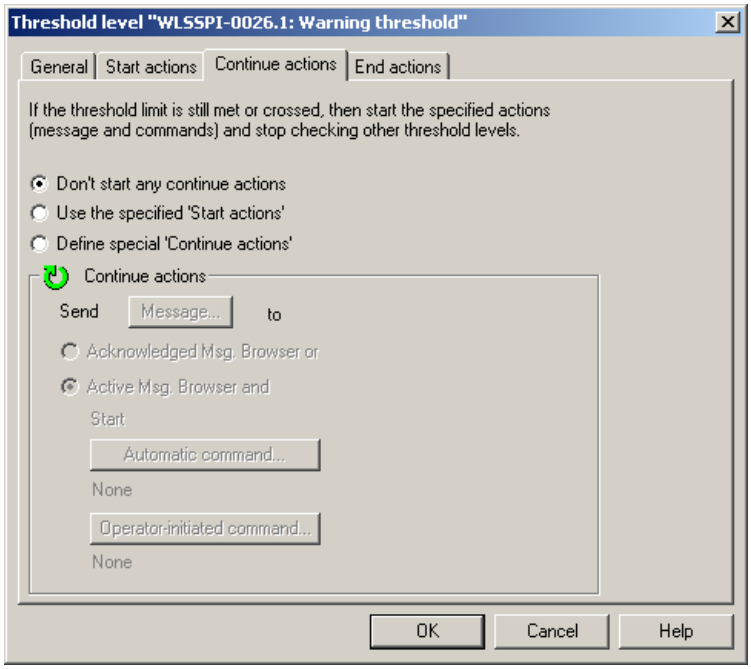
- **Threshold limit**. The value that triggers a message if it is met or crossed.
- **Short-term peaks**. A minimum time period over which the monitored value must exceed the threshold before generating a message. For a message to be sent, the value must be greater than the threshold each time the value 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 one of the actions tabs to set the following:

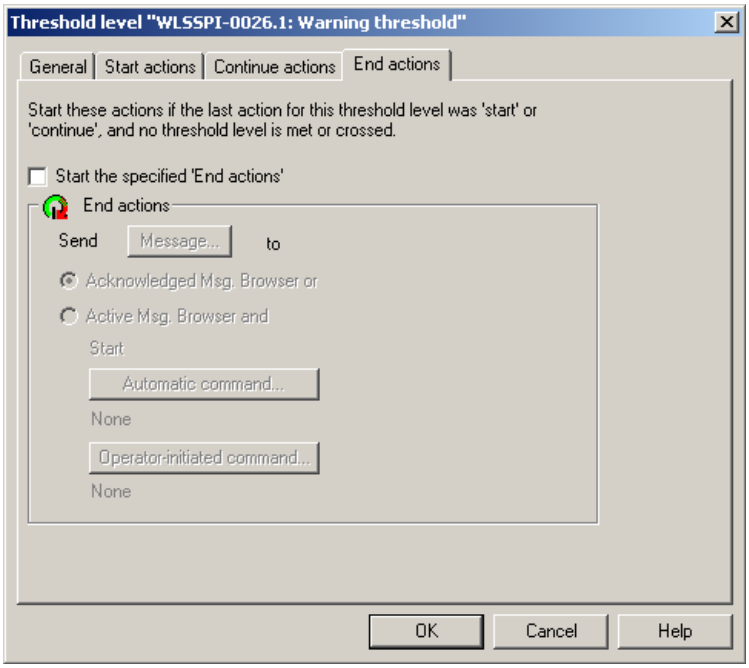
- **Start actions**. Actions carried out the first time that the threshold is crossed.



- **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:

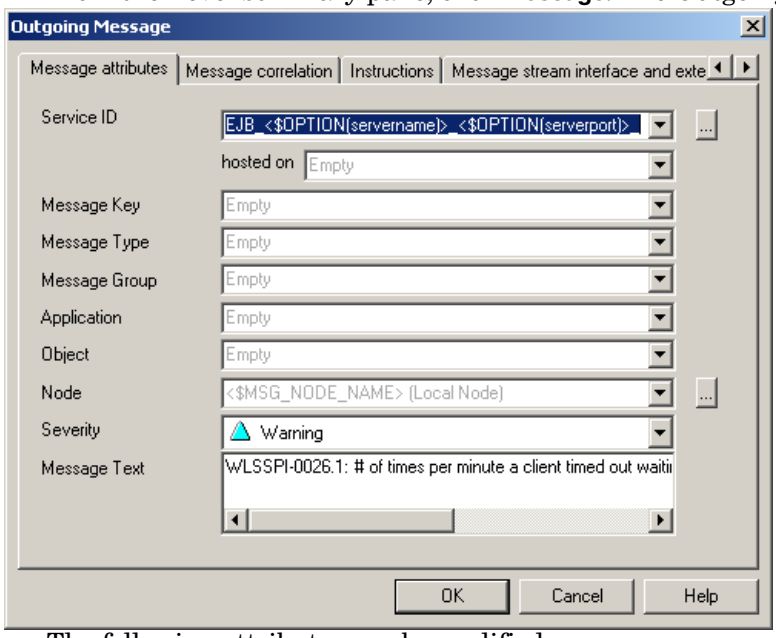
- **Automatic command**. A command run when the rule is matched. The automatic command delivered with the WLS-SPI generates a snapshot annotations report that shows the data values at the time the action was triggered from an exceeded threshold. You can view the report in the message annotations.
- **Operator-initiated 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 a policy.
2. Select the Threshold levels tab.

3 From the Level summary pane, click **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—beginning with $ and surrounded by <> brackets—in a message.

