HP OpenView Service Oriented Architecture Manager

WSM J2EE Agent Administrator Guide

Version: 2.1

Windows, HP-UX, Linux

June 2006

Legal Notices

Warranty

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

The information contained herein is subject to change without notice.

Restricted Rights Legend

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

Copyright Notices

© Copyright 2004-2006 Hewlett-Packard Development Company, L.P.

Trademark Notices

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

Linux is a U.S. registered trademark of Linus Torvalds

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

UNIX® is a registered trademark of The Open Group

Support

You can visit the HP OpenView 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

Table of Contents

1

Introduction

This chapter contains overview information about the WSM J2EE Agent Administrator Guide and the WSM J2EE Agent. In particular, this chapter includes the following sections:

- Document Overview
- Contextual Overview
- Common Handlers

Document Overview

The WSM J2EE Agent Administrator Guide provides instructions for deploying and using the WSM J2EE Agent. The Agent is currently offered for the WebLogic Server (WLS) platform. The instructions in this guide are specific to WLS. The guide also provides a brief contextual overview and instructions for configuring the Agent's components and runtime behaviors.

The guide does not provide instructions for setting up an end-to-end management solution. The *HP OpenView Service Oriented Architecture (SOA) Manager Administrator's Guide*, together with this guide, should be used when implementing a complete Web services management solution.

Audience

The guide is primarily intended for Systems Integrators and System Administrators who are responsible for integrating and enabling management components in their application environments.

Prerequisites

Users must have fundamental knowledge of the Java programming language and Java platform technologies. Users should also have fundamental knowledge of Web services principals and be familiar with the WebLogic Server platform.

Contextual Overview

The WSM J2EE Agent is responsible for collecting and exposing management data for Web services that are deployed on Java-based Web services platforms. The Agent runs in the same Java Virtual Machine (JVM) process as the Web services platform. In the case of the WSM J2EE Agent for WLS, the Web service platform is WebLogic's Web services platform.

> If you want to use your current Java-based Web services platform (not WebLogic), you can use a WSM Broker to manage Web services instead of the WSM J2EE Agent. For instructions on using the WSM Broker, see the *WSM Broker Administrator Guide*, located in the `/Documentation` directory of the SOA Manager CD.

The J2EE Agent provides a JAX-RPC handler that mediates the SOAP communication between a client and a Web service. The JAX-RPC handler is configured with sub-handlers (referred to as common handlers) that provide varying levels of manageability (Monitoring, Logging/Auditing, etc…). A JAX-RPC handler chain is automatically created for each Web service that the J2EE Agent discovers in the Web services platform.

Another core component of the Agent is the SBA. The SBA contains Managed Objects (MOs) that represent the data and metrics collected by the handlers. The SBA is also responsible for exposing MOs as Web services that comply with the Web Services Management Framework (WSMF) specification. WSMF is an HP-authored precursor version of standard WS-based management protocols. Lastly, SOA Manager uses the management Web services to retrieve management data and metrics and display them in the Business Services Explorer (BSE).

Figure 1-1 provides a high-level overview of the architecture and context of the WSM J2EE Agent.
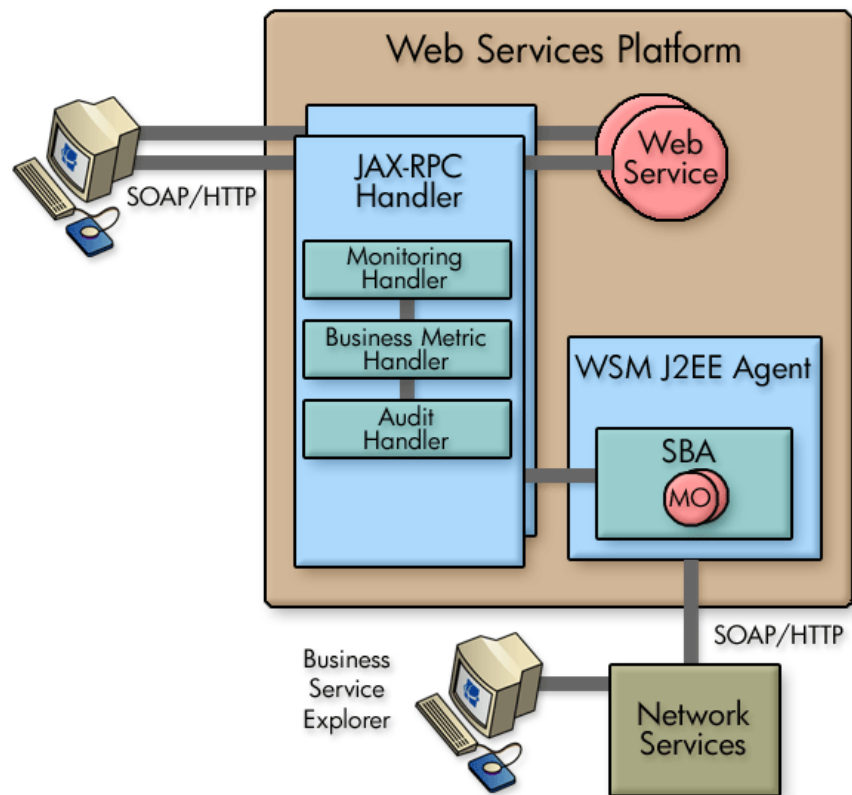
Figure 1-1: Contextual Overview

Common Handlers

As shown in Figure 1-1, a JAX-RPC handler can contain any number of sub-handlers known as common handlers. Together, the handlers are considered a handler chain. Three handlers are included with the WSM J2EE Agent for WLS and are described below. For more information on configuring the common handlers, see Chapter 3.

Monitoring Handler

The Monitoring Handler collects performance data for a Web service. In particular, the handler reports:

- Average Response Time
- Maximum Response Time
- Minimum Response Time
- Total Message Count
- Failed Message Count
- Success Message Count

Audit Handler

The Audit Handler provides message trace capabilities for a Web service. In particular, the handler captures all SOAP requests, responses, and failures. Trace messages are sent to the Network Services server database and viewed using the BSE.

Business Metric Handler

The Business Metric Handler captures specific data that is found in a SOAP request, response, and failure message. The data is defined using an XPath expression and is applied to individual Web services. When the data is found in a SOAP message, the data is sent to the Network Services server and a business content alert is displayed in the BSE.

Deploy the WSM J2EE Agent for WebLogic

This chapter provides detailed instructions for installing/deploying the WSM J2EE Agent for WebLogic Server (WLS). The Agent supports WLS scenarios that use standalone managed servers as well as managed servers that are part of a server cluster. The WebLogic instructions in this chapter are specific to deploying the Agent and do not include detailed WLS instructions. For detailed WLS instructions, refer to the WLS documentation.

