Symantec™ High Availability Agent for WebLogic Server Installation and Configuration Guide

Windows

6.1

June 2014

Symantec

Symantec High Availability Agent for WebLogic Server Installation and Configuration Guide

The software described in this book is furnished under a license agreement and may be used only in accordance with the terms of the agreement.

Agent Version: 6.1.0.0

Document version: 6.1.0.0.0

Legal Notice

Symantec Corporation
350 Ellis Street
Mountain View, CA 94043

http://www.symantec.com

Technical Support

Symantec Technical Support maintains support centers globally. Technical Support's primary role is to respond to specific queries about product features and functionality. The Technical Support group also creates content for our online Knowledge Base. The Technical Support group works collaboratively with the other functional areas within Symantec to answer your questions in a timely fashion. For example, the Technical Support group works with Product Engineering and Symantec Security Response to provide alerting services and virus definition updates.

Symantec's support offerings include the following:

- A range of support options that give you the flexibility to select the right amount of service for any size organization
- Telephone and/or Web-based support that provides rapid response and up-to-the-minute information
- Upgrade assurance that delivers software upgrades
- Global support purchased on a regional business hours or 24 hours a day, 7 days a week basis
- Premium service offerings that include Account Management Services

For information about Symantec's support offerings, you can visit our website at the following URL:

www.symantec.com/business/support/index.jsp

All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policy.

Contacting Technical Support

Customers with a current support agreement may access Technical Support information at the following URL:

www.symantec.com/business/support/contact_techsupp_static.jsp

Before contacting Technical Support, make sure you have satisfied the system requirements that are listed in your product documentation. Also, you should be at the computer on which the problem occurred, in case it is necessary to replicate the problem.

When you contact Technical Support, please have the following information available:

- Product release level
- Hardware information

- Available memory, disk space, and NIC information
- Operating system
- Version and patch level
- Network topology
- Router, gateway, and IP address information
- Problem description:
  - Error messages and log files
  - Troubleshooting that was performed before contacting Symantec
  - Recent software configuration changes and network changes

Licensing and registration

If your Symantec product requires registration or a license key, access our technical support Web page at the following URL:

www.symantec.com/business/support/

Customer service

Customer service information is available at the following URL:

www.symantec.com/business/support/

Customer Service is available to assist with non-technical questions, such as the following types of issues:

- Questions regarding product licensing or serialization
- Product registration updates, such as address or name changes
- General product information (features, language availability, local dealers)
- Latest information about product updates and upgrades
- Information about upgrade assurance and support contracts
- Information about the Symantec Buying Programs
- Advice about Symantec's technical support options
- Nontechnical presales questions
- Issues that are related to CD-ROMs or manuals

Support agreement resources

If you want to contact Symantec regarding an existing support agreement, please contact the support agreement administration team for your region as follows:

Asia-Pacific and Japan	customercare_apac@symantec.com
Europe, Middle-East, and Africa	semea@symantec.com
North America and Latin America	supportsolutions@symantec.com

Documentation

Product guides are available on the media in PDF format. Make sure that you are using the current version of the documentation. The document version appears on page 2 of each guide. The latest product documentation is available on the Symantec website.

https://sort.symantec.com/documents

Your feedback on product documentation is important to us. Send suggestions for improvements and reports on errors or omissions. Include the title and document version (located on the second page), and chapter and section titles of the text on which you are reporting. Send feedback to:

doc_feedback@symantec.com

For information regarding the latest HOWTO articles, documentation updates, or to ask a question regarding product documentation, visit the Storage and Clustering Documentation forum on Symantec Connect.

https://www-secure.symantec.com/connect/storage-management/forums/storage-and-clustering-documentation

About Symantec Connect

Symantec Connect is the peer-to-peer technical community site for Symantec's enterprise customers. Participants can connect and share information with other product users, including creating forum posts, articles, videos, downloads, blogs and suggesting ideas, as well as interact with Symantec product teams and Technical Support. Content is rated by the community, and members receive reward points for their contributions.

http://www.symantec.com/connect/storage-management

Contents

Technical Support 4

Chapter 1 Introducing the Symantec High Availability Agent for WebLogic Server 10

About the Symantec High Availability agent for WebLogic Server 10
Supported software 11
About WebLogic Server 11
Agent functions 12
Online 12
Offline 12
Monitor 13
Clean 14
Resource configurations supported by the agent 14

Chapter 2 Installing, upgrading, and removing the agent for WebLogic Server 16

Before you install the agent for WebLogic Server 16
Installing the VCS agent for WebLogic Server 16
Removing the VCS agent for WebLogic Server 17
Upgrading the agent for WebLogic Server 18

Chapter 3 Configuring the agent for WebLogic Server 20

About configuring the agent for WebLogic Server 20
Agent attributes for WebLogic Server 21
Attributes used in different resource configurations 28
Using WebLogic provided scripts 30
Editing the WebLogic stop script 31
Managing WebLogic servers with identical names 32
Avoiding storing unencrypted credentials in startup/shutdown scripts 33

Chapter 4 Configuring the service groups for WebLogic Server 34

About configuring a service group for the agent for WebLogic Server 34
Configuring WebLogic Server for high availability using VCS 34

Chapter 5 Troubleshooting the agent for WebLogic Server 36

Using correct software and operating system versions 36
Problems starting a Managed server through the Administrative console 36
Unable to bring two or more VCS resources offline simultaneously 37
Unable to bring two or more resources offline simultaneously 37
Reviewing log files 38
Using VCS log files 38
Using log files 38
Inspecting agent logs 38
Inspecting temporary log files generated by agent 38
Using WebLogic Server components' log files 39

Appendix A Sample Configurations 40

About the sample configuration for the agent for WebLogic Server 40
Sample agent type definition for WebLogic Server 40
Sample service group configuration for WebLogic Server 41
Sample resource configurations for WebLogic Server 43
Node Manager without SLM enabled 43
Node Manager with SLM enabled 44
Administrative Server (NM) without SLM enabled 45
Administrative Server (NM) with SLM enabled 46
Managed Server (NM) without SLM enabled 47
Managed Server (NM) with SLM enabled 48
Managed Server (NNM) without SLM enabled 49
Managed Server (NNM) with SLM enabled 50
Administrative Server (NNM) without SLM enabled 51
Administrative Server (NNM) with SLM enabled 52
Service group dependencies for WebLogic Server 53
Sample configuration in a VCS environment 54

Appendix B Command line pattern matching for Node Manager based configurations 57

ServerRole is NodeManager 57
ServerRole is Administrative and ServerStartProgram is null 58
ServerRole is Managed and ServerStartProgram is null 59

Appendix C Command line pattern matching for non Node Manager based configurations 61

ServerRole is Administrative and ServerStartProgram is non-null 61
ServerRole is Managed and ServerStartProgram is non-null 62

Index 64

Chapter 1

Introducing the Symantec High Availability Agent for WebLogic Server

This chapter includes the following topics:

- About the Symantec High Availability agent for WebLogic Server
- Supported software
- About WebLogic Server
- Agent functions
- Resource configurations supported by the agent

About the Symantec High Availability agent for WebLogic Server

Symantec High Availability agents monitor specific resources within an enterprise application, determine the status of these resources, and start or stop them according to external events.

The Symantec High Availability agent for WebLogic Server is named weblogic. It consists of a resource type declaration and the agent DLL. The agent is responsible for starting, stopping, monitoring, and detecting failures of WebLogic Server (WLS) components.

See the following Technical Support TechNote for the latest updates or software issues for this agent:

http://seer.entsupport.symantec.com/docs/282004.htm

Supported software

For information on the software versions that the Symantec High Availability agent for WebLogic Server supports, see the Symantec Operations Readiness Tools (SORT) site: https://sort.symantec.com/agents.

About WebLogic Server

A WebLogic Server (WLS) domain is a logical organization of WebLogic servers. The WLS domain consists of an Administrative server and one or more Managed servers.

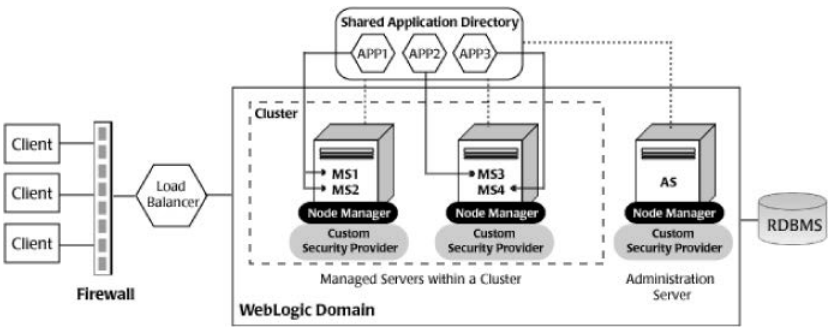
These components are described as follows:

- Administrative server
  An Administrative server is a J2EE application server that provides centralized administration for a WLS domain.
- Managed server
  A Managed server is a J2EE application server that hosts J2EE applications, components, and resources.

A Node Manager is a process that controls all WebLogic Server instances running on a single system or a virtual machine. The Node Manager can detect server failure and can restart the server almost instantaneously.

Figure 1-1 shows a typical WLS domain setup.

Figure 1-1 Typical WLS domain setup



The agent is WebLogic cluster agnostic. In other words, this agent can provide high availability for stand-alone WebLogic Servers and can support Managed servers that participate in a WebLogic cluster.

Agent functions

The agent consists of resource type declarations and agent executables. The agent executables are organized into online, offline, monitor, and clean functions.