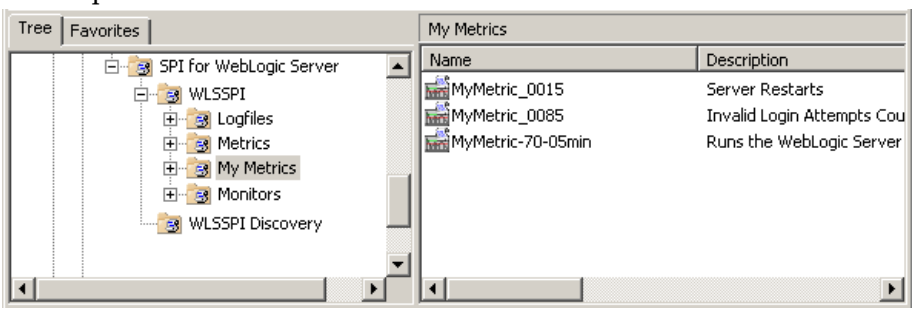
Advanced Policy Customizations

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

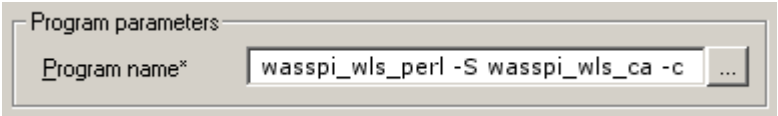
1. Determine the metrics you want to change and what policies within the group you want to use.
2. Create a new policy group:
   a. Right-click the policy group you want to copy and select **Copy**. For example, right-click the Metrics policy group under WLSSPI and select **Copy**.
   b. Right-click the group under which this policy group is located and select **Paste**. For example, right-click WLSSPI and select **Paste**.
   c. Rename the new group. For example, right-click Copy of Metrics and select **Rename**. Type in a new name.
3. Rename the original policies within the new policy group:
   a. Right-click the policy and select **All Tasks** → **Edit**.
   b. Click **File** → **Save As**.
   c. Enter a new policy name and select **OK**.
4. Delete all original policies within the new policy group:
   a. Select the policies and hit **Delete**.
5. 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.



Using the WLS-SPI Collector/Analyzer Command with Parameters

The `wasspi_wls_perl -S 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 Basic 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	Description
`-c` **Required**	(collector) Specifies the collector policy name. This *must* match the collector policy name in which it appears. **Syntax**: `-c` *<collector_policy_name>* **Example**: `-c WLSSPI-61-05min`
`-e`	(exclude) Allows you to exclude specific servers; may not be used with `-i` option. **Syntax**: `-e` *<server_name>* **Example:** `-e server2,server4`
`-i`	(include) Allows you to list specific servers to monitor. This option may not be used with `-e` option. **Syntax**: `-i` *<server_name>* **Example:** `-i server1,server3`
`-m`	(metric) Specifies the metric numbers or number ranges on which to collect data. **Syntax**: `-m` *<metric_number,metric_number_range>* **Example**: `-m 1,3-5,9-11,15`
`-matchver`	(match version) Specifies the exact WebLogic Server version to monitor. This option may not be used with the -minver nor -maxver options. If no matching versions are found, the command does not execute. **Syntax**: `-matchver` *<version_number>* **Example**: `-matchver 6.1`
`-maxver`	(maximum version) Specifies the highest WebLogic Server version to monitor. Use with -minver to specify a range of versions. If no versions are found, the command does not execute. **Syntax**: `-maxver` *<version_number>* **Example**: `-matchver 7`
`-minver`	(minimum version) Specifies the lowest WebLogic Server version to monitor. Use with -maxver to specify a range of versions. If no matching versions are found, the command does not execute. **Syntax**: `-minver` *<version_number>* **Example**: `-matchver 6.1`
`-r`	(report) Generate an ASCII report for the specified metric(s). **Syntax**: `-r`

Parameter	Description	
-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). **Syntax**: wasspi_wls_ca *<prefix>-<collector _policy>* -m *<metric_number>* -t *<prefix>-* **Example:** wasspi_wls_ca -c WLSSPI-61-05min -m 220-223 -t DEV-	
-x	Allows you to specify a property and value. **Syntax**: -x *<property>=<property_value>*	
	alarm	When off, overrides any alarming condition as set up in the metric policy. **Example**: -x alarm=off
	prefix	Default: JMXUDM_. Specify the prefix of the metric ID. **Example**: -x prefix=SALES_
	print	When on, prints the metric name, instance name, and metric value to STDOUT in addition to any configured alarming or logging. **Example**: -x print=on
	log	When off, prevents graphing or reporting functions. **Example**: -x log=off

Examples

- To collect specific data on all configured servers:

```
wasspi_wls_ca -c WLSSPI-61-05min -m 10-14,25,26
```

- To collect data from specific servers only:

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

- To not collect data from specific servers:

```
wasspi_wls_ca -c STAGE-WLSSPI-61-15min -m 220-225
-e server1,server2
```

Using JMX Actions Command Parameters

The command parameters described in this section are used to run JMX actions. JMX actions are one or more JMX calls (invoke, get, set) performed on an MBean instance or type. A single JMX call can be performed from the command line. Multiple JMX calls can be specified in an XML file or as a Metric subelement in a UDM file.