The WSM J2EE Agent software for WLS is located on the SOA Manager installation CD. Make sure you have the CD before beginning the instructions in this chapter.

> The WSM J2EE Agent is licensed. During the installation, an evaluation license is installed, and the Agent may be evaluated for 60 days. Thereafter, a license must be purchased. Unlicensed components will fail to start and a license expiration message is written to the log file. For information on licensing, see the "Entering a SOA Manager License" section in the *SOA Manager Installation Guide* after completing the installation.

Software Requirements

The following software is needed when using the WSM J2EE Agent for WebLogic:

- WebLogic Server 8.1.x – A free developer edition of the WebLogic application server can be downloaded from the BEA Web site, http://www.bea.com. The Agent supports Windows, UNIX, and Linux installations.

  > The WSM Agent has been tested against WebLogic Server 8.1 service pack 2 (8.1.2.0).

- Ant 1.5.x – Ant is a Java-based build tool available from the Apache open source organization. You can download Ant from http://ant.apache.org/.

Pre-Deployment Steps

You must install ANT and the WebLogic Server software before you can deploy the Agent. General setup requirements are provided in this section.

ANT 1.5.x Setup

ANT is used by the agent's setup script and does not require that you have any practical knowledge of ANT. ANT should be installed to a directory with no spaces in the path.

Create an environment variable called ANT_HOME and set it to the directory where you installed ANT.

WebLogic Server 8.1.x Setup

WLS should be installed to a directory with no spaces in the path. Create an environment variable called BEA_HOME and set it to the directory where you installed BEA.

In addition, the WSM J2EE Agent must be deployed to a WLS domain. If you do not currently have a WLS domain set up, use the Configuration Wizard that comes with WLS to create a basic WebLogic Server domain. See the WLS documentation for instructions on using the Configuration Wizard to create a WLS Domain.

> ➤ It is recommended that a WLS Domain be created specifically for testing the WSM J2EE Agent.

Deploy to the Administration Server

The WSM J2EE Agent must be deployed to the WLS Administration Server. Each WLS domain must have one server that acts as the Administration Server. In a typical production environment, the Administration Server is where the Administration Console is run and used to perform administrative tasks. By default, the Administration Server is called `myserver`.

> ➤ Make sure you are logged in to the system as a `root`/`admin` user.

A typical production environment may also have any number of managed servers, which are instances of WebLogic servers used to host enterprise applications. These managed servers can be standalone or part of a cluster. The WSM J2EE Agent must be deployed to the Administration Server before you can deploy additional agents to standalone managed servers or a cluster of managed servers.

Figure 2-1 shows an example WLS environment that includes an administrator server, a standalone managed server, and managed servers in a cluster.
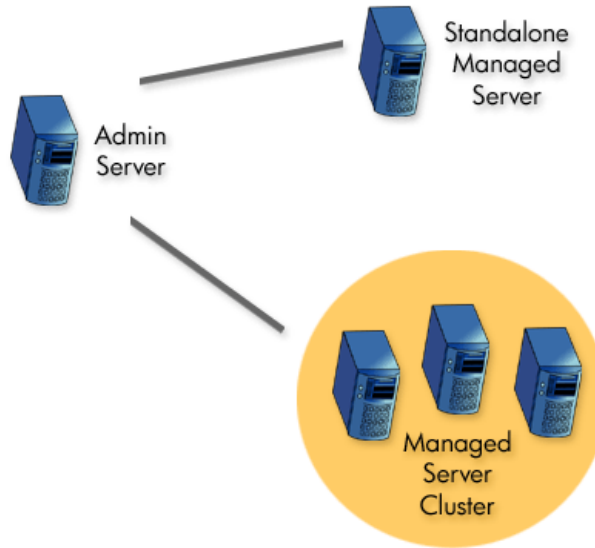
Figure 2-1: Example WLS Production Environment

▶ If your current environment contains standalone managed servers or a cluster of managed servers, see the "Deploying to Managed Servers" section after completing the instructions in this section.

Extract the WSM J2EE Agent for WLS

Log in to the Administration Server computer as a root/admin user. From the SOA Manager CD, extract `WSM_Agents/Java/WebLogic/wsm-agent-wls8.zip` to a location on the system (i.e., `C:\wsm-agent-wls8 or /opt/wsm-agent-wls8`). This location will be referred to as *<install_dir>*.

Modify the Environment Setup Script

The WSM J2EE Agent for WLS includes a script that is used to setup its environment. A script is provided for both the Windows and UNIX platforms. As part of the script, a variable called WSM_AGENT_HOME is set. You must modify this variable to point to *<install_dir>*.

To edit the environment setup script:

1 From the *<install_dir>*`\bin` directory, open the environment setup script for your platform (`setLocalAgentEnv.bat` or `setLocalAgentEnv.sh`).

2 Change the value of the WSM_AGENT_HOME variable to point to *<install_dir>*. For example:

`WSM_AGENT_HOME=c:\wsm-agent-wls8`

Or,

`WSM_AGENT_HOME=/opt/wsm-agent-wls8`

3   Save and close the file.

Modify the Agent-Setup Script

The agent-setup script is used to configure the Agent based on your WLS environment. A script is provided for both the Windows and UNIX platforms.

To edit the agent-setup script:

1   From the *<install_dir>*\bin directory, open the agent-setup script for your platform (`agent-setup.bat` or `agent-setup.sh`).

2   Modify the following variables:

   — **WL_DOMAIN –** The name of the WebLogic domain where the Agent will be deployed.

   — **WL_USER –** The WebLogic administrator user name.

   — **WL_ADMIN_SERVER_NAME** - The name of the WebLogic Administration Server.

   — **AGENT_HOST** – The fully qualified DNS address of the host on which the Agent will run. This value is the same as the host where your WebLogic instance is installed.

   — **AGENT_PORT** – The port number where the Agent will listen for SOAP requests. This value is the same as the port number being used by the Administration Server.

   — **AGENT_PROTOCOL** – The protocol used by the Agent within the WLS environment. This value can be either HTTP or HTTPS. If you select HTTPS, see the "Configuring a Trust Keystore" in Chapter 3 before completing the deployment.

   — **OV_LOG_DIR –** The name of the OV log directory where the agent log file will be stored.

3   Save and close the agent-setup script.

Run the Agent-Setup Script

The agent-setup configures the Agent based on your WLS environment and automatically deploys two Web applications called `wsm` and `wsmf` to the Administration Server instance. The applications are automatically deployed the next time you start the Administration Server.

To run the agent-setup script:

1   Stop WLS if it is currently started.

2   From a command prompt, change directories to *<install_dir>*/bin.

3   Run the agent-setup script that is appropriate for your platform (either `agent-setup.bat` or `agent-setup.sh`).

Modify the WebLogic Startup Script

The WSM J2EE Agent for WLS runs in the same process as WebLogic. Therefore, you must modify your domain's startWebLogic startup script to include the Agent's dependencies.