Online

The online function performs the following tasks:

- Performs a preliminary check to ensure that the WebLogic Server component is not already running.
- Checks the value of the ServerRole attribute set for the resource. If the value of the attribute is Managed, the online operation may delay the Managed server startup process till the Administrative server is initialized. For details, refer to description of attributes AdminServerMaxWait and RequireAdminServer.
- Starts the WebLogic Server component.
- Ensures that the component is up and running successfully. The operation uses the wait period that the OnlineTimeout attribute specifies, to enable the component to initialize fully before allowing the monitor operation to probe the newly running server instance.
  The online function starts the WebLogic Server component using the following mechanisms:

Node Manager	Uses the wlst command `startNodeManager`.
Administrative server (NM)	Uses the wlst commands `nmConnect` and `nmStart`.
Managed server (NM)	Uses the wlst commands `nmConnect` and `nmStart`.
Administrative server (NNM)	Uses the script configured in ServerStartProgram attribute.
Managed server (NNM)	Uses the script configured in ServerStartProgram attribute.

Offline

The offline function performs the following tasks:

- Performs a preliminary check to ensure that the WebLogic Server component is not already offline.

- Stops the WebLogic Server component gracefully.
- Ensures that the resource is given enough time to go offline successfully. The operation uses a wait period that the OfflineTimeout attribute specifies, to allow the WebLogic Server component to complete the offline sequence before allowing further probing of the resource.
  The offline function stops the WebLogic Server component using the following mechanisms:

Node Manager	Terminates the Node Manager process.
Administrative server (NM)	Uses the wlst commands `connect` and `shutdown`.
Managed server (NM)	Uses the wlst commands `connect` and `shutdown`.
Administrative server (NNM)	Uses the script configured in ServerStopProgram attribute.
Managed server (NNM)	Uses the script configured in ServerStopProgram attribute.

Monitor

The monitor function performs the following tasks:

- Conducts a first level check on the WLS component to ensure that the WLS component's process is running. The agent identifies the process for the WLS component by applying the pattern matching on command lines of processes running in the system. Review the pattern matching information.
- Depending on the settings that you make, the monitor function can conduct a second level check on the WebLogic Server component.
  The second level check uses the wlst.cmd scripting utility to attempt to connect to the WebLogic Server component.
- Depending upon the value of the MonitorProgram attribute, the monitor function can perform a customized check using a user-supplied monitoring utility.
  The following wlst commands can be used to connect to the WebLogic Server component:

Node Manager	Uses the wlst command `nmConnect`.
Administrative server (NM)	Uses the wlst command `connect`.
Managed server (NM)	Uses the wlst command `connect`.

Administrative server (NNM)	Uses the wlst command `connect`.
Managed server (NNM)	Uses the wlst command `connect`.

Clean

The clean function performs the following tasks:

- Attempts to gracefully shut down the WebLogic Server component.
- For Administrative and Managed server Node Manager based configurations, the clean function attempts the wlst `nmKill` command.
- Identifies the process for the WLS component and kills it.

The default value of the CleanTimeout attribute is 60 seconds. As the clean function may execute two wlst.cmd operations, 60 seconds may be insufficient. You can set this attribute to 120 seconds or more.

Resource configurations supported by the agent

The agent for WebLogic Server supports the following kinds of resource configurations:

Node Manager	
Administrative server Node Manager based configuration (NM)	In this configuration, the agent directs the Node Manager to start the Administrative server.
Managed server Node Manager based configuration (NM)	In this configuration, the agent directs the Node Manager to start the Managed server. The advantage of Node Manager based configurations for WebLogic servers is that the Node Manager is capable of detecting server failure using internal protocol and restarting it almost instantaneously.
Administrative server non-Node Manager based Configuration (NNM)	In this configuration, the agent uses custom or WebLogic provided scripts configured by the user in the ServerStartProgram and ServerStopProgram attributes to start and stop the Administrative server.

Managed server non-Node Manager Configuration (NNM)	In this configuration, the agent uses custom or WebLogic provided scripts configured by the user in the ServerStartProgram and ServerStopProgram attributes to start and stop the Managed server.

The agent distinguishes between Node Manager and non-Node Manager based configurations based on whether the ServerStartProgram attribute is null or non-null. If the value is null, the agent assumes a Node Manager based configuration, otherwise it assumes a non-Node Manager based configuration.

Chapter 2

Installing, upgrading, and removing the agent for WebLogic Server

This chapter includes the following topics:

- Before you install the agent for WebLogic Server
- Installing the VCS agent for WebLogic Server
- Removing the VCS agent for WebLogic Server
- Upgrading the agent for WebLogic Server

Before you install the agent for WebLogic Server

Ensure that you install and configure Symantec Cluster Server before installing the agent for WebLogic Server.

Installing the VCS agent for WebLogic Server

Use the Product Installer to install the agent for WebLogic Server.

Note: Ensure that you have uninstalled the previous version of this agent, if installed.

To install the VCS agent for WebLogic Server

1. Log on to any node in the cluster.

   Ensure that the logged on user has the domain administrative privileges.

2. Download the Agent Pack from the Symantec Operations Readiness Tools (SORT) site: https://sort.symantec.com/agents.

   You can download the complete Agent Pack tar file or the individual agent tar file.

3. Uncompress the file to a temporary location.

4. If you downloaded the complete Agent Pack tar file, navigate to the directory containing the package for the platform running in your environment.

Windows 2003	`cd1\windows\w2k3\vcs\ application\weblogic_agent\` `vcs_version\version_agent\pkgs`
Windows 2003 (IA64)	`cd1\windows\w2k3IA64\vcs\application\weblogic_agent\` `vcs_version\version_agent\Pkgs`
Windows 2003 (x64)	`cd1\windows\w2k3x64\vcs\application\weblogic_agent\` `vcs_version\version_agent\Pkgs`
Windows 2008 (x64)	`cd1\windows\w2k8x64\vcs\application\weblogic_agent\` `vcs_version\version_agent\Pkgs`

5. Double-click **vrtsvcsweblogic.msi.**

   Follow the instructions that the install program provides, to complete the installation of Symantec Cluster Server agent for WebLogic Server.

Removing the VCS agent for WebLogic Server

Perform the following procedure to uninstall the agent for WebLogic Server from a cluster. Perform these steps while the cluster is active.

To uninstall the VCS agent for WebLogic Server

1. Ensure that all clustered VCS resources are offline.

2. From the cluster, remove all the resources that use the agent for WebLogic Server.

3. Perform the following steps on each node from which you want to uninstall the agent. Ensure that you have a user with administrative privileges.

- Click **Start > Settings > Control Panel**.
  - On Windows 2008: Navigate to **Programs and Features**
  - On Windows 2008R2/2012: Navigate to **Programs**>**Programs and Features**
- From the list of programs, select **Veritas Cluster Server 6.1.0.0 Agent for WebLogic Server**.

4 Click **Change/Remove**.

5 Follow the instructions that the uninstall program provides, to complete the uninstallation of the agents for WebLogic Server.

Upgrading the agent for WebLogic Server

Perform the following steps to upgrade the agent with minimal disruption, in a VCS environment.

To upgrade the agent in a VCS environment

1 Login as domain administrator.

2 Verify that your path is *drive*:\Program Files\Veritas\Cluster Server\bin

3 Persistently freeze all the service groups that host the application:

```
C:\> hagrp –freeze GroupName -persistent
```

4 Stop the cluster services forcibly:

```
C:\> hastop -all –force
```

5 Ensure that the agent operations are stopped on all the nodes.

6 Take a back up of the main.cf and types.cf files:

```
C:\> copy drive:\> Program Files\Veritas\Cluster
Server\conf\config\main.cf drive:\>backup\main.cf
```

```
C:\> copy drive:\> Program Files\Veritas\Cluster
Server\conf\config\types.cf drive:\>backup\types.cf
```

7 Uninstall the agent package from all the nodes.

See “Removing the VCS agent for WebLogic Server” on page 17.

8 Install the new agent on all the nodes.

See “Installing the VCS agent for WebLogic Server” on page 16.

9 Navigate to *drive*:\> Program Files\Veritas\Cluster Server\conf\config\types.cf file and verify if any duplicate type definitions exists for WebLogic Server on all the nodes.

If duplicate type definitions exist, remove old type definition from types.cf file and save the file.

Note: To identify the old type definition, compare the new type definition file with the old (backed up) types.cf file.

---

10 Check for the changes in the resource values required, if any, due to the new agent types definition.

Note: To note the list of changed attributes, compare the new type definition file with the old type definition file.

---

11 Start VCS on all nodes in the cluster:

```
C:\> hastart
```

12 Start the agent on all nodes:

```
C:\> haagent -start WebLogic -sys SystemName
```

13 Unfreeze the service groups once all the resources come to an online steady state:

```
C:\> hagrp -unfreeze GroupName -persistent
```

Chapter 3

Configuring the agent for WebLogic Server

This chapter includes the following topics:

- About configuring the agent for WebLogic Server
- Agent attributes for WebLogic Server
- Attributes used in different resource configurations
- Using WebLogic provided scripts
- Managing WebLogic servers with identical names
- Avoiding storing unencrypted credentials in startup/shutdown scripts

About configuring the agent for WebLogic Server

After installing the agent for WebLogic Server, you can create and configure a WebLogic Server resource. Before you configure a resource, review the attributes table that describes the WebLogic Server resource type and its attributes.

To review the sample agent type definition file and service group, refer to the Sample Configurations chapter.

The logging information generated by the agent can be seen in the agent log file, in the Cluster Manager Java console and in the temporary log files generated by the agent when it invokes external scripts. For more details, refer to the following sections:

See “Inspecting agent logs” on page 38.

See “Inspecting temporary log files generated by agent” on page 38.

For

To provide high availability for WebLogic components in a WLS domain in a environment, you must first configure the resources.

Agent attributes for WebLogic Server

Table 3-1 describes the WebLogic Server agent attributes.