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

Parameter	Description	
-mbean	Performs a JMX call on the specified MBean(s). This option may not be used with the -m nor -xml options. **Syntax:** -mbean *<objectname>* *<action>* **Example:** -mbean *:*,Type=JMSServerConfig -get MessagesMaximum where *<action>* (a JMX call) is one of the following:	
	-get	Returns the value of the apecified attribute. **Syntax:** -mbean *<objectname>* -get *<attribute>* **Example:** -get MessagesMaximum
	-invoke [-type]	Executes an MBean operation with the specified parameters. An operation may not require parameters (therefore, -type is not specified). A type parameter must be specified for operations which accept parameters. -type supports operation overloading. **Syntax:** -mbean *<objectname>* -invoke *<operation>* [-type *<parameter_type>* *<parameter_value>*]... where *<parameter_type>* is one of the following: short, int, long, double, float, boolean, java.lang.Short, java.lang.Integer, java.lang.Long, java.lang.Double, java.lang.Float, java.lang.Boolean, and java.lang.String. **Example:** -invoke stagingEnabled -type java.lang.String examplesServer
	-set	Assigns the specified value to the specified attribute. **Syntax:** -mbean *<objectname>* -set *<attribute>* *<value>* **Example:** -set MessagesMaximum 250000
-o	(object) Specifies an MBean instance. **Syntax:** -o *<mbean_instance>* **Example:** -o examplesJMSServer	
-xml	Specifies the XML file that contains the JMX action(s) to perform. This option may not be used with the -m nor -mbean options. **Syntax:** -xml *<filename>* **Example:** -xml myJMXActions.xml	

Examples

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

```
wasspi_wls_perl -S wasspi_wls_ca -a
-mbean "PetStore:*,Type=ExecuteQueueConfig"
-set ThreadsMaximum 50 -o <$OPTION(instancename)>
```

- Set the MessagesMaximum attribute to 25000 on multiple MBean instances:

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

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

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

- Invoke an operation on multiple MBean instances:

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

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

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

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

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

- Use the sample actions xml file:

```
wasspi_wls_perl -S wasspi_wls-ca -a
-xml /var/opt/OV/wasspi/wls/conf/wls_JMXActions-sample.xml
-i examplesServer
```

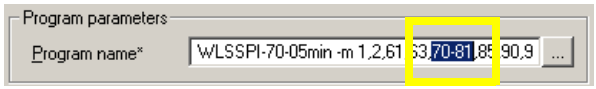
Changing the Collection Interval for All Scheduled Metrics

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 WLSSPI 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** → **WLSSPI**.
2. Select **Monitors**.
3. Right-click the collector policy **WLSSPI-70-05min** and select **All Tasks** → **Edit**.
4. Click **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 **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, rename it with a name reflecting the new interval, delete all but the metrics you are changing, set the new interval, and edit the original policy to remove the changed 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** → **WLSSPI**.
2. Select **Monitors**.
3. Right-click the collector policy **WLSSPI-70-05min** and select **All Tasks** → **Edit**.
4. Click **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 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 **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 **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 the metric to open it for customization (for example, double-click **WLSSPI_0012**). The Measurement Threshold window displays.
2. Select the Threshold levels tab.
3. Click **Specify instance filters**.
4. Select the Condition tab.
5. Enter a Rule description (for example, "Rule for all servers except SERVER_1").
6. Click **OK**.

7 In the Measurement Threshold window, click **Copy** to make a copy of the rule.

8 Double-click 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 the condition.

12 Change the name of the condition to WLSSPI-0012.2.

13 Change the Threshold limit to 20.

14 Click **OK**.

15 Click **OK.**

16 Verify the order of your rules (for example, make "Rule for SERVER_1" the first rule).

17 Click **Matching test** 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 the policy group you want to copy and select **Copy**. For example, right-click the Metrics policy group under WLSSPI and select **Copy**.

   b Right-click the group under which this policy group is located and select **Paste**. For example, right-click WLSSPI and select **Paste**.

   c Right-click `Copy of Metrics` and select **Rename**. Rename the new group according to how you plan to identify the new metric policies. For example, rename the group to CLIENT01Metrics.