To modify the WebLogic startup script:

1   From your WLS domain (i.e., *<bea_home>*`\user_projects\domains\mydomain`), open either start `WebLogic.cmd` or `startWebLogic.sh` depending on your platform.

2   Just before the comment near the end of the startWebLogic script which reads "Call WebLogic Server", add a call to the *<install_dir>*`/bin/setLocalAgentEnv` script that is appropriate for your platform. For example:

    ```
    call c:\wsm-agent-wls8\bin\setLocalAgentEnv.bat
    ```

    Or,

    ```
    . /opt/wsm-agent-wls8/bin/setLocalAgentEnv.sh
    ```

3   In the line %JAVA_HOME%\bin\java %JAVA_VM%... weblogic.Server, add `-Djava.util.logging.config.class=com.hp.ov.xpl.log.OvLogConfig` right before `weblogic.Server`. For example:

    ```
    %JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS%
    -Dweblogic.Name=%SERVER_NAME%
    -Dweblogic.ProductionModeEnabled=%PRODUCTION_MODE%
    -Djava.security.policy="%WL_HOME%\server\lib\weblogic.policy"
    ```
    -Djava.util.logging.config.class=com.hp.ov.xpl.log.OvLogConfig
    ```
    weblogic.Server
    ```

4   Save and close the startup script.

Modify the WebLogic Policy File

The WSM J2EE Agent for WLS requires specific permissions in order to run within WLS. The permissions are set in the WLS weblogic.policy file.

To modify the WLS policy file:

1   Open the *<bea_home>*`\weblogic81\server\lib\weblogic.policy` file.

2   Open *<install_dir>*`\config\weblogic_policy_include.txt`.

3   Copy the content of weblogic_policy_include.txt and append it to the end of `weblogic.policy`.

4   Save `weblogic.policy` and close both files.

Start the WebLogic Administration Server

To start the WLS Administration Server for a domain:

1   From the command line, change directories to your WLS domain (i.e., `bea\user_projects\domains\mydomain`).

2   Depending on your platform, run either startWebLogic.cmd or startWebLogic.sh. The startup is complete when the following log message is received in the console window:

```
Smart Business Agent Controller has completed initial discovery;
it is now fully initialized and started!
```

Verify the Deployment

There is a quick method to verify that the WSM J2EE Agent for WLS is successfully deployed to the Administration Server and that it is working properly.

To verify the deployment:

1   Open a browser.

2   Enter the address where the `wsmf` Web application is deployed. For example:

```
http://myhost.com:7001/wsmf
```

(Change the host and port to match you deployment.)

3   From the WSMF SBA WebApp screen, click the **List Deployed Services** link.

You should have at least two management Web services on the list: `WLS` and `WLS.<administration_server_name>`. Additional management Web services may be listed if the domain contains: standalone managed servers, managed servers in a cluster, and Web services that are deployed in the Administration Server.

▶   If your domain contained standalone managed servers and/or a cluster of managed servers. You must deploy a WSM J2EE Agent to each managed server before the management Web services become operational and Web services in those servers are managed. See the "Deploy to Managed Servers" section below.

4   Click the WSDL link for the `WLS` managed service. The WSDL displays as shown below:

The `WLS` and `WLS.<administration_server_name>` managed services represent a set of basic operations and attributes for the WLS environment. The Agent created these services based on management information being exposed through WebLogic's JMX MBeans.

Verify the Automatic Discovery of Web Services

You can verify that the J2EE Agent is automatically discovering Web services by deploying a Web Service and verifying that a management service has been created for it. If you do not have a Web service to deploy, a simple Web service (HelloWorldService) is included in the *<install_dir>*\sample directory. The service is packaged as a Web application.

To verify the automatic discovery of Web Services:

1   Deploy *<install_dir>*\sample\HelloWorldService to the Administration Server using the WLS Administration Console.

2   Open a browser.

3   Enter the address where the `wsmf` Web application is deployed. For example:

    http://myhost.com:7001/wsmf

    (Change the host and port to match you deployment.)

4   Click the **List Deployed Services** link. The management service for the HelloWorldService is listed.

> The Agent polls The WLS environment every 2 minutes to discover any changes that have occurred in the WLS servers. It may take up to 2 minutes for the management service for the HelloWorldService to be listed. You can change the polling interval using the Agent's SBA configuration file. See "Changing the SBA Polling Frequency" in chapter 3.

5   Click on the WSDL link for the *<admin_server_name>*`.HelloWorldService.HelloWorldService` service. The management WSDL for this service displays as shown below:

> To actually see the runtime data and metrics being collected for the HelloWorldService by the HelloWorldService management service, you must use the SOA Manager's Business Service Explorer. See the *SOA Manager Administration Guide*.

Deploy to Managed Servers

Beyond the Administration Server, a typical WLS production environment may have any number of managed servers, which are instances of WLS used to host enterprise applications. These managed servers can be standalone or part of a cluster. In addition, managed servers can be local to the Administration Server process or located on remote computers. To support such scenarios, the WSM J2EE Agent for WLS can be deployed to any managed server.

Figure 2-2 shows an example WLS environment that includes an administrator server, a standalone managed server, and managed servers in a cluster.
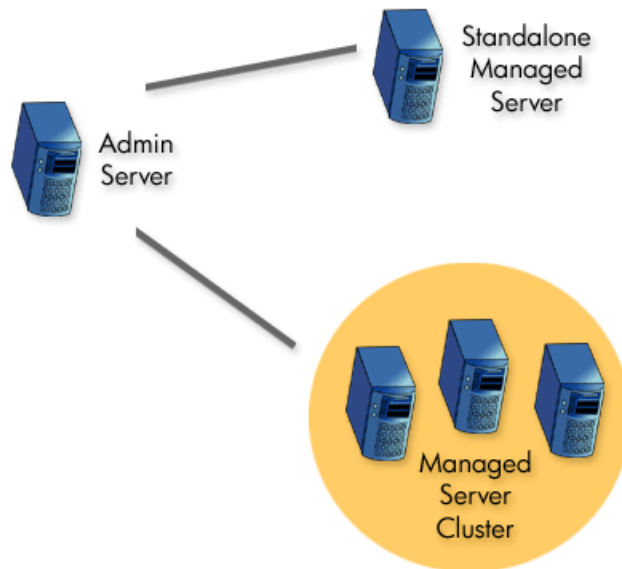
Figure 2-2: Example WLS Production Environment

The WSM J2EE Agent for WLS must be deployed to an Administration Server before it can be deployed to a managed server. See the "Deploy to Administration Server" section before completing the instructions in this section. In addition, when deploying the Agent to a remote computer, access to the computer is required in order to complete the instructions in this section.