Table 3-1 Attributes

Attribute	Description
BEA_HOME	The absolute path to the BEA Home directory of WebLogic installation. The value is used to uniquely identify the ServerRole processes. Type and dimension: string-scalar Default: No default value. Example: c:\bea
AdminURL	The URL of the WebLogic Administrative server. The value of this attribute is used to determine if the Administrative server for the domain is fully online when the ServerRole attribute is Managed and the value of RequireAdminServer and AdminServerMaxWait attributes are set appropriately. Managed WebLogic servers use this URL to establish a connection to the Administrative server to download their configuration. This attribute is only required for a resource whose ServerRole attribute is Managed. If the ServerRole attribute is NodeManager or Administrative, no value should be specified. Type and dimension: string-scalar Default: No default value Example: http://wls90host:7001
DomainName	The name of the WebLogic domain to which the WebLogic server belongs. The attribute is required to connect to the Node Manager using WebLogic utility wlst.cmd, which requires the DomainName and DomainDir attributes to start the Administrative and Managed servers. Type and dimension: string-scalar Default: No default value Example: `WLS90Domain`

Table 3-1 Attributes *(continued)*

Attribute	Description
DomainDir	The domain directory for the WebLogic domain to which the WebLogic server belongs. The attribute is required to connect to the Node Manager using WebLogic utility wlst.cmd, which requires the DomainName and DomainDir attributes to start the Administrative and Managed servers. For Non Node Managed based configuration, this attribute is only required to distinguish the Multiple Servers with same Server Name (Refer to "Managing weblogic server with identical names"). Type and dimension: string-scalar Default: No default value Example: c:\bea\user_projects\domains\WLS90Domain
ListenAddressPort	The hostname and port of the WebLogic server. The format is hostname:port. Ensure that the ListenAddress string resolves to the proper IP Address, using the network name service that you used on the host. The WebLogic Server connects to the ListenAddress on the specified port through the wlst.sh API. Specify this attribute for Administrative and Managed Servers only. Type and dimension: string-scalar Default: No default value Example: `wls90host:7001`
nmListenAddressPort	The hostname and port of the WebLogic Node Manager. The format is hostname:port. The agent for WebLogic Server uses the ListenAddress on the specified port to connect through the wlst.sh API. Type and dimension: string-scalar Example: `wls90host:5556` Default: No default value
nmType	The WebLogic Node Manager type. This type is used while connecting to the Node Manager through the wlst.cmd script. Valid values include the following: ■ plain: plain socket Java-based implementation ■ rsh: RSH implementation ■ ssh: script-based SSH implementation ■ ssl: Java-based SSL implementation Type and dimension: string-scalar Default: ssl Example: `ssl`

Table 3-1 Attributes *(continued)*

Attribute	Description
ResLogLevel	Specifies the logging detail performed by the agent for the resource. Valid values include the following: INFO - Logs error/informational messages and trace messages when error occurs. TRACE - Logs trace messages. The Trace messages are stored in the agent log when the agent monitor function completes. To see trace messages while the agent monitor function is executing, add value DBG_20 to LogDbg attribute of WebLogic resource type. Type and dimension: string-scalar Example: `TRACE` Default: INFO
ServerName	The name of the WebLogic Server. You must specify this attribute for Administrative and Managed Servers only. See "Managing WebLogic servers with identical names" on page 32. Type and dimension: string-scalar Default: No default value Example: AdminServer
WLSUser	The username used for connecting WLST to the Application Server or Node Manager. Type and dimension: string-scalar Default: No default value Example: `weblogic`
WL_HOME	Absolute path to the Product Installation Directory of WebLogic installation. The value is used to locate wlst.cmd utility and Node Manager Home directory. Type and dimension: string-scalar Default: No default value. Example: c:\bea\weblogic

Table 3-1 Attributes *(continued)*

Attribute	Description
ServerRole	Type of WLS component. Must be either Administrative, Managed, or NodeManager. Type of WebLogic Server valid values are: ■ NodeManager: Online operation executes wlst.sh script with startNodeManager() API. Example: startNodeManager(verbose='true',NodeManagerHome='C:\Oracle\Middleware\wlserver_10.3\common\nodemanager',ListenPort='5556',ListenAddress='wls90adminsol') ■ Administrative: Online operation executes wlst.sh script with nmConnect() and nmStart() API. Example: nmStart (‘AdminServer1’) ■ Managed: Online operation executes wlst.sh script with nmConnect() and nmStart() API. Example: nmStart (‘ManagedServer1’) Type and dimension: string-scalar Default: Administrative Example: Managed
WLSPassword	Password used for connecting WLST to Application Server or Node Manager.Use the vcsencrypt -agent command to encrypt the password. If you are using the VCS GUI, the GUI automatically encrypts the password. Type and dimension: string-scalar Default: No default value Example: HTIvKTlTNnINjNKnL

Table 3-1 Attributes *(continued)*

Attribute	Description
AdminServerMaxWait	Specifies the maximum number of seconds that a Managed server's online entry point waits for the domain's Administrative server to respond to a test probe. While this attribute is not required to successfully start, WebLogic Managed servers typically initiate a connection to the Administrative server for downloading updated configuration information. In cases in which the administrator is starting all the WebLogic servers within the cluster at the same time, it would be advantageous for each Managed server to delay the start until the Administrative server has fully initialized. The AdminServerMaxWait attribute provides a way for the administrator to orchestrate such a delay. The online agent function uses the AdminServerMaxWait value to control a repeating cycle of probe, wait, probe, and wait until it successfully detects the presence of the Administrative server. After it detects the server, it then proceeds with the Managed server startup. If the online agent function finds that the Administrative server is not available before the wait time expires, it generates a log warning message and proceeds with server startup. Set the RequireAdminServer attribute to 1 (True) to force the online agent function to wait for a successful Administrative server response until entire duration of online agent function. If this attribute is set to True, the online agent function ignores the value of the AdminServerMaxWait time limit. Type and dimension: integer-scalar Default: 60 Example: 90

Table 3-1 Attributes *(continued)*

Attribute	Description
MonitorProgram	Contains the full path name and command-line arguments for an externally provided monitor program. The monitor entry point will execute this program to perform a user defined WebLogic resource state check. The monitor entry point executes the MonitorProgram under the following conditions: ■ The monitor entry point's first-level process check indicates the WebLogic resource is online ■ The SecondLevelMonitor is set to 0 (False) or SecondLevelMonitor is set to 1 (True) and the second-level check indicates that the WebLogic resource is online. This program is not supplied with the Symantec agent for WebLogic and is externally developed by the end user to satisfy unique requirements. The exit code of the program is interpreted by the monitor entry point as follows: ■ 110 or 0:The WebLogic resource state is ONLINE. ■ 100 or 1: The WebLogic resource state is OFFLINE. ■ 99: The WebLogic resource state is UNKNOWN. ■ Other: The WebLogic resource state is UNKNOWN. Symantec recommends storing the external monitor program on the shared storage device, in the directory specified by the BEA_HOME attribute, to ensure that the file is always available on the ONLINE system. Type and dimension: string-scalar Default No default value Example: c:\bea\monitor.cmd
RequireAdminServer	Set the RequireAdminServer attribute to 1 (True) to force the online entry point to wait for a successful Administrative server response until entire duration of Online entry point. If this attribute is set to True, the online entry point ignores the value of the AdminServerMaxWait time limit. Type and dimension: boolean-scalar Default: 0 (False) Example: 1 (True)

Table 3-1 Attributes *(continued)*

Attribute	Description
SecondLevelMonitor	Used to enable second-level monitoring and specify how often it is run. Second-level monitoring is a deeper, more thorough state check of the configured WebLogic resource. The numeric value specifies how often the second-level monitoring routines are run. Examples are as follows: ■ Zero (0) means never run the second-level monitoring routines. ■ One (1) would mean to run it every monitor interval. ■ Two (2) means to run the second-level monitoring routines every second monitor interval, and so on. The agent uses the BEA supplied WebLogic Scripting Tool wlst.cmd command to perform second-level monitoring using its commands `connect()`, `nmConnect()` depending upon the ServerRole. Care should be taken when setting this attribute to large numbers. For example, if the MonitorInterval is set to 60 seconds, and the SecondLevelMonitor is set to 100, then the wlst.cmd command would only get executed every 100 minutes, which may not be as often as intended. To provide maximum flexibility, the value set is not checked for an upper limit. Thus, you could cause the wlst.cmd command to run once a month, if that is what is desired. Type and dimension: integer-scalar Default: 0 Example: 1
ServerStartProgram	The full command line for the script used to start the WebLogic server. Type and dimension: string-scalar Default: No default value Example: c:\bea\user_projects\domains\WLS90Domain\bin\startManagedWebLogic.cmd ManagedServer01 t3://wls90host:7001
ServerStopProgram	The full command line for the script used to stop the WebLogic server. Type and dimension: string-scalar Default Value: No default value Example: c:\bea\user_projects\domains\WLS90Domain\bin\stopManagedWebLogic.cmd ManagedServer01 t3://wls90host:7002

Attributes used in different resource configurations

For each resource configuration, some attributes may be used by the agent and others may not be used. Use the following tables to figure out which attributes must be configured for your resource depending on the required configuration for your resource. In these tables, the following conventions hold true:

- SLM stands for SecondLevelMonitor attribute.
- "Yes" implies that attribute is mandatory for the given configuration.
- "Opt" implies that configuring the attribute is optional for the given configuration.
- "-" implies that the attribute is not used by the agent for the given configuration.

Table 3-2 shows the attributes used by Node Manager based configurations.