2 Rename the original policies within the new policy group. The names 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 1,12,16
-t CLIENT01-
```

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

a Right-click the policy and select **All Tasks** → **Edit**.

b Click **File** → **Save As**.

c Enter a new policy name and select **OK**.

3 Delete all original policies within the new policy group: Select the original policies and press **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 11 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:

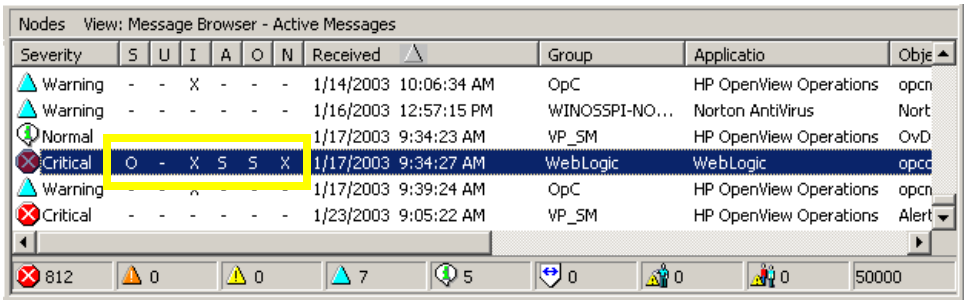
- **To view the Message Properties**. Double-click a message in the OVO message browser. Reports can be viewed from the Annotations tab area, showing data values on a single server.
- **To view reports.** From the OVO console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **Metric Reports**. Double-click the report. A report is generated for all WebLogic Server instances on the selected managed node.
- **To view graphs.** Double-click 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.



How to view the report

To view the report, double-click 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 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_ExQThrdUtilPct

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


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

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

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

Monitoring WebLogic on Unsupported Platforms

The WLS-SPI supports monitoring WebLogic Server-installed systems running on HP-UX, Solaris, Linux, Windows 2000, and AIX. 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 HP-UX, Solaris, Linux, Windows 2000, or AIX, 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 1 pricing).
- The WLS-SPI runs on at least one managed node on a supported platform: HP-UX, Solaris, Linux, Windows 2000, or AIX.
- (Optional, for logfile monitoring) The remote system runs on a platform supported by the OVO agent software.

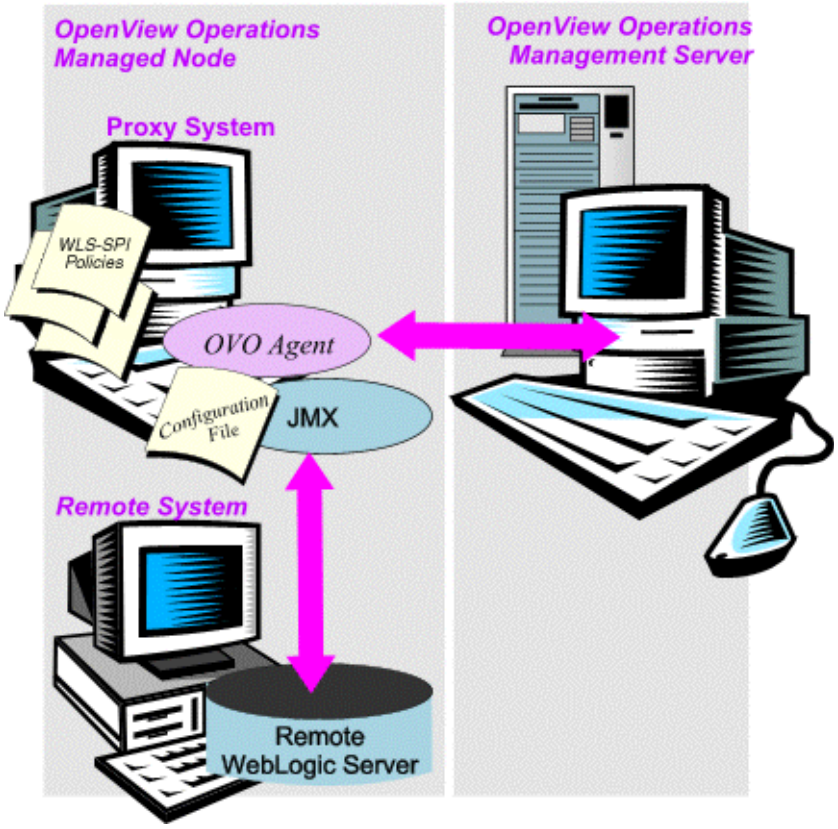
Overview

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. (Refer to Example Configuration on page 47, showing how the remote entry appears (with system IP address).

Policy deployment requirement: Policies must be deployed on the local node.

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, Linux, Windows 2000, or AIX platforms by completing the following tasks.

Task 1: Configure the Remote WebLogic server

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

1 Launch the Configure WLSSPI tool.

2 Select a WebLogic managed node from which to monitor the remote WebLogic Server.

3 In the configuration, include an entry for each remote WebLogic server: 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/wlsserver6.0spi
JAVA_HOME=/opt/bea/jdk130

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

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

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 system on a platform unsupported by WLS-SPI. The remote system is configured similar to that of the local system but contains the 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, and if you use OpenView Performance Manager 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 SPI for WebLogic Server node group.

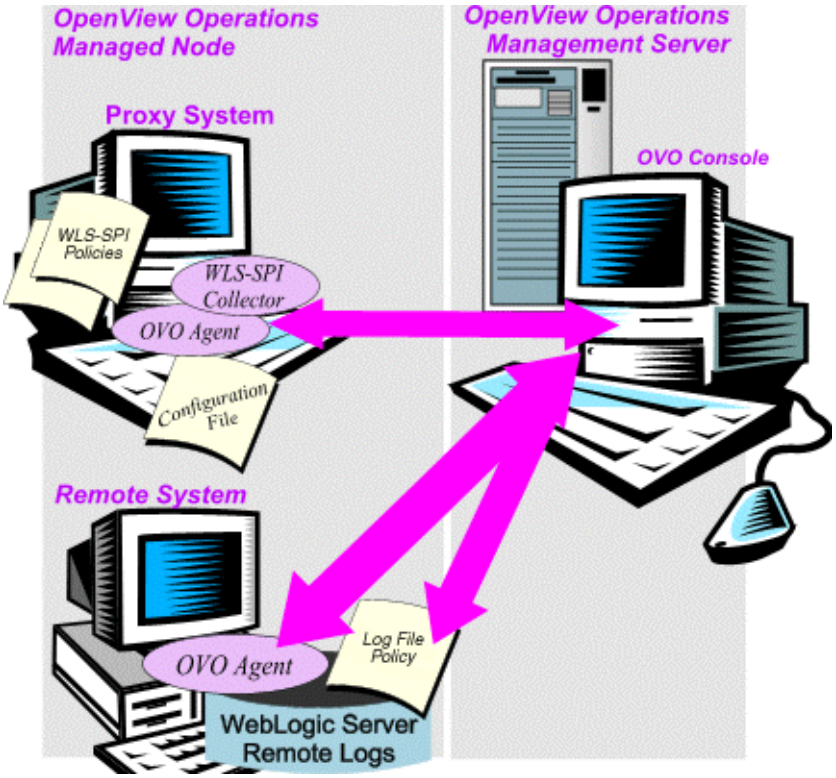
Configuring Remote Monitoring for Logfiles (Optional)

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 re-version 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** → **WLSSPI**.
   b. Select **Logfiles**.
   c. Double-click 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 an OVO agent is not present on the remote system, WebLogic logfiles on the remote system cannot be monitored.
- WLS-SPI tools cannot be executed on remote systems.
- WLS-SPI does not support application servers with the same name.

5 Using HP OpenView Reporting and Graphing Features with WLS-SPI

The Smart Plug-in for BEA WebLogic Server (WLS-SPI) can be integrated with the following HP OpenView reporting and graphing products (these products must be purchased separately):

- HP OpenView Reporter (Reporter)
- HP OpenView Performance Insight (OVPI)
- HP OpenView Performance Manager (OVPM)

HP OpenView Reporter. Reporter produces management-ready, web page reports, showing historical and trending information. This is *not* the version of Reporter that is included with HP OpenView Operations for Windows.

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

Refer to Integrating WLS-SPI with Reporter to Generate Reports on page 51 for more information on how to integrate WLS-SPI with Reporter. After integrating WLS-SPI with Reporter, every night, Reporter generates reports that show the performance and availability of a WebLogic Application Server on configured managed nodes.

HP OpenView Performance Insight. OVPI is a network management system that collects, processes, and reports data. The data is used to generate reports. Refer to the *HP OpenView Performance Insight Administration Guide* for more information about OVPI. Refer to the *Application Server Report Pack User Guide* for more information about WLS-SPI reports and how to integrate WLS-SPI with OVPI.

HP OpenView Performance Manager. OVPM provides graphing capability. This is *not* the version of OVPM that is included with HP OpenView Operations for Windows.