This section assumes knowledge of setting up and configuring managed servers and clusters. To complete the instructions in this section, you must have at least one managed server in addition to the Administration Server. The managed server can be standalone and /or part of a cluster. To simplify testing, setup the managed server on the same computer as the Administration Server. See the BEA WLS documentation if you do not know how to create and configure servers and/or clusters.

Extract the WSM J2EE Agent for WLS

From the SOA Manager CD, extract `WSM_Agents/Java/WebLogic/wls8-wsme-managed-server.zip` to a location on the managed server computer (i.e., `C:\wsm-agent-wls8 or /opt/wsm-agent-wls8`). This location will be referred to as *<managed_install_dir>*. The Agent installed on a managed server is a "lite" version of the Agent.

Modify the Environment Setup Script

The WSM J2EE Agent for WLS includes a script that is used to setup its environment. A script is provided for both the Windows and UNIX platforms. As part of the script, a variable called WSM_AGENT_HOME is set. You must modify this variable to point to *<managed_install_dir>*.

To edit the environment setup script:

1   From the *<managed_install_dir>*\bin directory, open the environment setup script for your platform (`setLocalAgentEnv.bat` or `setLocalAgentEnv.sh`).

2   Change the value of the WSM_AGENT_HOME variable to point to *<managed_install_dir>*. For example:

    ```
    WSM_AGENT_HOME=c:\wsm-agent-wls8
    ```

    Or,

    ```
    WSM_AGENT_HOME=/opt/wsm-agent-wls8
    ```

3   In each line with %JAVA_HOME%\bin\java %JAVA_VM%... `weblogic.NodeManager`, add `-Djava.util.logging.config.class=com.hp.ov.xpl.log.OvLogConfig` right before `weblogic.NodeManager`.

4   Save and close the file.

Modify the BEA Node Manager

The BEA Node Manager is a Java utility that runs as a separate process from WLS on each managed server. An Administration Server uses the Node Manager to perform common operations and tasks on managed servers (i.e., starting and stopping). The WSM J2EE Agent for WLS leverages the Node Manager's capabilities. To achieve this integration, the Node Manager's classpath must be modified on each managed server host to include the Agent's dependencies.

> ► The use of the Node Manager is required to use the Agent on a managed server.

The Node Manager's classpath can be modified two different ways depending on how the Node Manager is running (either a standalone process or as a Windows service). Instructions for both scenarios are provided.

Modify the Node Manager Windows Service

To modify the Node Manager when using it as a Windows service:

1   Stop the managed server if it is currently started.

2   Stop and uninstall the Node Manager Windows service if it is currently started.

3   Open *<bea_home>*\weblogic81\server\bin\installNodeMgrSvc.cmd.

4   Just after the comment near the end of the installNodeMgrSvc.cmd script which reads "rem *** Set Command Line for service to execute within created JVM" add a call to the *<managed_install_dir>*/bin/setLocalAgentEnv.bat. For example:

```
call c:\wsm-agent-wls8\bin\setLocalAgentEnv.bat
```

5   Save and close the installNodeMgrSvc.cmd script.

6   Run the installNodeMgrSvc.cmd script.

7   Start the service using the Windows Services Administrative tool.

Modify the Node Manager Standalone Process

To modify the Node Manager when using it as a standalone process:

1   Shut down the Node Manager process if it is currently started.

2   Open the *<bea_home>*\weblogic81\server\bin\StartNodeManager script that is appropriate for your platform (startNodeManager.cmd or startNodeManager.sh).

3   On Windows, just after the set JAVA_VM= entry, add a call to set the Java vendor to Sun. For example:

```
set JAVA_VENDOR=Sun
```

4   Just after the set CLASSPATH entry, add a call to the *<managed_install_dir>*/bin/setLocalAgentEnv script that is appropriate for your platform. For example:

```
call c:\wsm-agent-wls8\bin\setLocalAgentEnv.bat
```

Or,

```
. /opt/wsm-agent-wls8/bin/setLocalAgentEnv.sh
```

5   Save and close the script.

6   Run the startNodeManager script to start the Node Manager.

Start the Managed Server

Managed servers are started through the Administration Server console. When a managed server is started, the Agent is initialized and a managed service is created for each Web service that is running in the managed server.

To start a managed server:

1   Log in to the WLS Administration Console.

2   From the Left frame, select the managed server that you want to start.

3   From the Right frame, click the **Control** tab.

4   From the Start/Stop tab, select **Start this server…**. The server is started. It may take several seconds for the managed server to become operational.

5   Repeat this procedure to start additional managed servers.

Verify the Agent Deployment to a Managed Server

There is a quick method to verify that the WSM J2EE Agent for WLS is successfully deployed to a managed server and that the Agent created a managed service for it. If a cluster is configured, a managed service is also created for the cluster and the managed servers in the cluster.

To verify the Agent deployment to a managed server:

1   Open a browser.

2   Enter the address where the `wsmf` Web application is deployed on the Administration Server. For example:

    http://myhost.com:7001/wsmf

(Change the host and port to match you deployment.)

3   From the WSMF SBA WebApp screen, click the **List Deployed Services** link.

The services list displays the management services that were created based on the managed servers and/or clusters that were discovered in your environment. The following screen shot displays the management services created for an environment that contains:

—   1 cluster: `WLS.MyCluster`

—   1 managed server in the cluster: `WLS.MyCluster.MyServer-2`

—   1 standalone managed server: `WLS.MyServer-1`

—   1 Administration Server: `WLS.myserver`
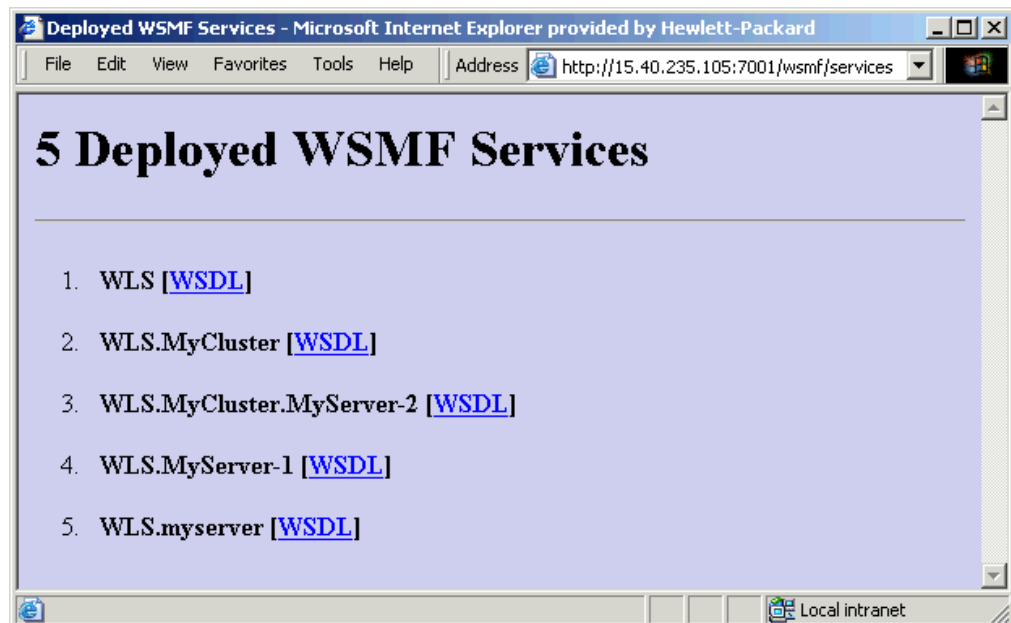
The management services are organized in a `contains` relationship. That is, `WLS` is the root management service. All management services are contained by the root. If a cluster is used, the managed server is contained in the cluster which is contained in the root. This hierarchy mirrors the WLS hierarchy. Any Web services that are being discovered are also organized according to this convention. See "Verify the Automatic Discovery of Web Services" section below.

4   Click on a management service's WSDL link to see its WSDL.

Verify the Automatic Discovery of Web Services

You can verify that the J2EE Agent is automatically discovering Web services on a managed server by deploying a Web Service to each managed server and verifying that a management service has been created for it. If you do not have a Web service to deploy, a simple Web service (HelloWorldService) is included in the *<install_dir>*`\sample` directory. The service is packaged as a Web application.

To verify the automatic discovery of Web Services:

1   Start the WLS Administration Console on the Administration Server.

2   Deploy *<install_dir>*`\sample\HelloWorldService` to each managed server.

3   Open a browser.

4   Enter the address where the `wsmf` Web application is deployed. For example:

    `http://myhost.com:7001/wsmf`

    (Change the host and port to match you deployment.)

5   Click the **List Deployed Services** link. The management services for the HelloWorldService are listed based on your environment. Notice in the following example that each management server where the application is deployed has its own HelloWorldService management service.

The Agent polls The WLS environment every 2 minutes to discover any changes that have occurred in the WLS servers. It may take up to 2 minutes for the management service for the HelloWorldService to be listed. You can change the polling interval using the Agent's SBA configuration file. See "Changing the SBA Polling Frequency" in chapter 3.

6   Click on a HelloWorldService management service WSDL link to see its WSDL.

Un-Deploying on the Administration Server

The following steps are used to un-deploy the WSM J2EE Agent for WLS from the Administration Server.

To un-deploy the Agent from the WLS Administration Server:

1   Open a browser.

2   Go to the Agent's wsm/handlerRemoval.jsp. For example:

    `http://myhost.com:7001/wsm/handlerRemoval.jsp`

    (Change the host and port to match you deployment.)

3   On the Handler Removal Form, select the remove check box to remove the WSM Agent handlers from the Web service.

4   Click the **Remove Handlers** button. The Processing screen displays and verifies the removal of the WSM handlers for the selected Web services.

5   Close the browser.

6   Open a command prompt and change directories to *<install_dir>*\bin.

7   Run the appropriate agent-setup script (.bat, .sh) for your platform and specify the "undeploy" parameter. For example,

```
agent-setup undeploy-from-wls
```

8   Close the command window.

9   Remove the JLA-removed.proprties file from your home directory (i.e., C:\Documents and Settings\Administrator or /usr/admin).

10 Open the *<bea_home>*\weblogic81\server\lib\weblogic.policy file and remove the policies for the WSM Agent.

11 Save and close the file.

12 Remove the call to configure the local agent environment that was added to the startWebLogic script. For example remove:

```
call c:\wsm-agent-wls8\bin\setLocalAgentEnv.bat
```

  Or,

```
. /opt/wsm-agent-wls8/bin/setLocalAgentEnv.sh
```

13 Save and close the startup script.

14 Stop and restart WLS.

15 Delete *<install_dir>*.

Un-Deploying on Managed Servers

The following steps are used to un-deploy the WSM J2EE Agent for WLS on any managed servers that are standalone or part of a cluster.

To un-deploy the Agent from a managed server:

1   On the managed server, stop the BEA Node Manager if it is currently started.

2   Remove the call to setLocalAgentEnv from the *<bea_home>*\weblogic81\server\bin\StartNodeManager script that is appropriate for your platform (startNodeManager.cmd or startNodeManager.sh).

  Or if using the Node as a Windows service,

  Remove the call to setLocalAgentEnv from *<bea_home>*\weblogic81\server\bin\installNodeMgrSvc.cmd. You must then uninstall and reinstall the node manager.

3   Delete *<managed_install_dir>*.

4   Repeat this procedure for each managed server where the Agent is installed.

Deploying the MOM Agent for WLS

The SOA Manager's MOM Agent for WLS is an optional management agent that provides monitoring capabilities for Web services that utilize SOAP/JMS. In particular, the agent provides monitoring capabilities to detect when messages get stuck in the messaging infrastructure and are not delivered to the end destination.

> The MOM Agent is not required when running the WSM J2EE Agent for WebLogic. The MOM Agent is an optional standalone application for JMS support. In addition, a tutorial demonstrating JMS management is provided in the `/Documentation` directory of the distribution. Refer to the instructions included with the tutorial.

The MOM Agent communicates management information about WLS JMS servers, topics, and queues to the Network Services server. The steps in this section assume that a WLS JMS server has been created and that the server contains topics and/or queues. If you are unfamiliar with JMS, refer to the WLS documentation for detailed instructions.

To deploy the MOM Agent:

1   From the SOA Manager CD, copy `/MOM_Agents/Java/Weblogic/MOMAgent.ear` to a location on the WLS computer. This location is referred to as *<install_dir>*.

2   Connect to the BEA WLS Administrator Console.

> The WLS user name and password must be `weblogic`/`weblogic`.

3   In the left frame, click **Applications**. The Applications screen displays in the right frame.

4   Click **Deploy a new Application...**. The Deploy an Application screen displays.

5   In the right frame, use the Location: URL to navigate to the *<install_dir>*/`MOMAgent.ear` file.

6   Select **MOMAgent.ear**.

7   Click **Continue**. The Deploy an Application screen redisplays and allows you to review your deployment selection.

8   Click **Deploy**. The MOM Agent (`MOMAgent`) is deployed and is listed under Applications in the left frame. The deployment may take a few seconds to complete. Once deployed, the MOM Agent automatically discovers all JMS servers including their contained topics and queues.