Table 3-2 Attributes used by Node Manager based configurations

Resource Configuration/Attributes	Node Manager (SLM=0)	Node Manager (SLM>0)	Administrative Server (NM)	Managed Server (NM)
ResLogLevel	Yes	Yes	Yes	Yes
AdminURL	-	-	-	Yes
BEA_HOME	Yes	Yes	Yes	Yes
WL_HOME	Yes	Yes	Yes	Yes
DomainName	-	Yes	Yes	Yes
DomainDir	-	Yes	Yes	Yes
ListenAddressPort	-	-	Yes	Yes
MonitorProgram	Opt	Opt	Opt	Opt
nmListenAddressPort	Yes	Yes	Yes	Yes
nmType	Yes	Yes	Yes	Yes
ServerName		-	Yes	Yes
ServerRole	Yes	Yes	Yes	Yes
WLSUser	-	Yes	Yes	Yes
WLSPassword	-	Yes	Yes	Yes

Table 3-2 Attributes used by Node Manager based configurations *(continued)*

Resource Configuration/Attributes	Node Manager (SLM=0)	Node Manager (SLM>0)	Administrative Server (NM)	Managed Server (NM)
RequireAdminServer	-	-	-	Yes
AdminServerMaxWait	-	-	-	Yes
SecondLevelMonitor	Yes	Yes	Yes	Yes
ServerStartProgram	-	-	-	-
ServerStopProgram	-	-	-	-

Table 3-3 shows the attributes used by non-Node Manager based configurations.

Table 3-3 Attributes used by non-Node Manager based configurations

Resource Configuration/Attributes	Managed Server (NNM) (SLM=0)	Managed Server (NNM) (SLM>0)	Administrative Server (NNM) (SLM=0)	Administrative Server (NNM) (SLM>0)
ResLogLevel	Yes	Yes	Yes	Yes
AdminURL	Yes	Yes	-	-
BEA_HOME	Yes	Yes	Yes	Yes
WL_HOME	-	Yes	-	Yes
DomainName	-	-	-	-
DomainDir	Opt	Opt	-	Opt
ListenAddressPort	-	Yes	-	Yes
MonitorProgram	Opt	Opt	Opt	Opt
nmListenAddressPort	-	-	-	-
nmType	-	-	-	-
ServerName	Yes	Yes	Yes	Yes
ServerRole	Yes	Yes	Yes	Yes
WLSUser	-	Yes	-	Yes

Table 3-3 Attributes used by non-Node Manager based configurations *(continued)*

Resource Configuration/Attributes	Managed Server (NNM) (SLM=0)	Managed Server (NNM) (SLM>0)	Administrative Server (NNM) (SLM=0)	Administrative Server (NNM) (SLM>0)
WLSPassword	-	Yes	-	Yes
RequireAdminServer	-	-	-	-
AdminServerMaxWait	Yes	Yes	-	-
SecondLevelMonitor	-	Yes	-	Yes
ServerStartProgram	Yes	Yes	Yes	Yes
ServerStopProgram	Yes	Yes	Yes	Yes

The following list shows the kind of resource configuration and the corresponding sample configuration:

- Node Manager with SLM enabled.
- Administrative server (NM) without SLM enabled.
- Administrative server (NM) with SLM enabled.
- Managed server (NM) without SLM enabled.
- Managed server (NM) with SLM enabled.
- Managed server (NNM) without SLM enabled.
- Managed server (NNM) with SLM enabled.
- Administrative server (NNM) without SLM enabled.
- Administrative server (NNM) with SLM enabled.

You can use the sample configurations, mentioned in Appendix A, as reference while configuring your resource.

Using WebLogic provided scripts

WebLogic built-in scripts can be used in non-Node Manager based configurations as values of ServerStartProgram and ServerStopProgram attributes. When you create a domain using the config.cmd utility, WebLogic generates some scripts.

You can use the following scripts to start or stop WebLogic Server instances present in the WebLogic domain:

- To start an Administrative server instance

  DomainDir\bin\startWebLogic.cmd

- To stop an Administrative server instance

  DomainDir\bin\stopManagedWebLogic.cmd

Using `stopWebLogic.cmd` to stop a Administrative server forces you to specify the user name and password in plain-text as command line parameters or as environment variables. Hence, the user can use stopManagedWebLogic.cmd to stop an Administrative server instance.

You can use the following scripts to start or stop a Managed server instance:

- To start a Managed server instance

  DomainDir\bin\startManagedWebLogic.cmd

- To stop a Managed server instance

  DomainDir\bin\stopManagedWebLogic.cmd

Using the script `stopManagedWebLogic.cmd` to stop a Managed server with the admin_url argument causes the shutdown operation to fail when the Administrative server is unavailable. To overcome this, the user can use the `stopManagedWebLogic.cmd` with the Managed server's url as argument in place of the admin_url. Passing the Managed server's url as argument causes the script to execute WLST command `connect()` to the Managed server's URL and execute the WLST `shutdown()` command subsequently. Hence the script succeeds in shutting down the Managed server even when the Administrative server is unavailable.

Editing the WebLogic stop script

A configured resource for a WebLogic Server can use a WebLogic supplied stop script to go offline by specifying it in the ServerStopProgram attribute.

You may encounter an issue with the WebLogic supplied stop scripts, *DomainDir*/bin/stopWebLogic.cmd and *DomainDir*/bin/stopManagedWebLogic.cmd.

These stop scripts send commands to the `wlst.cmd` utility. These commands are written into a temporary file, shutdown.py.

An issue may occur if you have configured two or more resources for servers belonging to the same WebLogic domain. When you attempt to bring these resources

offline at the same time, all the stop scripts attempt to write the wlst commands into the same shutdown.py file. This attempt may create race conditions and some of the stop scripts may fail to complete execution.

To resolve the race condition issue

1. Create a copy of the *DomainDir*/bin/stopWebLogic.cmd file.
2. Rename the copy as *DomainDir*/bin/stopWebLogic_old.cmd.
3. In the stopWebLogic.cmd file, ensure that the wlst commands are sent directly to the stdin of the wlst.cmd utility, instead of being written into a temporary file.

   For example, replace these lines:

```
echo connect^(%userID% %password%
url='%ADMIN_URL%',adminServerName='%SERVER_NAME%'^)
>"shutdown.py" echo
shutdown^('%SERVER_NAME%','Server'^) >>"shutdown.py"
echo exit^(^) >>"shutdown.py" echo Stopping Weblogic
Server...%JAVA_HOME%\bin\java %JAVA_OPTIONS%
weblogic.WLST shutdown.py 2>&1
```

   with the following lines:

```
echo Stopping Weblogic Server...echo connect^(%userID%
%password%
url='%ADMIN_URL%',adminServerName='%SERVER_NAME%'^);s
hutdown^(' %SERVER_NAME%','Server'^) ;exit^(^) |
%JAVA_HOME%\bin\java %JAVA_OPTIONS% weblogic.WLST
```

Managing WebLogic servers with identical names

In a non-Node Manager based configuration, if two Administrative servers having identical names and belonging to different domains, are running on the same system, the agent monitor may yield multiple results while matching the pattern on process command lines.

To avoid any discrepancy, follow these steps for the WebLogic servers.

To manage WebLogic servers with identical names

1. For an Administrative server instance, make a copy of the startWebLogic.cmd file. Rename the copy as startWebLogic_new.cmd.
2. In the startWebLogic_new.cmd file, add this line:

   ```
   set JAVA_OPTIONS=
   -Dweblogic.system.BootIdentityFile=%LONG_DOMAIN_HOME%\servers\
   %SERVER_NAME%\security\boot.properties %JAVA_OPTIONS%
   ```

3. Specify the startWebLogic_new.cmd file in the ServerStartProgram attribute for the weblogic resource.
4. Set the value for DomainDir attribute for the weblogic resource. These steps ensure that *DomainDir* appears in the command line for the Administrative server process. Hence, the Administrative server process is uniquely identified, even if another WebLogic Server instance with the same name is running in the system.

Avoiding storing unencrypted credentials in startup/shutdown scripts

Whenever you configure a weblogic resource that uses WebLogic provided scripts to start and stop the WebLogic server it is recommended to have the boot identity files to avoid storing unencrypted credentials in startup/shutdown scripts. The boot identity file boot.properties should be created for the WebLogic server and placed in the security directory of the server.

For example,

```
c:\bea\wls90\admin\user_projects\domains\WLS90Domain\servers\
ManagedServer01\security
```

For details, refer to

http://edocs.bea.com/wls/docs90/server_start/overview.html#1068976

Note: If you do not have the boot.properties file, and have not provided the username/password to start/stop scripts, the start and stop scripts will prompt you for a username and password. If the cluster invokes the start or stop operation, this prompt causes the operation to fail.

Chapter 4

Configuring the service groups for WebLogic Server

This chapter includes the following topics:

- About configuring a service group for the agent for WebLogic Server
- Configuring WebLogic Server for high availability using VCS

About configuring a service group for the agent for WebLogic Server

To provide high availability for WebLogic Server components in the environment, you must first configure the resources of type Process.

Configuring WebLogic Server for high availability using VCS

Do the following steps to make WebLogic Server components, which are part of a domain, highly available using VCS.

To configure a WebLogic Server for high availability using VCS

1. Create and configure a VCS Service Group that consists of a Lanman, an IP address, a mount point directory, and disk group resources. Refer to the cluster documentation for details about a Service Group.
2. Bring the Service Group online.

3 Install WebLogic Server software. Ensure that you select the mount directory that you created in step 1 as the BEA home directory. While creating the domain using the config.cmd utility, ensure that you configure the WebLogic servers to listen on the virtual address of the Lanman resource.

4 Configure individual weblogic resources for each of the components you want VCS to manage in the service group.

5 Attempt to do the following:

- Online the Service Group.
- Offline the Service Group.
- Switchover the Service Group to remaining systems that are part of the Service groups SystemList attribute.