Refer to Integrating WLS-SPI with HP OpenView Performance Manager on page 56 for more information on how to integrate WLS-SPI with OVPM. After integrating WLS-SPI with OVPM, graphs are available the following day.

Using the OpenView Performance Agent

WLS-SPI relies on the default performance subagent (CODA) to collect and store performance data. CODA is automatically deployed on all OVO managed nodes. The performance data collected by CODA is used by integrated OVO for Windows graphing and reporting features (WLS-SPI Metric Reports tools, automatic command reports, and operator-generated graphs), Reporter, OVPI, and OVPM. The integrated OVO for Windows reporting and graphing features only work with data collected by CODA.

However, you may want to use the OpenView Performance Agent 3.x (OVPA or MeasureWare Agent) to collect and store performance data (OVPA is a product that must be purchased separately). The data collected by OVPA is used by Reporter, OVPI, and OVPM. The integrated OVO for Windows reporting and graphing features cannot use data collected by OVPA and therefore these features do not work if you use OVPA.

If you are running OVPA 4.x for Linux, you do not need to configure the WLS-SPI data collector to use OVPA. By default, the WLS-SPI detects and uses this version of OVPA to collect and store performance data. This version of OVPA works with the integrated OVO for Windows reporting and graphing features.

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
HP-UX, Linux, Solaris	`/var/opt/OV/conf/dsi2ddf/`
AIX	`/var/lpp/OV/conf/dsi2ddf/`
Windows	`\Program Files\HP OpenView\Installed Packages\{790C06B4 ...}\conf\dsi2ddf\`

   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

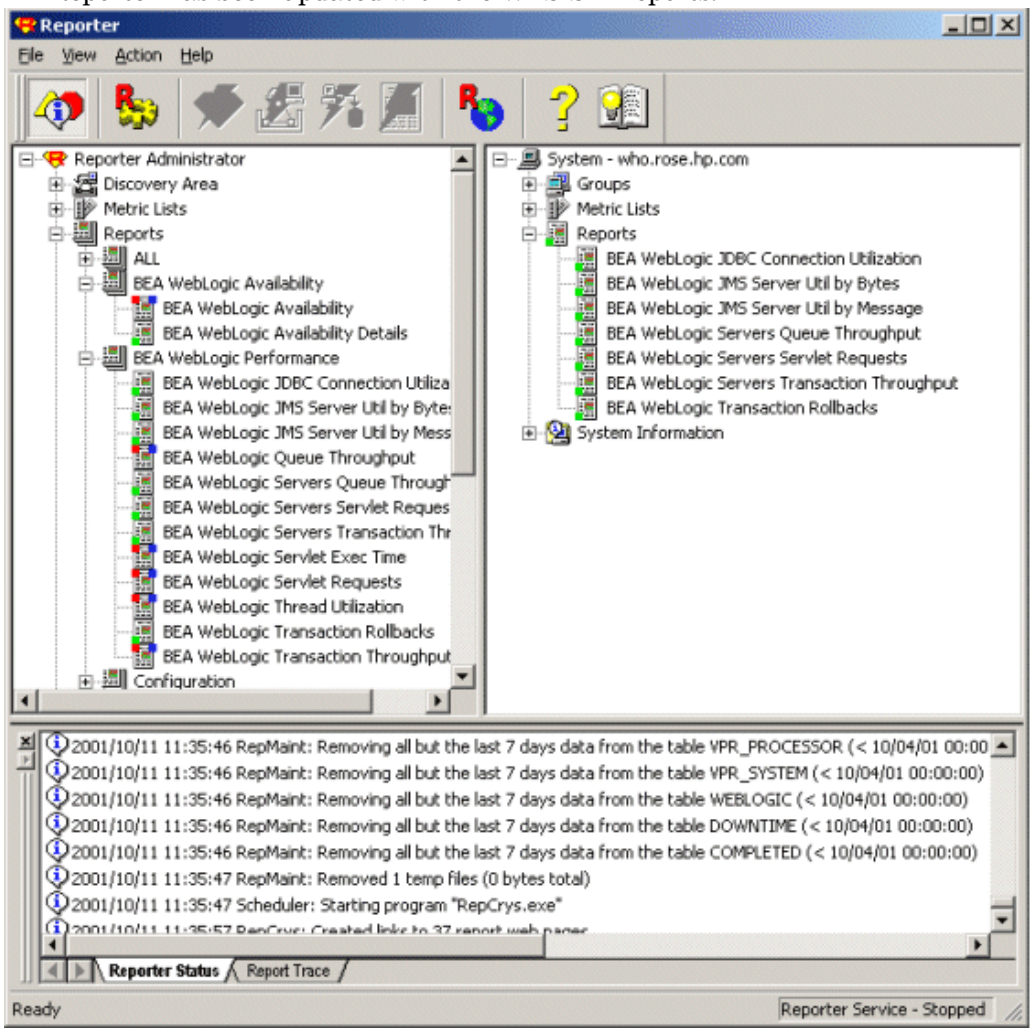
Prerequisite: you must configure the WLS-SPI, which includes software deployment, server connection configuration, and assignment/deployment of policies to targeted nodes.

1. Install the WLS-SPI report package on the Windows system running Reporter:
   a. Insert the HP OpenView Smart Plug-ins CD into the CD drive of the Windows system running Reporter.
   b. Go to the `/WebLogic SPI Reporter Package` directory.
   c. Double-click **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. (Refer to Integrating WLS-SPI with Reporter to Generate Reports on page 51 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
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

Table 1 Performance

Report Title	Description	WebLogic Version	Metric
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 **Add/Remove Programs**.
2. From the Add/Remove Programs window, select **WLSSPI-Reporter**.
3. Click **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 OVPM, do the following:

1. Install the WLS-SPI graph package on the Windows system running Performance Manager:
   a. Insert the Smart Plug-ins and Integration Modules, New and Upgraded, Volume 1 CD into the CD drive of the Windows system running Performance Manager.
   b. Go to the `/WebLogic SPI OVPM Configuration Package` directory.
   c. Double-click **WLSSPI-Grapher.msi**
   d. Complete the installation by following the instructions in the windows that open.
2. To graph any WebLogic server metric, use the data source name: WLSSPI_METRICS

Viewing Graphs that Show Alarm Conditions

For graphing purposes, the WLS-SPI organizes metrics according to type and includes four groupings. When a message is generated for any metric appearing in a table in the section that follows, you can view a chart of its and other metric values.

To view a graph associated with an alarm condition (Operator-initiated action has been defined with the WLS-SPI policy), complete these steps:

1. In the OVO Message Browser double-click the message.
2. In the Message Properties window select the Commands tab.
3. Click **Start** to start the Operator-initiated command.

The resulting action displays the metric's WLS-SPI graph, which charts its values along with the values of other metrics in the same group.

Viewing Graphs that Show Past/Current Conditions

You can generate any of the available graphs manually:

1. From the OVO console, select **Operations Manager** → **Reports & Graphs** → **Graphs** → **SPI for WebLogic Server**.
2. Double-click the graph you want to generate.

WLS-SPI Metrics Available for Add-On Graphs

The following tables show the graphs available for mapping collected metric values. If you are interested in viewing any one of the metrics included in any 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.
Active Socket Count	B014_ActiveSocketCnt	Number of socket connections opened.
JDBC Connect Pool Wait Count	B061_JDBCConPlWtCnt	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.

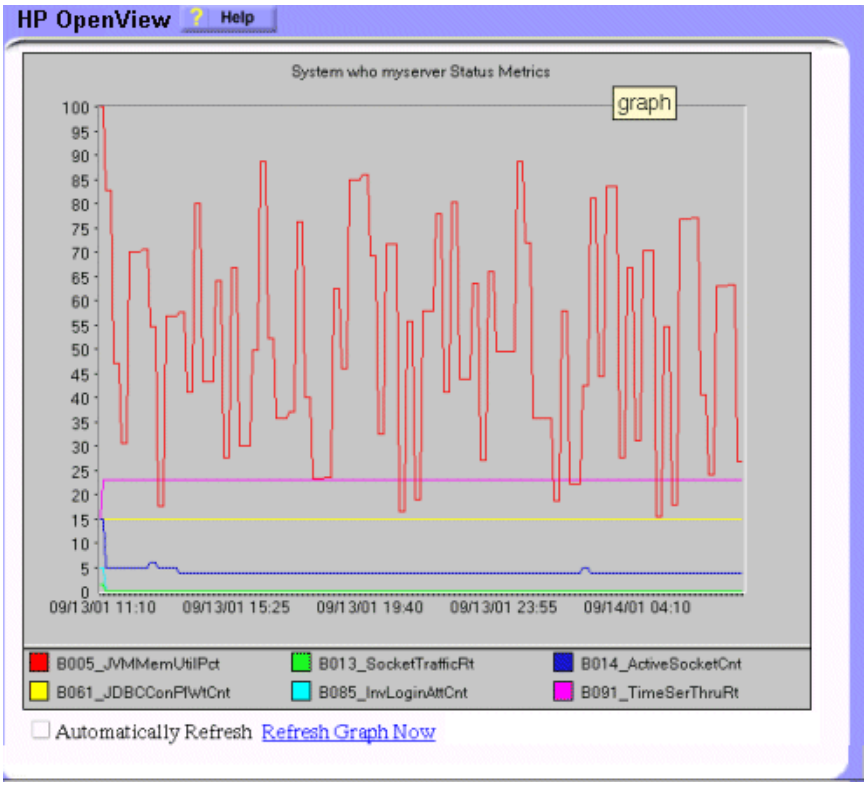
Table 6 Transaction: 70, 71, 72, 73, 74, 75, 76, 77

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

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. Refer to the online help for instructions on changing display settings.

Removing the WLS-SPI Grapher Package

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

6 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.

Refer to 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	`/var/opt/OV/wasspi/wls/conf/`