9   Open a browser and go to the following address:

    `http://<wls_system>:7001/MOM/MOMAgent`

    The WLS standard Web service testing page displays for the MOM Agent's management Web service as shown below.
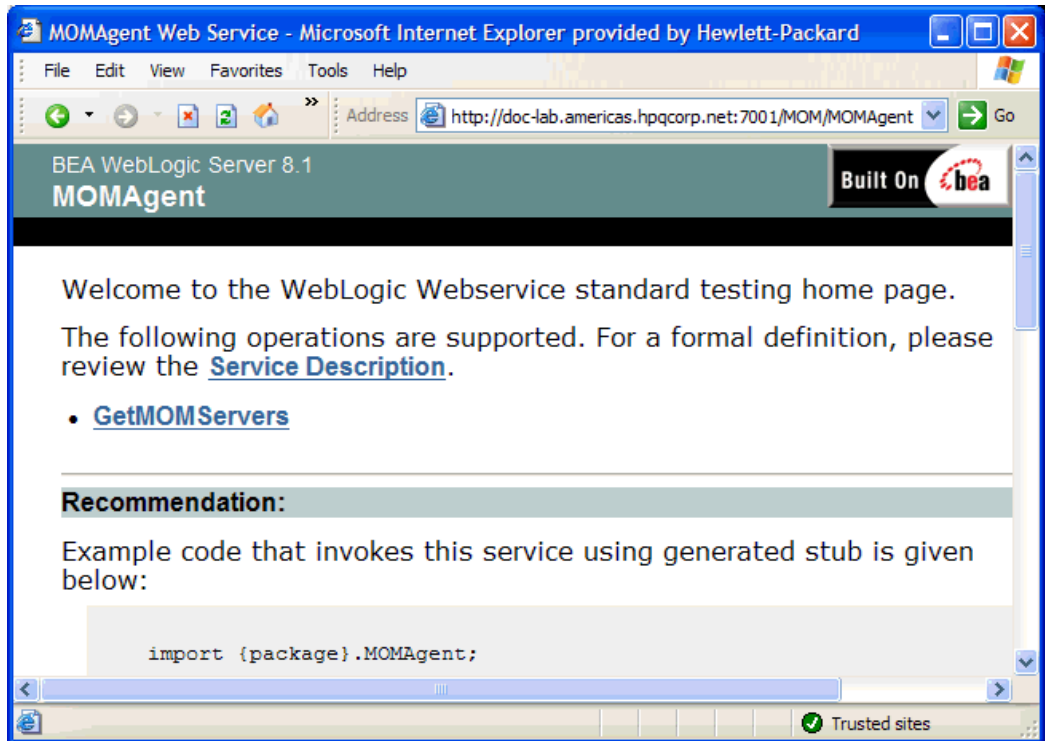
10 Click **Service Description** to see the MOM Agent's WSDL.

11 Refer to the *SOA Manager Administrator Guide* for instructions on creating a MOM IT Service that contains a MOM Agent so that the JMS servers, topics, and queues can be monitored from the BSE.

Configure the WSM J2EE Agent

This chapter provides instructions for configuring the WSM J2EE Agent as well as the Agent's handlers. The following sections are provided in this chapter:

- Configuration Overview
- Configuring Common Handlers
- Excluding Web Services from Being Managed
- Changing the SBA Polling Frequency
- Using the Logging Feature

Configuration Overview

The WSM J2EE Agent contains several configuration files that are used to configure the various components of the WSM J2EE Agent. The configuration files are located in the *<install_dir>*\config directory of the distribution. These configuration files are discussed throughout this chapter.

Table 3-1: WSM J2EE Agent Configuration Files

Configuration Files	Description
agent.xml	The J2EE Agent's main configuration file. In addition to general configuration, this file is used to configure the audit notification component's behavior.
notificationmanager.xml	The notification manager's configuration file. This file configures the manager to send and receive WSMF notifications. This file does not need to be modified.
WSDiscoveryFilter.xml	A configuration file that allows you to exclude Web service from being managed.
sba-wsms.xml	The SBA's configuration file.

Configuration Files	Description
`logging.properties`	The logging configuration file. The file can be used to enable different logging levels as well as configure where log messages are emitted.

Configuring Common Handlers

The J2EE Agent contains a set of common handlers (JAX RPC Handlers) that are used to provide management for Web services. There are three handlers:

- Monitoring handler
- Auditing handler
- Business Metric handler

Common handlers are enabled and disabled in a Web services' web-services.xml deployment descriptor. By default, the Monitoring and Auditing handler are added to this file when a Web service is discovered and therefore are enabled. However, you must manually add the Business Metric handler to enable business content monitoring.

> The monitoring handler is required in the handler chain in order to manage the Web service. In addition, the monitoring handler must be the first handler in the chain.

Configuring the Business Metric Handler

Business content monitoring allows you to find specific content within a SOAP request, response, and failure. When the content is found, an alert is sent to the Network Services server and viewed in the BSE. For example, an OrderEntry Web service may have an OrderTotalAmount field. Using business content monitoring, you can be alerted whenever the value of OrderTotalAmount is greater than $25,000.00.

> Defining a business content alert requires knowledge of the W3C XPath expression language. It is beyond the scope of this documentation to cover the details of XPath. Several books on XPath are available and you can also refer to the W3C website for details. If you are not familiar with XPath, you should consult a developer before defining a business content alert.

To configure the Business Metric handler and enable business content monitoring:

1   Using a text editor, open a Web service's `web-services.xml` file. The file is typically located in a Web application's `/WEB-INF` directory.

2   Add the following XML content:

```
<handler class-name=
      "com.hp.wsm.agent.bizmetrichandler.BusinessMetricHandler">
   <init-params>
      <init-param value="" name=""/>
   </init-params>
</handler>
```

3   Use the following table for a reference of valid parameters. A full example is provided after the table.

Table 3-2: Business Metric Parameters

Parameter Name	Value
catalog	The Web application's name.
servicename	The Web service's name.
serviceporttype	The port type as defined in the WSDL file for the Web service.
Name	A name for the business content metric. Any name can be used.
OperationName	An operation in the service that contains the business metric value you want to monitor.
Expression	An XPath expression which selects the business metric value from the operation.
ProcessRequest	Allows a request to be processed. Valid values are true or false.
ProcessResponse	Allows a response to be processed. Valid values are true or false.
ProcessFault	Allows a fault to be processed. Valid values are true or false.
Numeric	Allows you to define whether the metric represents a sum or a running total of matches for the metric. True indicates that the metric is a sum, false indicates the number of matches for the metric.
Comparator	A comparator used to test the business metric returned by the expression.
Threshold	A threshold value for the business metric. If the metric is a string, the threshold value must be lexical (i.e., hpq). If the metric is a number, the threshold value must be numeric (i.e., 30).
LogPayload	Includes the payload in the log. Valid values are true or false.
Message	A message to be included with the business content alert.
""	Any name space prefixes (including URI) that appear in the XPath expression. This is entered as a value/name pair (i.e., value=<uri> name="<namespace>).