Chapter 5

Troubleshooting the agent for WebLogic Server

This chapter includes the following topics:

- Using correct software and operating system versions
- Problems starting a Managed server through the Administrative console
- Unable to bring two or more VCS resources offline simultaneously
- Unable to bring two or more resources offline simultaneously
- Reviewing log files

Using correct software and operating system versions

Ensure that no issues arise due to incorrect software and operating system versions. For the correct versions of operating system and software to be installed on the resource systems, refer to SORT site: https://sort.symantec.com/agents

Problems starting a Managed server through the Administrative console

You may encounter problems while starting a Managed server through the Administrative console. When you start a Managed server through the console, the Administrative server sends a request to the Node Manager to start the Managed server. The Administrative server sends this request using SSL communication.

If the Node Manager is running on a virtual host, this communication may fail. This failure may occur because the Node Manager uses default SSL certificates that contain the real host name of the physical node on which the Node Manager is

running. The URL used for connecting to the Node Manager contains the virtual host name of the Node Manager, which is different from the physical host name of the node. The Administrative server rejects the communication because of this mismatch.

To overcome this mismatch, you can perform one of the following tasks:

- Generate new SSL certificates
  You can generate new SSL certificates that contain the virtual host name of the Node Manager. Then, configure the Node Manager to use the new SSL certificates.

  For more details about creating SSL certificates, refer to the following links:

  - http://e-docs.bea.com/wls/docs90/secmanage/ssl.html
  - http://edocs.bea.com/wls/docs90/server_start/nodemgr.html
  - http://e-docs.bea.com/wls/docs90/secmanage/identity_trust.html

  BEA Systems recommends generating new SSL certificates using reliable certification authorities as best security practice. Otherwise, you can generate certificates and keystores which use virtual hostname, using the tools, CertGen and ImportPrivateKey that WebLogic provides.
- Disable the host name verification function
  You can disable the host name verification function in the Administrative server properties. For details about disabling the function, refer to the following link:
  http://e-docs.bea.com/wls/docs90/ConsoleHelp/taskhelp/security/DisableHostNameVerification.html.

Unable to bring two or more VCS resources offline simultaneously

This error may occur if you have configured two or more VCS resources for servers belonging to the same WebLogic domain and VCS attempts to bring these resources offline simultaneously.

See “Editing the WebLogic stop script” on page 31.

Unable to bring two or more resources offline simultaneously

This error may occur if you have configured two or more resources for servers belonging to the same WebLogic domain and attempts to bring these resources offline simultaneously.

See "Editing the WebLogic stop script" on page 31.

Reviewing log files

If the configured VCS or resource is not working properly, you can review the log files to diagnose the problem.

Using VCS log files

In case of problems while using the agent for WebLogic Server, you can access the VCS engine log file for more information about the particular resource.

The VCS engine log file is in the following location:

```
c:\program files\veritas\cluster server\log\engine_A.txt.
```

For a VCS Process resource, you can view the following Process agent log file:

```
c:\program files\veritas\cluster server\log\Process_A.txt.
```

Using log files

In case of problems while using the agent for WebLogic Server, you can access the client log file for more information about the particular resource.

The client log file is present in c:\program files\veritas\cluster server\log\haclient_A.txt.

Inspecting agent logs

The agent log file is present in c:\program files\veritas\cluster server\log\weblogic_A.txt file.

Error and informational messages logged by the agent can be seen in the cluster manager java console Agent logs under weblogic resource type.

Inspecting temporary log files generated by agent

The agent logs output of scripts run by it in %Windir%/temp/VRTSweblogic in files of the form *resourcename.entrypointname*.out.

Example

```
c:\windows\temp\VRTSweblogic\wls90sg_adminserver_weblogic.online.
out
```

Using WebLogic Server components' log files

If a WebLogic Server is facing problems, you can view the server log files to further diagnose the problem.

The logs for WebLogic Server are located in the following directories:

Administrative and Managed servers	Path <DomainDir>\servers\<ServerName>\logs Example `c:\bea\wls90\admin\user_projects\domains\WLS90Domain\servers\managedserver01\logs`
Node Manager	Path <WL_HOME>\common\nodemanager Directory `c:\bea\wls90\admin\weblogic0\common\nodemanager`

Appendix A

Sample Configurations

This appendix includes the following topics:

- About the sample configuration for the agent for WebLogic Server
- Sample agent type definition for WebLogic Server
- Sample service group configuration for WebLogic Server
- Sample resource configurations for WebLogic Server
- Service group dependencies for WebLogic Server
- Sample configuration in a VCS environment

About the sample configuration for the agent for WebLogic Server

The sample configuration depicts the resource types, resources, and resource dependencies within the service group. Review these dependencies carefully before configuring the agent for WebLogic Server. For more information about these resource types, see the *Symantec Cluster Server Bundled Agents Reference Guide*.

Sample agent type definition for WebLogic Server

An example of the WebLogic Server agent type definition file is as follows:

```
type WebLogic (
static keylist LogDbg = { DBG_21 }
static i18nstr ArgList[] = { ResLogLevel, State, IState,
AdminURL, BEA_HOME, WL_HOME, DomainName, DomainDir,
ListenAddressPort, MonitorProgram, nmListenAddressPort, nmType,
ServerName, ServerRole, WLSUser, WLSPassword, RequireAdminServer,
```

```
AdminServerMaxWait, SecondLevelMonitor, ServerStartProgram,
ServerStopProgram }
str ResLogLevel = INFO
str AdminURL
str BEA_HOME
str WL_HOME
str DomainName
str DomainDir
str ListenAddressPort
str MonitorProgram
str nmListenAddressPort
str nmType = ssl
str ServerName
str ServerRole
str WLSUser
str WLSPassword
boolean RequireAdminServer = 0
int AdminServerMaxWait = 60
int SecondLevelMonitor
str ServerStartProgram
str ServerStopProgram
)
```

Sample service group configuration for WebLogic Server

A WebLogic Server resource consists of the following:

Disk Group: Veritas Volume Manager disk group contains information required by the DiskGroup agent to import and export the shared disk object used in support of a clustered WebLogic Server instance. While the use of shared disk is not required to cluster an instance of WebLogic Server, Symantec recommends the use of a shared volume to eliminate the requirement to synchronize local copies of the WebLogic Server binaries and configuration files on each node in a multi-node cluster.

Mount: This resource mounts, monitors, and unmounts the file system that is dedicated to the WebLogic Server installation and configuration files. Use the resource type Mount to create this resource.

Network Interface: This resource monitors the network interface card through which the WebLogic Server communicates with other services.

Virtual IP: This resource configures the virtual IP address dedicated to the WebLogic Server. External services, programs, and clients use this address to communicate with this WebLogic Server instance.

WebLogic Server: This resource starts, stops, and monitors the WebLogic Server instance. Use the WebLogic Server resource type to create this resource.

Figure A-1 shows an example of a single service group with an Administrative Server.

Figure A-1 Service group configuration with Administrative server

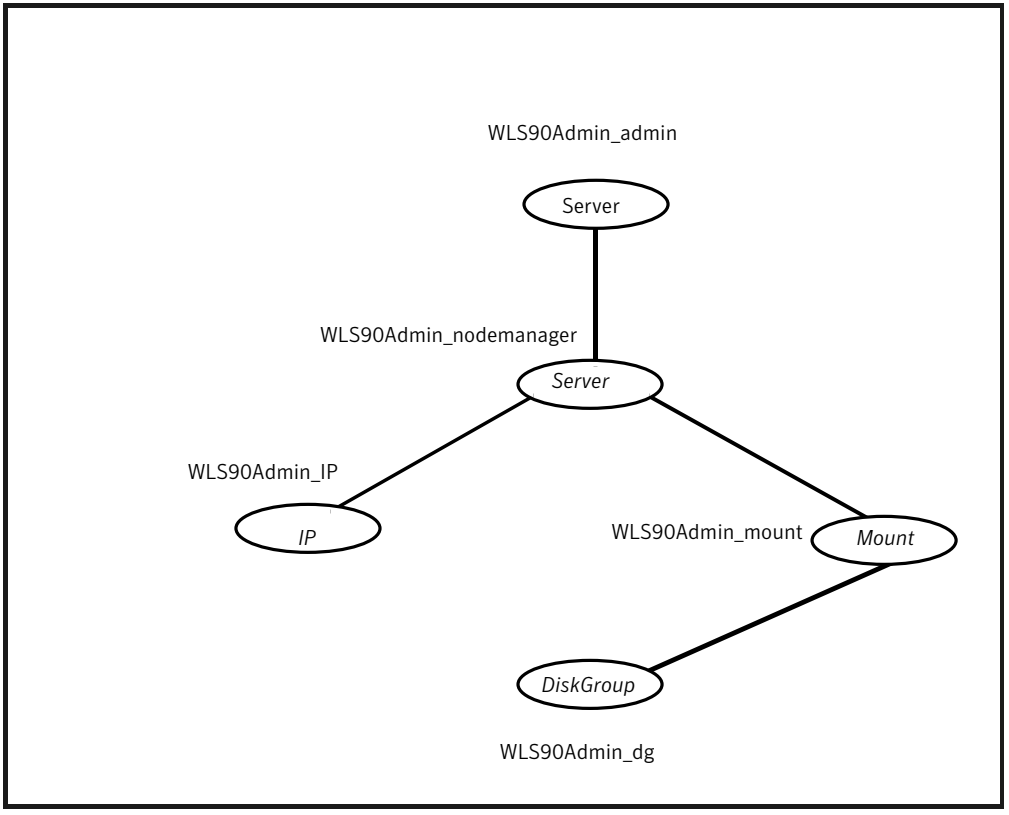
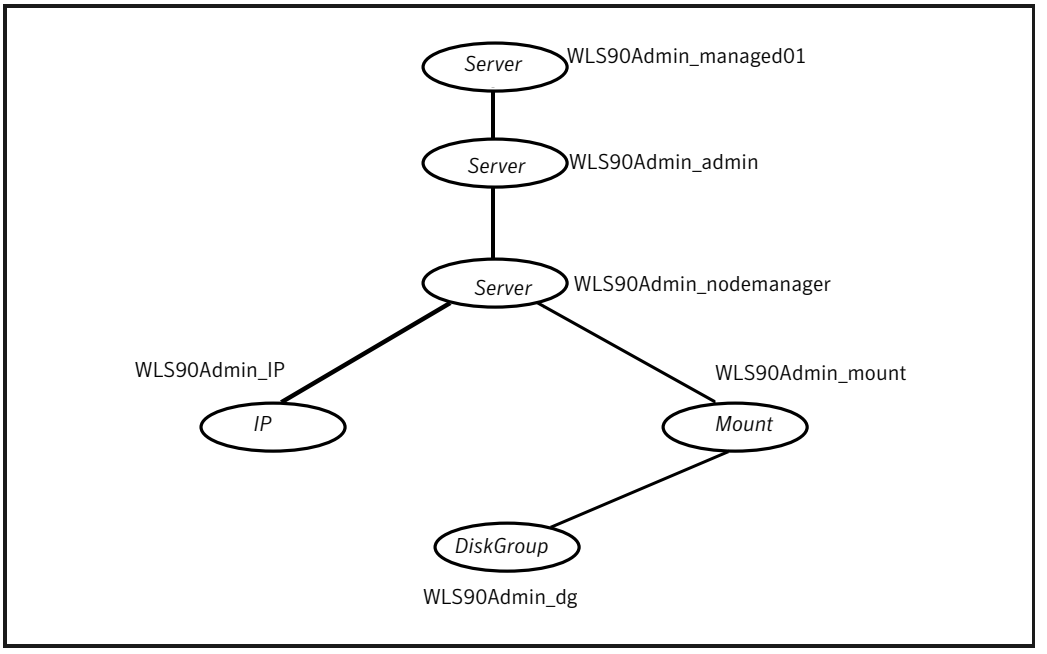