Windows	*%OvAgentDir%*`\wasspi\wls\conf\`

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. The name can be up to 20 characters in length.
alarm	"yes" "no"	no	"no"	If yes, the metric value is sent to the agent via opcmon.
report	"yes" "no"	no	"no"	If yes, the metric value is logged for reporting.
previous	"yes" "no"	no	"yes"	If yes, the metric value is saved in a history file so that deltas can be calculated. If you are not calculating deltas on a metric, set this to "no" for better performance.
graph	"yes" "no"	no	"no"	If yes, the user-defined metric is graphed.
description	text	no	""	A description of the metric.

Example

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

The MBean Element

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

- **ObjectName** - the JMX-compliant object name of the MBean. The object name can include JMX-compliant pattern matching.
- **Attribute** - the MBean attribute name.
- **AttributeValueMapping** (optional) - numeric values that should be substituted for the values returned by the MBean attribute. This can be used to convert string attributes to numbers so they can be compared to a threshold. Each AttributeValueMapping contains one or more **Map** elements. Each Map element specifies one value to be mapped.
- **AttributeFilter** (optional) - provides basic filtering of MBeans based on MBean attribute values.

- **FromVersion/ToVersion** (optional) - the versions of the WebLogic Server for which the MBean element is valid. See FromVersion and ToVersion Elements on page 65 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.

Mαπ 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, `<FromVersion server="6.0"/>`
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: 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 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 62.

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 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 **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_DEFINITONS_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 the WLS-SPI group you want to use as a starting point, and select **Copy**.

   c Right-click **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 the policy group and select **Rename**.

   b Type in the new name.

3 Edit and rename each policy:

   a Double-click 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 34 for more information.

   c Configure thresholds in the policy, as appropriate. Refer to Advanced Policy Customizations on page 34 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 the policy and select **Delete**.

Task 5: Deploy the policy group

1 Right-click the policy group and select **All Tasks → Deploy on**.

2 Select the nodes on which to deploy the policy group.

3 Click **OK**.

Task 6: 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 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.

7 Troubleshooting WLS-SPI

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.

Using the Self-Healing Info Tool

Preferred troubleshooting method: Troubleshooting information is enhanced with cross-referencing capabilities available through the HP support site: **http://support.openview.hp.com/self_healing.jsp**. To access this information and the search capabilities available on the site, you must first download and configure the Self-Healing Services client software, an added benefit of your HP support contract. A link is then automatically set up to the self-healing services web site, where your SPI problem can be matched to similar SPI problems/solutions for increased troubleshooting effectiveness. Refer to the Self-Healing Services section in the *SPI CD Installation Guide* for information on how to download, configure, and use the software and its automatic link to the HP web site support service.

Alternative troubleshooting method: If self-healing services is not installed/configured on the node, you can use the Self-Healing Info tool. This tool also gathers SPI troubleshooting data and stores it in a file that you submit to HP support for assistance. Refer to the online help for more information about using this tool.

Log and Trace Files

Management Server

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

File Type	Log
Filename	`\<%OvInstallDir%>\install\WASSPI\WLSSPI\English\Discovery\log\<managed_node>_disc_server.log`
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/` or `/var/lpp/OV/`):

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 `<OvAgentDir>/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 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	Log
Directory	/<*OvAgentDir*>/wasspi/wls/log/wasspi_wls_discovery.log
Description	Records output from the WLS-SPI discovery process.

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 Start Tracing 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	Log
Directory	\<*OvAgentDir*> `\wasspi\wls\log\wasspi_wls_discovery.log`
Description	Records output from the WLS-SPI discovery process.

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 Start Tracing tool.

The WLSSPI Discovery Policies

Problem

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

Solution

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

  — If the WLSSPI Discovery policies are not listed, check the message browser for the following messages