Listing 3-1: An Example Business Metric Configuration for HelloWorldService

```
<handler class-name=
        "com.hp.wsm.agent.bizmetrichandler.BusinessMetricHandler">
    <init-params>
        <init-param value="HelloWorldService" name="catalog"/>
        <init-param value="HelloWorldService" name="servicename"/>
        <init-param value="HelloWorldServicePort"
                    name="serviceporttype"/>
        <init-param value="IntVal" name="Name"/>
        <init-param value="sayHello" name="OperationName"/>
        <init-param value="//m:sayHello/intVal" name="Expression"/>
        <init-param value="true" name="ProcessRequest"/>
        <init-param value="false" name="ProcessResponse"/>
        <init-param value="false" name="ProcessFault"/>
        <init-param value="true" name="Numeric"/>
        <init-param value="equals" name="Comparator"/>
        <init-param value="20" name="Threshold"/>
        <init-param value="false" name="LogPayload"/>
        <init-param value="intVal is 20" name="message" />
        <init-param value="http://www.hp.com/wsme/samples/webservices"
                    name="xmlns:m"/>
    </init-params>
</handler>
```

This business metric will monitor all requests sent to the HelloWorldService. If the sayHello operation is found and contains a value equal to 20, an alert for the metric is captured and sent to the BSE.

4   Save and close the `web-services.xml`.

5   Redploy the application.

Configuring the Audit Publisher

The audit handler uses a publisher to send audit data to the Network Services Audit Service. Audit data is viewed using the BSE. The behavior of the publisher can be configured. In addition, you can use the audit publisher to disable the audit handler.

To configure the audit publisher:

1   Using a text editor, open *<install_dir>*`\config\agent.xml`.

2   Under the `<components>` node, find the `<component>` element with the `id` attribute `Auditing`.

3   Enter parameters as a name/value pair. Valid parameters are listed in the table below.

```
<component id="Auditing"
          class="com.hp.wsm.agent.auditing.WeblogicAuditingImpl">
    <parameters>
        <parameter name="disabled" value="false" />
        <parameter name="traceThreshold" value="200" />
        <parameter name="publishInterval" value="3600000" />
        <parameter name="payloadOption" value="request-response" />
        <parameter name="payloadFilter" value="all" />
    </parameters>
</component>
```

Table 3-3: Audit Notification Component Parameters

Parameter	Description
disabled	This parameter is used to disable auditing in the Agent. Auditing is disabled even if the auditing handler is defined in web-services.xml. If set to false, auditing is enabled. If set to true, auditing is disabled.
traceThreshold	The number of message traces to keep in memory before submitting a batch to the Audit Service. The default value is 200 messages.
publishInterval	The maximum amount of time (in Milliseconds) that is allowed to elapse with no message traces being published. The default value is 3600000 Milliseconds.
payloadOption	Which messages are logged. Valid values are **none**, **request**, **reponse**, and **request-response**.
payloadFilter	How to filter message logging. Valid values are **all**, and **failure**.

4   Save and close the file.

5   Restart the WebLogic server for the changes to take effect.

Disabling a Handler

If you want to disable a handler, you must remove (or comment) the handler from the application's `web-services.xml` file.

To disable a handler:

1   Using a text editor, open a Web service's `web-services.xml` file. The file is typically located in a Web application's `/WEB-INF` directory.

2   Comment the handler you want to disable. For example:

```
<!--

<handler class-name="com.hp.wsm.agent.auditing.AuditingHandler">
    <init-params>
        <init-param value="HelloWorldService" name="catalog"/>
        <init-param value="HelloWorldService" name="servicename"/>
        <init-param
            value="http://www.hp.com/wsme/samples/webservices"
            name="servicenamespace"/>
        <init-param value="HelloWorldServicePort"
                    name="serviceporttype"/>
    </init-params>
</handler>

-->
```

3   Save and close the `web-services.xml`.

4   Redploy the application.

Removing All Handlers

A Web service's `web-services.xml` file is modified in order to add management handlers into the SOAP pipeline. If you no longer want the Web service to contain the management handlers, the Agent provides a simple JSP to remove the handlers.

To remove all handlers from a Web service:

1   Make sure the WLS administration server is started.

2   Open a browser.

3   Go to the Agent's `wsm/handlerRemoval.jsp`. For example:

    `http://myhost.com:7001/wsm/handlerRemoval.jsp`

    (Change the host and port to match you deployment.)

4   On the Handler Removal Form, select a check box to remove the WSM Agent handlers from a Web service.

5   Click the **Remove Handlers** button. The Processing screen displays and verifies the removal of the WSM handlers for the selected Web services.

6   Close the browser.

7   Un-deploy the Web service application.

> If you do not un-deploy the application, the Web service will be rediscovered and the handlers will be written to the `web-services.xml` file again. To exclude a Web service from being managed, See "Excluding Web Services from Being Managed" section below.

Excluding Web Services from Being Managed

You can configure the WSM J2EE Agent to exclude a Web service from being managed. Web services that are deployed to standalone managed servers, as well as managed servers in a cluster, are excluded as well. Specifically, the Web service's management WSDL will not be available.

To exclude a Web service from being managed:

1   Using a text editor, open *<install_dir>*`\config\WSDiscoveryFilter.xml`.

2   In the `<eliminate>` node, create a `<pattern>` element for the Web services you would like to exclude. The pattern is based on an application name and a service name. For example:

    ```
    <pattern web-app-name-pattern="HelloWorldService" service-name-
    pattern=".*"/>
    ```

    This pattern would exclude all services that are part of the HelloWorldService application.

3   Save and close the `WSDiscoveryFilter.xml` file.

4   Restart the WebLogic server for the change to take effect.

Changing the SBA Polling Frequency

The SBA component periodically polls the Web services platform to discover any new or removed Web services and creates or removes management services based on the discovery. The frequency of the poll is set for every 2 minutes by default.

To change the polling frequency:

1   Using a text editor, open *<install_dir>*`\config\sba-wsms.xml`.

2   In the `<DiscoveryObjects>` node, change the `<PollFrequency>` element to a value in Milliseconds.

3   Save and close the `sba-wsms.xml` file.

4   Restart the WebLogic server for the changes to take effect.

Configuring a Trust Keystore

The instructions in this section are only required if you selected HTTPS as the Agent's protocol in `agent-setup.bat` during deployment. If you want the Agent to use HTTPS within the WLS environment and you did not select HTTPS, you must un-deploy the agent then redeploy the agent with HTTPS enabled. See "Modify the Agent-Setup Script" in Chapter 2.