Figure A-2 shows a service group with Administrative and Managed Servers.

Figure A-2 Service group configuration with Administrative and Managed servers



Sample resource configurations for WebLogic Server

The sample resource configurations for WebLogic Server are shown in the following sections.

Node Manager without SLM enabled

Table A-1 depicts a typical configuration for Node Manager with second level monitoring (SLM) not enabled.

Table A-1 Node Manager without SLM enabled

Attribute	Value
ResLogLevel	INFO
AdminURL	
BEA_HOME	C:\Oracle\Middleware

Table A-1 Node Manager without SLM enabled *(continued)*

Attribute	Value
WL_HOME	C:\Oracle\Middleware\wlserver_10.3
DomainName	
ListenAddressPort	
MonitorProgram	
nmListenAddressPort	wlsadmin1:5556
nmType	ssl
ServerName	
ServerRole	NodeManager
WLSUser	weblogic
WLSPassword	EQFsHqkkMNRkL
ServerStartProgram	
ServerStopProgram	
RequireAdminServer	false
AdminServerMaxWait	60
SecondLevelMonitor	0

Node Manager with SLM enabled

Table A-2 depicts a typical configuration for Node Manager with second level monitoring (SLM) enabled.

Table A-2 Node Manager with SLM enabled

Attribute	Value
ResLogLevel	INFO
AdminURL	
BEA_HOME	C:\Oracle\Middleware
WL_HOME	C:\Oracle\Middleware\wlserver_10.3

Table A-2 Node Manager with SLM enabled *(continued)*

Attribute	Value
DomainName	wls_domain1
DomainDir	C:\Oracle\Middleware\user_projects\domains\wls_domain1
ListenAddressPort	
MonitorProgram	
nmListenAddressPort	wlsadmin1:5556
nmType	ssl
ServerName	
ServerRole	NodeManager
WLSUser	weblogic
WLSPassword	EQFsHqkkMNRkL
ServerStartProgram	
ServerStopProgram	
RequireAdminServer	false
AdminServerMaxWait	60
SecondLevelMonitor	1

Administrative Server (NM) without SLM enabled

Table A-3 depicts a typical configuration for Administrative server (NM) with second level monitoring (SLM) not enabled.

Table A-3 Administrative Server (NM) without SLM enabled

Attribute	Value
ResLogLevel	INFO
AdminURL	
BEA_HOME	C:\Oracle\Middleware
WL_HOME	C:\Oracle\Middleware\wlserver_10.3

Table A-3 Administrative Server (NM) without SLM enabled *(continued)*

Attribute	Value
DomainName	wls_domain
DomainDir	C:\Oracle\Middleware\user_projects\domains\wls_domain
ListenAddressPort	wlsadmin:7011
MonitorProgram	
nmListenAddressPort	wlsadmin:5556
nmType	ssl
ServerName	AdminServer
ServerRole	Administrative
WLSUser	weblogic
WLSPassword	EQFsHqkkMNRkL
ServerStartProgram	
ServerStopProgram	
RequireAdminServer	false
AdminServerMaxWait	60
SecondLevelMonitor	0

Administrative Server (NM) with SLM enabled

Table A-4 depicts a typical configuration for Administrative Server (NM) with the second level monitoring (SLM) enabled.

Table A-4 Administrative Server (NM) with SLM enabled

Attribute	Value
ResLogLevel	INFO
AdminURL	
BEA_HOME	C:\Oracle\Middleware
WL_HOME	C:\Oracle\Middleware\wlserver_10.3

Table A-4 Administrative Server (NM) with SLM enabled *(continued)*

Attribute	Value
DomainName	wls_domain
DomainDir	C:\Oracle\Middleware\user_projects\domains\wls_domain
ListenAddressPort	wlsadmin:7011
MonitorProgram	
nmListenAddressPort	wlsadmin:5556
nmType	ssl
ServerName	AdminServer
ServerRole	Administrative
WLSUser	weblogic
WLSPassword	EQFsHqkkMNRkL
ServerStartProgram	
ServerStopProgram	
RequireAdminServer	false
AdminServerMaxWait	60
SecondLevelMonitor	3

Managed Server (NM) without SLM enabled

Table A-5 depicts a typical configuration for Managed Server (NM) with second level monitoring (SLM) not enabled.

Table A-5 Managed Server (NM) without SLM enabled

Attribute	Value
ResLogLevel	INFO
AdminURL	http://wlsadmin:7011
BEA_HOME	C:\Oracle\Middleware
WL_HOME	C:\Oracle\Middleware\wlserver_10.3

Table A-5 Managed Server (NM) without SLM enabled *(continued)*

Attribute	Value
DomainName	wls_domain
DomainDir	C:\Oracle\Middleware\user_projects\domains\wls_domain
ListenAddressPort	wlsadmin:7012
MonitorProgram	
nmListenAddressPort	wlsadmin:5556
nmType	ssl
ServerName	ManagedServer01
ServerRole	Managed
WLSUser	weblogic
WLSPassword	EQFsHqkkMNRkL
ServerStartProgram	
ServerStopProgram	
RequireAdminServer	false
AdminServerMaxWait	15
SecondLevelMonitor	0

Managed Server (NM) with SLM enabled

Table A-6 depicts a typical configuration for Managed Server (NM) with second level monitoring (SLM) enabled.

Table A-6 Managed Server (NM) with SLM enabled

Attribute	Value
ResLogLevel	INFO
AdminURL	http://wlsadmin:7011
BEA_HOME	C:\Oracle\Middleware
WL_HOME	C:\Oracle\Middleware\wlserver_10.3

Table A-6 Managed Server (NM) with SLM enabled *(continued)*

Attribute	Value
DomainName	wls_domain
DomainDir	C:\Oracle\Middleware\user_projects\domains\wls_domain
ListenAddressPort	wlsadmin:7012
MonitorProgram	
nmListenAddressPort	wlsadmin:5556
nmType	ssl
ServerName	ManagedServer01
ServerRole	Managed
WLSUser	weblogic
WLSPassword	EQFsHqkkMNRkL
ServerStartProgram	
ServerStopProgram	
RequireAdminServer	false
AdminServerMaxWait	15
SecondLevelMonitor	1

Managed Server (NNM) without SLM enabled

Table A-7 depicts a typical configuration for Managed Server (NNM) with the second level monitoring (SLM) not enabled.

Table A-7 Managed Server (NNM) without SLM enabled

Attribute	Value
ResLogLevel	INFO
AdminURL	t3://wlsadmin:7001
BEA_HOME	C:\Oracle\Middleware
WL_HOME	

Table A-7 Managed Server (NNM) without SLM enabled *(continued)*

Attribute	Value
DomainName	wls_domain
DomainDir	
ListenAddressPort	
MonitorProgram	
nmListenAddressPort	
nmType	ssl
ServerName	ManagedServer01
ServerRole	Managed
WLSUser	
WLSPassword	
ServerStartProgram	C:\Oracle\Middleware\user_projects\domains\wls_domain \bin\ startManagedWebLogic.cmd ManagedServer01 t3://wlsadmin:7001
ServerStopProgram	C:\Oracle\Middleware\user_projects\domains\wls_domain\bin\ stopManagedWebLogic.cmd ManagedServer01 t3://wlsadmin:7001
RequireAdminServer	false
AdminServerMaxWait	15
SecondLevelMonitor	0

Managed Server (NNM) with SLM enabled

Table A-8 depicts a typical configuration for Managed server (NNM) with second level monitoring (SLM) enabled.