  ```
  WASSPI-302: Updating WLS-SPI configuration in 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.

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

Solution (cont.)

- 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.
- 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.
- 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).
- On a Windows managed node, if the `HKEY_LOCAL_MACHINE\\Software\\BEA Systems\\BEAHOMELIST` registry key does not exist, either configure it, create the file *%SystemDrive%*`\BEA\beahomelist`, or configure the BEA_HOME_LIST property for that managed node.
- Verify the Java home directory (see Verifying the Java Home Directory on page 83).
- Verify that the discovery agent is running on the managed node:
  - a Run the command **`opcagt -status`**
  - b Look for the following:

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

    If the agent is not running, start it by running the command **`opcagt -start -id 13`**
- If you are running WebLogic Server 7.0 or higher and did not save the domain configuration file (for example, `config.xml`) in the default directory (<*BEA_Home_Dir*>`/user_projects/`<*WebLogic_Domain_X*>`/`, where <*BEA_Home_Dir*> is the directory that contains the `registry.xml` file), then do one of the following:
  - — Manually set the server using the Configure WLSSPI tool, OR
  - — Manually configure ADMIN_PORTS, the port number(s) of the WebLogic Admin server(s) listed in the domain configuration file, using the Configure WLSSPI tool. The global LOGIN and PASSWORD must be configured for the node on which these WebLogic Admin servers are running.