The SBA controller, embedded in the Agent, sends http/s requests from the commons-httpclient to the wsmf publisher. The commons-httpclient uses the JSSE keystores for SSL communication, while the publisher, living within Weblogic, uses Weblogic's trust keystore for the SSL handshake. As they do not share a common keystore the client cannot confirm the server's identity. There are two methods you can use to ensure that both the SBA controller and the wsmf publisher use the same trust keystore.

*   Add the Weblogic identity key to the JSSE trust store so that the JSSE implementation is aware of Weblogic's identity.

*   Change the JSSE trust store to point to the Weblogic's trust keystore. For example,

    `Djavax.net.ssl.trustStore=`*<bea_home>*`\weblogic81\server\lib\DemoTrust.jks`

    > This may not be desirable as it will force all JSSE users/applications in the JVM to use the same keystore for all SSL communications.

Using XPL Logging

The WSM J2EE Agent uses HP OpenView Cross Platform (XPL) logging. Instructions for installing and configuring XPL logging are described below.

Installing XPL

The XPL Component contains logging and tracing tools. Install the component by running the appropriate installer in the `/Support` directory of the SOA Manager CD.

Configuring XPL

The WSM J2EE Agent automatically creates log files in the log subdirectory of the HP OpenView data directory. On Windows, the default directory is `c:\Program Files\HP OpenView\data\log`. On Unix, the default directory is `/var/opt/OV/log`.

The log file name has the format:

mip-j2ee-agent[unique].sequence.locale

For example:

```
mip-j2ee-agent 0.0.en_US
```

This file is the WSM J2EE Agent log file created for the US English locale. The WSM J2EE Agent creates a log file for an English locale and a second file for your system's locale if it is different from English.

The WSM J2EE Agent will create up to 10 log files, each file containing up to 1 megabyte of data. The log files will have sequence numbers 0 through 9. When the maximum number of log files is exceeded, the sequence 0 log file is overwritten.

You can change the maximum number of log files and log file size using the HP OpenView Cross Platform tool, `ovconfchg`. This program is installed in the /bin directory of the HP OpenView installation directory. An example of using this tool is shown below.

```
ovconfchg -ns xpl.log.OvLogFilesHandler -set filecount 12 -set size 2
```

This command sets the maximum number of log files to 12 and the maximum log file size to 2 megabytes. For more information about `ovconfchg`, see the help documentation in the help subdirectory of the HP OpenView installation directory.

Configuring Log Levels

You can change the log levels by editing the `logging.properties` file in the JRE lib directory. The log levels are: SEVERE, WARNING, INFO, FINE, FINER, and FINEST. By default the log level is set to INFO. You must restart the WSM J2EE Agent for the changes to take effect.

For example, you can add the following line to the end of `logging.properties`:

```
com.hp.ov.mip.j2ee = SEVERE
```

Troubleshooting

This chapter provides common troubleshooting tasks when using the WSM J2EE Agent. In addition, refer to the *SOA Manager Release Notes* for the latest information on the WSM J2EE Agent.

Runtime Problems

Remote Services is not Discovered

The WSM J2EE Agent does not discover services deployed on a remote managed server.

Solution:

Check for errors in the remote managed server's log file (*<bea_home>*/weblogic8/common/nodemanager/*<managed_server>*/*<managed_server* >.log file on the remote system. If there's an error similar to:

```
####<Sep 1, 2004 7:36:31 AM PDT> <Error> <HTTP> <ovw025> <remote2>
<ExecuteThread: '2' for queue: 'weblogic.kernel.System'> <<WLS
Kernel>> <> <BEA-101216> <Servlet: "WebServiceServlet" failed to
preload on startup in Web application: "wsm".
javax.servlet.ServletException: ERROR: The handler class:
com.hp.wsm.agent.handler.WsmfHandler from handler-chain:
WsmfEventHandlerChain could not be loaded. Please ensure that this
class name is correct and is included in the web service.
        at
weblogic.webservice.server.servlet.WebServiceServlet.initLocal()V(Web
ServiceServlet.java:132)
        at
weblogic.webservice.server.servlet.WebServiceServlet.init()V(WebServi
ceServlet.java:86)
        at
javax.servlet.GenericServlet.init(Ljavax.servlet.ServletConfig;)V(Gen
ericServlet.java:258)
        at
weblogic.servlet.internal.ServletStubImpl$ServletInitAction.run()Ljav
a.lang.Object;(ServletStubImpl.java:1018)
```

1   Verify that the WSM_AGENT_HOME variable is set correctly. In the
    <managed_install_dir>/bin/setLocalAgentEnv.<bat | sh> file,
    WSM_AGENT_HOME must be set to the *<managed_install_dir>*. The
    <managed_install_dir> is the directory where you unzipped the wls8-wsme-
    managed-server.zip file.

2   Verify that the WebLogic Node Manager startup script
    (*<bea_home>*/weblogic81/server/bin/startNodeManager.<cmd | sh> includes
    a call to *<managed_install_dir>*/bin/setLocalAgentEnv.bat.

3   Restart the Node Manager if you modified either of the files in the above steps.

Performance Data not Available from Cluster

Requests are sent to servers in a cluster but the performance data is not updated.

Solution:

1   Check the CLASSPATH, wsm-commons-qname.jar must be at the beginning. If
    weblogic.jar appears earlier, the call to setLocalAgentEnv.<bat | sh> is most
    likely misplaced. In this case, you may find an exception in the Weblogic log files.

    ```
    java.lang.NoSuchMethodError:
    javax.xml.namespace.QName.<init>(Ljava/lang/String;Ljava/lang/Stri
    ng;Ljava/lang/String;)V.>
    ```

2   Restart all of the servers in the cluster.

Non-Existent Services in deployed services list

When you run http://myhost.com:7001/wsmf (change host and port to match your
deployment), there are services listed that no longer exist.

Solution:

This situation should only occur if you modified the config.xml file by hand. You'll need
to manually cleanup the *<install_dir>*/webapps/wsmf/WEB-INF/server-
config.wsdd file. Remove all the services that no longer exist and restart the WebLogic
admin server.

WsmfEventHandlerChain could not be loaded

Received a message when starting up the WebLogic Server:

```
ERROR: The handler class: com.hp.wsm.agent.handler.WsmfHandler from
handler-chain: WsmfEventHandlerChain could not be loaded. Please
ensure that this class name is correct and is included in the web
service.
```

Solution:

1   Verify that the WSM_AGENT_HOME is set to the correct location. Refer to the "Modify
    the WebLogic Startup Script" section in Chapter 2 for details.

2   Verify that the setLocalAgentEnv script is called from the WebLogic startup script.
    Refer to the "Modify the WebLogic Startup Script" section in Chapter 2 for details.

Index