Table A-8 Managed Server (NNM) with SLM enabled

Attribute	Value
ResLogLevel	INFO
AdminURL	t3://wlsadmin:7001
BEA_HOME	C:\Oracle\Middleware

Table A-8 Managed Server (NNM) with SLM enabled *(continued)*

Attribute	Value
WL_HOME	
DomainName	wls_domain
DomainDir	
ListenAddressPort	wlsadmin:7011
MonitorProgram	
nmListenAddressPort	
nmType	ssl
ServerName	ManagedServer01
ServerRole	Managed
WLSUser	weblogic
WLSPassword	EQFsHqkkMNRkL
ServerStartProgram	C:\Oracle\Middleware\user_projects\domains\wls_domain\bin\ startManagedWebLogic.cmd ManagedServer01 t3://wlsadmin:7001
ServerStopProgram	C:\Oracle\Middleware\user_projects\domains\wls_domain\bin\ stopManagedWebLogic.cmd ManagedServer01 t3://wlsadmin:7001
RequireAdminServer	false
AdminServerMaxWait	60
SecondLevelMonitor	1

Administrative Server (NNM) without SLM enabled

Table A-9 depicts a typical configuration for Administrative server (NNM) with second level monitoring (SLM) not enabled.

Table A-9 Administrative Server (NNM) without SLM enabled

Attribute	Value
ResLogLevel	INFO
AdminURL	

Table A-9 Administrative Server (NNM) without SLM enabled *(continued)*

Attribute	Value
BEA_HOME	C:\Oracle\Middleware
WL_HOME	
DomainName	wls_domain
DomainDir	
ListenAddressPort	wls90admsol:7011
MonitorProgram	
nmListenAddressPort	
nmType	ssl
ServerName	AdminServer
ServerRole	Administrative
WLSUser	
WLSPassword	
ServerStartProgram	C:\Oracle\Middleware\user_projects\domains\ wls_domain\ bin\startWebLogic.cmd
ServerStopProgram	C:\Oracle\Middleware\user_projects\domains\ wls_domain\bin\ startWebLogic.cmd
RequireAdminServer	false
AdminServerMaxWait	60
SecondLevelMonitor	0

Administrative Server (NNM) with SLM enabled

Table A-10 depicts a typical configuration for Administrative Server (NNM) with the second level monitoring (SLM) enabled.

Table A-10 Administrative Server (NNM) with SLM enabled

Attribute	Value
ResLogLevel	INFO

Table A-10 Administrative Server (NNM) with SLM enabled *(continued)*

Attribute	Value
AdminURL	
BEA_HOME	C:\Oracle\Middleware
WL_HOME	C:\Oracle\Middleware\wlserver_10.3
DomainName	wls_domain
DomainDir	
ListenAddressPort	wlsadmin:7001
MonitorProgram	
nmListenAddressPort	
nmType	ssl
ServerName	AdminServer
ServerRole	Administrative
WLSUser	weblogic
WLSPassword	EQFsHqkkMNRkL
ServerStartProgram	C:\Oracle\Middleware\user_projects\domains\ wls_domain\bin\ startWebLogic.cmd
ServerStopProgram	C:\Oracle\Middleware\user_projects\domains\ wls_domain\bin\ stopWebLogic.cmd
RequireAdminServer	false
AdminServerMaxWait	15
SecondLevelMonitor	1

Service group dependencies for WebLogic Server

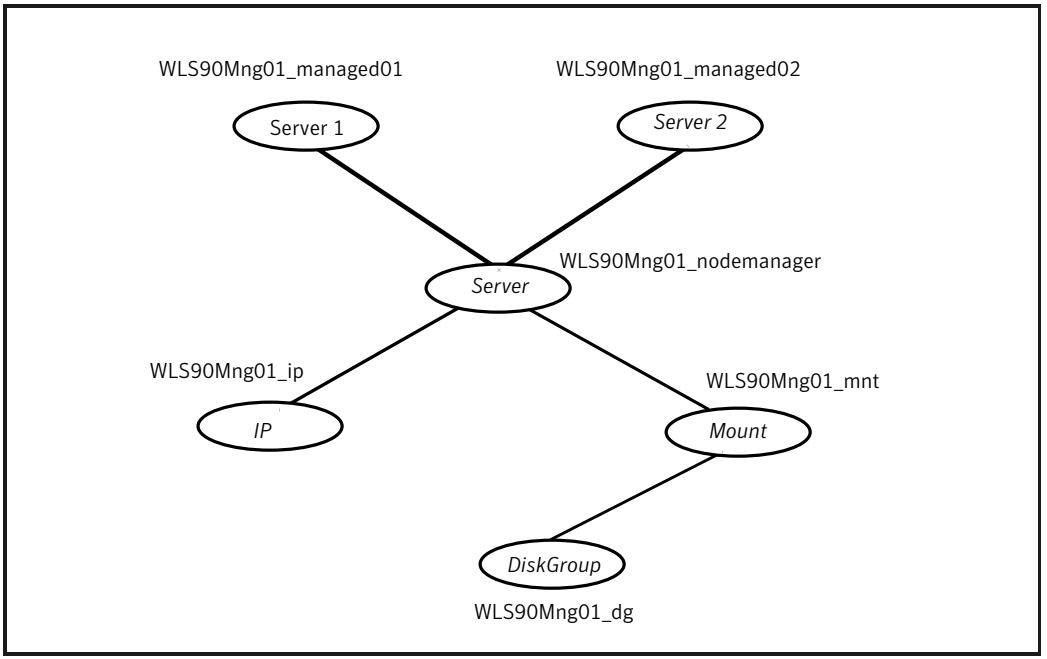
Cluster administrators use Service Group dependencies to create links between unrelated Service Group objects within a cluster. In this version of WebLogic Server, you no longer require Service Group dependencies.

The Managed Server online operation can automatically perform an Administrative Server probe. So even though Managed Server instances depend on the domain

Administrative Server instance, you can have a Service Group with Managed Servers only.

Figure A-3 shows a single Service Group looks with Managed Servers only.

Figure A-3 Single Service group with Managed Servers only



Sample configuration in a VCS environment

```
group wlsadmin
(
SystemList = { systemA = 1, systemB = 2 }
)
DiskGroup wlsadmin_dg
(
DiskGroup = wlsadmin
)
Mount wlsadminmnt
(
MountPoint = "/wls90/admin"
BlockDevice = "/dev/vx/dsk/wls90admin/wlsadmin"
FSType = vxfs
```

```
FsckOpt = "-y"
)
NIC wlsadminnic
(
Device = hme0
NetworkType = ether
)
IP wlsadminip
(
Device = hme0
Address = "192.126.5.166"
NetMask = "255.255.255.0"
)
WebLogic wls_domain (
Critical = 0
BEA_HOME = "C:\Oracle\Middleware"
WL_HOME = "C:\Oracle\Middleware\wlserver_10.3"
DomainName = wls_domain
ListenAddressPort = "wlsadmin:7001"
ServerName = AdminServer
ServerRole = Administrative
WLSUser = weblogic
WLSPassword = gunSlsIssSvsNspSjmHmiMj
SecondLevelMonitor = 5
ServerStartProgram = "C:\Oracle\Middleware\user_projects\domains\
wls_domain\bin\startWebLogic.cmd"
ServerStopProgram = "C:\Oracle\Middleware\user_projects\domains\
wls_domain\bin\stopWebLogic.cmd"
)
WebLogic wls_managedserver1 (
Critical = 0
AdminURL = "t3://wlsadmin:7001"
BEA_HOME = "C:\Oracle\Middleware"
WL_HOME = "C:\Oracle\Middleware\wlserver_10.3"
ListenAddressPort = "10.209.73.90:7003"
ServerName = ManagedServer1
ServerRole = Managed
WLSUser = weblogic
WLSPassword = gunSlsIssSvsNspSjmHmiMj
SecondLevelMonitor = 5
ServerStartProgram = "C:\Oracle\Middleware\user_projects\domains\
wldomain\bin\startManagedWebLogic.cmd ManagedServer1 t3://wlsadmin:7001"
ServerStopProgram = "C:\Oracle\Middleware\user_projects\domains\
```

```
wldomain\bin\stopManagedWebLogic.cmd ManagedServer1 t3://wlsadmin:7001"
)
wls_domain_res requires wls_ip
wls_managedserver1 requires wls_domain_res
wls_domain_res requires wlsadmin_mnt
wlsadmin_mnt requires wlsadmin_dg
wlsadmin_ip requires wlsadmin_nic
```

Appendix B

Command line pattern matching for Node Manager based configurations

This appendix includes the following topics:

- ServerRole is NodeManager
- ServerRole is Administrative and ServerStartProgram is null
- ServerRole is Managed and ServerStartProgram is null

ServerRole is NodeManager

The following pattern matching applies:

- The command line begins with <BEA_HOME>, followed by 0 or more characters, followed by the string java.
- The command line contains the string weblogic.NodeManager.
- The command line contains the string `ListenAddress=<nmListenAddress>` followed by a space.
- The command line contains the string `ListenPort=<nmListenPort>` followed by a space.

Example command line

```
C:\bea\JROCKI~1\jre\bin\javaw.exe -classpath
"C:\bea\JROCKI~1\jre\lib\rt.
jar;C:\bea\JROCKI~1\jre\lib\i18n.jar;C:\bea\patch_weblogic901\profi
```

```
les\default\sys_manifest_classpath\weblogic_patch.jar;C:\bea\JROCKI
~1\lib\tools.jar;C:\bea\WEBLOG~1\server\lib\weblogic_sp.jar;C:\bea\
WEBLOG~1\server\lib\weblogic.jar;C:\bea
\WEBLOG~1\server\lib\webservices.jar;C:\Program
Files\VERITAS\Security\Authentication\bin\AtWrapper.jar;C:\Program
Files\VERITAS\Security\Authentication\bin\vssatgui.jar;C:\Program
Files\VERITAS\Security\Authentication\bin\VxHelpViewer.jar;
C:\Program
Files\VERITAS\Security\Authentication\bin\VxHelpViewerl10n.jar"
-DListenAddress=localhost
-DNodeManagerHome=c:/bea/weblogic90/common/nodemanager
-DQuitEnabled=true -DListenPort=5556 weblogic.NodeManager "-v"
```

ServerRole is Administrative and ServerStartProgram is null

The following pattern matching applies:

- The command line begins with *<BEA_HOME>*, followed by 0 or more characters, followed by the string java followed by 0 or more characters, followed by `weblogic.Name=<AdminServerName>`, followed by space.
- The command line ends with weblogic.Server.
- The command line contains *<DomainDir>* followed by front slash or back slash.