Solution (cont.)

- On a UNIX managed node, verify that `BEA_HOME_LIST` and `HOME_LIST` directory path names do not include spaces. The discover process currently does not support spaces in directory names.
- If you deployed the Discovery policies at the WLSSPI Discovery level or not in the order shown in Manually Deploying the Discovery Policies on page 83, uninstall and redeploy the Discovery policies:
  - a From the OVO console, select **Operations Manager → Policy management → Policy groups → SPI for WebLogic Server.**
  - b Right-click **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 83. The policies must deployed in order. If you deployed the policies as a group, the policies may not be deployed in the correct order.

Verify that the Configure WLSSPI 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.

Problem

The WLSSPI Discovery policies are adding inaccurate information to the configuration.

Solution

- Verify LOGIN and PASSORD are correct. Refer to Task 3: Collect Application Server Information on page 18 for more information.
- Verify the Java home directory. Refer to Verifying the Java Home Directory on page 83 for more information.

Update the configuration and uncheck the AUTO_DISCOVER check box in the configuration editor to prevent the WLSSPI Discovery policies from overwriting the configuration information.

Problem

Two or more WebLogic domains have managed WebLogic Servers on the same OVO managed node.

Solution

1. From the OVO console, select **Operations Manager → Policy management → Deployment jobs.**
2. Find the jobs in an `Error` state.
3. For each job you want to restart, right-click it and select **All Tasks → Restart job**

Problem The following error message appears:

```
PMD51) Error: Unable to deploy instrumentation files from
directory <directory_name>: (NUL16389E) Unspecified error
(0x80004005). Please check the error log on the managed
node.
```

Solution

1. From the OVO console, select **Operations Manager** → **Policy management** → **Deployment jobs.**
2. Find the jobs in an `Error` state.

For each job you want to restart, right-click it and select **All Tasks** → **Restart job**.

Problem The property of critical error messages in the OVO console is:

Errors occurred during the distribution of the monitors. Solve the problems and distribute the monitors again. (OpC30-1030.

Solution

1. From the OVO console, select **Operations Manager** → **Policy management** → **Deployment jobs.**
2. Find the jobs in an `Error` state.
3. Right-click each job you want to restart and select **All Tasks** → **Restart job**.

Manually Deploying the Discovery Policies

If the WLSSPI Discovery policies do not deploy successfully when you run the Discover WebLogic 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 **WLSSPI-Messages** and select **All Tasks** → **Deploy on**.
3. Select the nodes on which to deploy the auto-discovery policies.
4. Click **OK**.
5. Right-click **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 79 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 Configure WLSSPI tool:

a. From the OVO console, select **Operations Manager** → **Tools** → **SPI for WebLogic Server** → **SPI Admin**.

b. Double-click **Configure WLSSPI**.

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 **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 LOGIN and PASSWORD 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 `AUTO_DISCOVER` check box in the configuration editor to prevent the WLSSPI Discovery policies from overwriting the configuration information.

Problem The `Server status is unknown (down)` message appears in the message browser but the server is running

Solution Check that you have correctly set the PORT, PROTOCOL, and/or PASSPHRASE properties:

- Verify that PROTOCOL is set to one of two values: t3 (for non-SSL) or t3s (for SSL).
- If the application server is using SSL, verify that the PORT is set to a valid SSL port number and that PROTOCOL is set to t3s.
- If the application server is not using SSL, verify that the PORT is set to a valid non-SSL port number and that PROTOCOL is set to t3.
- If the keystore has a password defined, re-set the PASSPHRASE in case it has been mistyped.

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 the node and select **Properties**.
3. Select the **Network** tab.

Tools

Message

`Configuration variable SERVER<n>_START_CMD missing for server "Default Server"`

Solution

To successfully run the Start WebLogic tool, you must set the START_CMD property. Set this property using the Configure WLSSPI tool. Refer to the online help for more information about this tool.

Message

`Configuration variable SERVER<n>_STOP_CMD missing for server "Default Server"`

Solution

To successfully run the Stop WLSSPI tool, you must set the STOP_CMD property. Set this property using the Configure WLSSPI tool. Refer to the online help for more information about this tool.

Problem

Check WebLogic tool showing a wrong status for a server instance.

Solution

If a server is up and running but Check WebLogic tool returns the server status as `NOT_RUNNING`, then turn ON the monitoring for that particular server by following these steps:

1. Select **Tools** → **SPI for Weblogic Server** → **SPI Admin** → **Start Monitoring**.
2. Select the server instance for which the status is shown as `NOT_RUNNING`. Ensure that Monitoring is ON.
3. Relaunch the Check WebLogic tool and verify the server status.

Problem

When launching the tools, the tools hang or there is no output.

Solution

The tools will not work if the memory is low. Check the performance of the node and the management server. The physical memory available must be more than 500 MB.

Problem

Verify tool lists files and directories related to the management server as missing. For example:

```
/MGMT_SERVER/SPI-Share/wasspi/wls/bin/parseDefs.pl
/MGMT_SERVER/SPI-Share/wasspi/wls/bin/
processWASSPIDiscovMsg.pl
```

```
/MGMT_SERVER/SPI-Share/wasspi/wls/conf
```

Solution

This is a known problem. The verify tool lists management server related files if you install the WebLogic Server on the management server itself. This problem occurs if both the managed node and the management server are the same.

Problem	Check WebLogic tool does not give any output.
Solution	Ensure that the Collector is running for the WebLogic Server instance on that node.

Index

R

S

T

U

W