Example command line

```
C:\bea\wls90\admin\JROCKI~1\jre\bin\java
-Dweblogic.Name=AdminServer
-Djava.security.policy=C:\bea\wls90\admin\WEBLOG~1\server\lib\weblo
gic.policy
"-Djava.library.path=C:\bea\wls90\admin\WEBLOG~1\server\bin;.;C:\WI
NDOWS\system32;C:\WINDOWS;C:\bea\wls90\admin\WEBLOG~1\server\native
\win\32;C:\bea\wls90\admin\WEBLOG~1\server\bin;C:\bea\wls90\admin\J
ROCKI~1\jre\bin;C:\bea\wls90\admin\JROCKI~1\bin;C:\bea\wls90\admin\
WEBLOG~1\server\native\win\32\oci920_8;c:\program
files\mks\mksnt;C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\
Wbem;C:\Program Files\VERITAS\VERITAS Object Bus\bin;C:\Program
Files\VERITAS\VERITAS Volume Manager 4.3\;C:\Program
Files\VERITAS\VRTSjre\AccessBridge;C:\Program
Files\VERITAS\Security\Authentication\bin;C:\Program
Files\VERITAS\VRTSPerl\bin;C:\Program
```

```
Files\VERITAS\comms\llt;C:\Program
Files\VERITAS\comms\gab;C:\Program Files\VERITAS\Cluster
server\bin;C:\Program Files\VERITAS\Cluster
server\bin\VCW;C:\Program Files\Microsoft SQL Server\80\Tools\BINN"
-Djava.class.path=.;C:\bea\wls90\admin\patch_weblogic901\profiles\d
efault\sys_manifest_classpath\weblogic_patch.jar;C:\bea\wls90\admin
\JROCKI~1\lib\tools.jar;C:\bea\wls90\admin\WEBLOG~1\server\lib\webl
ogic_sp.jar;C:\bea\wls90\admin\WEBLOG~1\server\lib\weblogic.jar;C:\
bea\wls90\admin\WEBLOG~1\server\lib\webservices.jar
-Dweblogic.system.BootIdentityFile=C:\bea\wls90\admin\user_projects
\domains\WLS90Domain\servers\AdminServer\security\boot.properties
-Dweblogic.nodemanager.ServiceEnabled=true weblogic.Server
```

ServerRole is Managed and ServerStartProgram is null

The following pattern matching applies:

- Command line begins with *<BEA_HOME>*, followed by 0 or more characters, followed by the string *java*, followed by 0 or more characters, followed by `weblogic.Name=<ManagedServerName>`, followed by space.
- Command line ends with `weblogic.Server`.
- Command line contains *<DomainDir>* followed by front slash or back slash.
- Command line contains `management.server=<AdminURL>` followed by space.

Example command line

```
C:\bea\wls90\admin\JROCKI~1\jre\bin\java
-Dweblogic.Name=ManagedServer01
-Djava.security.policy=C:\bea\wls90\admin\WEBLOG~1\server\lib\weblo
gic.policy -Dweblogic.management.server=http://wls90host:7001
"-Djava.library.path=C:\bea\wls90\admin\WEBLOG~1\server\bin;.;C:\WI
NDOWS\system32;C:\WINDOWS;C:\bea\wls90\admin\WEBLOG~1\server\native
\win\32;C:\bea\wls90\admin\WEBLOG~1\server\bin;C:\bea\wls90\admin\J
ROCKI~1\jre\bin;C:\bea\wls90\admin\JROCKI~1\bin;C:\bea\wls90\admin\
WEBLOG~1\server\native\win\32\oci920_8;c:\program
files\mks\mksnt;C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\
Wbem;C:\Program Files\VERITAS\VERITAS Object Bus\bin;C:\Program
Files\VERITAS\VERITAS Volume Manager 4.3\;C:\Program
Files\VERITAS\VRTSjre\AccessBridge;C:\Program
Files\VERITAS\Security\Authentication\bin;C:\Program
```

```
Files\VERITAS\VRTSPerl\bin;C:\Program
Files\VERITAS\comms\llt;C:\Program
Files\VERITAS\comms\gab;C:\Program Files\VERITAS\Cluster
server\bin;C:\Program Files\VERITAS\Cluster
server\bin\VCW;C:\Program Files\Microsoft SQL Server\80\Tools\BINN"
-Djava.class.path=.;C:\bea\wls90\admin\patch_weblogic901\profiles\d
efault\sys_manifest_classpath\weblogic_patch.jar;C:\bea\wls90\admin
\JROCKI~1\lib\tools.jar;C:\bea\wls90\admin\WEBLOG~1\server\lib\webl
ogic_sp.jar;C:\bea\wls90\admin\WEBLOG~1\server\lib\weblogic.jar;C:\
bea\wls90\admin\WEBLOG~1\server\lib\webservices.jar
-Dweblogic.system.BootIdentityFile=C:\bea\wls90\admin\user_projects
\domains\WLS90Domain\servers\ManagedServer01\data\nodemanager\boot.
properties -Dweblogic.nodemanager.ServiceEnabled=true
-Dweblogic.security.SSL.ignoreHostnameVerification=true
-Dweblogic.ReverseDNSAllowed=false weblogic.Server
```

Appendix C

Command line pattern matching for non Node Manager based configurations

This appendix includes the following topics:

- ServerRole is Administrative and ServerStartProgram is non-null
- ServerRole is Managed and ServerStartProgram is non-null

ServerRole is Administrative and ServerStartProgram is non-null

The following pattern matching applies:

- Command line begins with <*BEA_HOME*>, followed by 0 or more characters, followed by the string java, followed by 0 or more characters, followed by `weblogic.Name=<AdminServerName>`, followed by space.
- Command line ends with `weblogic.Server`.
- If <*DomainDir*> is non-null, command line contains <*DomainDir*> followed by front slash or back slash.

Example command line of Admin server started with startWebLogic.cmd

```
c:\bea\wls90\admin\jdk150~1\bin\java -client   -xms256m -xmx512m
```

```
-xx:compilethreshold=8000 -xx:permsize=32m  -xx:maxpermsize=128m
-xverify:none  -da -dplatform.home=c:\bea\wls90\admin\weblog~1
-dwls.home=c:\bea\wls90\admin\weblog~1\server
-dwli.home=c:\bea\wls90\admin\weblog~1\integration
-dweblogic.management.discover=true
-dweblogic.productionmodeenabled=  -dwlw.iterativedev=
-dwlw.testconsole= -dwlw.logerrorstoconsole=
-dweblogic.ext.dirs=c:\bea\wls90\admin\patch_weblogic901\profiles\d
efault\sysext_manifest_classpath -dweblogic.Name=AdminServer
-djava.security.policy=c:\bea\wls90\admin\weblog~1\server\lib\weblo
gic.policy  weblogic.Server
```

ServerRole is Managed and ServerStartProgram is non-null

The following pattern matching applies:

- Command line begins with <*BEA_HOME*>, followed by 0 or more characters, followed by the string java, followed by 0 or more characters, followed by `weblogic.Name=<ManagedServerName>`, followed by space.
- Command line ends with `weblogic.Server`.
- If <*DomainDir*> is non-null, command line contains <*DomainDir*> followed by front slash or back slash.
- Command line contains `management.server=<AdminURL>` followed by space.

Example command line of server started using startManagedWebLogic.cmd

```
C:\bea\wls90\admin\JDK150~1\bin\java -server   -Xms256m -Xmx512m
-XX:MaxPermSize=128m
-Dweblogic.security.SSL.trustedCAKeyStore="C:\bea\wls90\admin\weblo
gic90\server\lib\cacerts"   -da
-Dplatform.home=C:\bea\wls90\admin\WEBLOG~1
-Dwls.home=C:\bea\wls90\admin\WEBLOG~1\server
-Dwli.home=C:\bea\wls90\admin\WEBLOG~1\integration
-Dweblogic.management.discover=false
-Dweblogic.management.server=t3://wls90host:7001
-Dwlw.iterativeDev=false -Dwlw.testConsole=false
-Dwlw.logErrorsToConsole=
-Dweblogic.ext.dirs=C:\bea\wls90\admin\patch_weblogic901\profiles\d
efault\sysext_manifest_classpath -Dweblogic.Name=ManagedServer01
```

```
Djava.security.policy=C:\bea\wls90\admin\WEBLOG~1\server\lib\weblog
ic.policy  weblogic.Server
```

Index

A

agent
- installing in VCS environment 16
- removing in VCS environment;uninstalling in VCS environment 17
- resource configurations 14
- upgrading 18

agent attributes
- Admin URL 21
- AdminServerMaxWait 25
- BEA_HOME 21
- DomainDir 22
- DomainName 21
- ListenAddressPort 22
- MonitorProgram 26
- nmListenAddressPort 22
- nmType 22
- RequireAdminServer 26
- ResLogLevel 23
- SecondLevelMonitor 27
- ServerName 23
- ServerRole 24
- ServerStartProgram 27
- ServerStopProgram 27
- WL_HOME 23
- WLSPassword 24
- WLSUser 23

agent for WebLogic Server
- attributes 21

agent function 12
- clean 14
- monitor 13
- offline 12
- online 12

I

inspecting
- agent logs 38
- temporary log files 38

S

sample configurations
- agent type definition 40
- sample service group 41
- VCS environment 54

service group dependencies 53

W

WebLogic scripts
- editing 31
- using 30