Veritas™ High Availability Agent for WebSphere MQ Installation and Configuration Guide

AIX, HP-UX, Linux, Solaris

5.1

Veritas High Availability Agent for WebSphere MQ 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: 5.1.10.0

Document version: 5.1.10.0.0

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 Web site 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 Web site.

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

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

Introducing the Veritas High Availability Agent for WebSphere MQ

This chapter includes the following topics:

- About the Veritas agent for WebSphere MQ
- What's new in this agent
- Supported software
- WebSphere MQ agent functions

About the Veritas agent for WebSphere MQ

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

The Veritas agent for WebSphere MQ provides high availability for all WebSphere MQ Queue Managers in a cluster. The agent can bring a specific WebSphere MQ Queue Manager online and monitor the state of the Queue Manager. The agent can also detect failures and shut down the Queue Manager in case of a failure.

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

http://www.symantec.com/business/support/index?page=content&id=TECH46455

What's new in this agent

The enhancement in this release of WebSphere MQ agent is as follows:

■ Fixed an issue with the handling of the PID files that the agent maintains for its Fast First Level Monitor (FFLM) feature. With this modification, the offline function removes PID files that are no longer required. As a result, the monitor function no longer processes stale PID files for FFLM.

■ Added support for AIX 7.1 and Red Hat Enterprise Linux 6.0.

For information on the changes introduced in the previous releases:

See "Changes introduced in previous releases" on page 75.

Supported software

The Veritas agent for WebSphere MQ supports the following software versions:

Veritas Cluster Server	■ AIX—VCS 4.0, 5.0, 5.1 ■ HP-UX—VCS 4.1, 5.0, 5.1 ■ Linux—VCS 4.0, 4.1, 5.0, 5.1 ■ Solaris—VCS 4.0, 4.1, 5.0, 5.1 and all intermediate Maintenance Packs of these releases.
Veritas Cluster Server One	VCS One 2.0, 5.0 on AIX, HP-UX, Linux, and Solaris
ACC Library	5.1 and later Review the ACC Library version for i18n support. See "Prerequisites for enabling i18n support" on page 18.
Operating Systems	■ AIX 5.1, 5.2, 5.3, 6.1, 7.1 on pSeries ■ HP-UX 11i v2, 11i v3 on Itanium and PA-RISC ■ Red Hat Enterprise Linux (RHEL) 3, 4, 5 on Intel and ppc64 RHEL 6.0 on Intel ■ SUSE Linux Enterprise Server 9, 10, 11 on Intel and ppc64 ■ Solaris 8, 9, 10 on SPARC and x86 **Note:** The agent supports zones on Solaris in both VCS and VCS One environments.

WebSphere MQ Queue Manager	5.3, 6.0, 7.0
	and all intermediate minor versions of these releases.

Support matrix for IMF and in-depth monitoring

Depending on your version of Veritas Cluster Server (VCS) and the WebSphere MQ agent, the following features and functionality are supported.

Table 1-1

VCS and agent version	IMF capability	SecondLevelMonitor attribute	LevelTwoMonitorFreq attribute
VCS 5.1 SP1 or later with WebSphere MQ agent 5.1.9.0 or later	Yes	No	Yes
VCS 5.1 SP1 with WebSphere MQ agent 5.1.8.0 or earlier	No	Yes	No
VCS 5.1 or earlier with WebSphere MQ agent 5.1.9.0 or later	No	Yes	No
VCS 5.1 or earlier with WebSphere MQ agent 5.1.8.0 or earlier	No	Yes	No

WebSphere MQ agent functions

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

Online

The online function performs the following tasks:

■ Verifies that the WebSphere MQ instance is not already online. If the instance is online, the online operation exits immediately.

■ If the partial set of WebSphere MQ processes are running, the agent performs a process level clean up before starting the queue manager.

- Uses an IBM provided start script to start the WebSphere MQ using the name of the Queue Manager.
  You can also configure the online function to source a shell script or a program that the EnvFile attribute specifies. This script or program ensures that the required shell environment variables are properly set before executing the start script.

- Ensures that the WebSphere MQ Queue Manager is up and running successfully. The operation uses the wait period that the OnlineTimeout attribute specifies, to enable the Queue Manager to initialize fully before allowing the monitor function to probe the resource.

Offline

The offline function performs the following tasks:

- Verifies that the WebSphere MQ instance is not already offline. If the instance is offline, the offline operation exits immediately.

- Uses an IBM provided stop script to stop the WebSphere MQ Queue Manager using the name of the Queue Manager.
  You can also configure the offline function to source a shell script or a program that the EnvFile attribute specifies. This script or program ensures that the required shell environment variables are properly set before executing the stop script.

- Ensures that the WebSphere MQ Queue Manager is given enough time to go offline successfully. The operation uses a wait period that the OfflineTimeout attribute specifies, to allow the WebSphere MQ Queue Manager to complete the offline sequence before allowing further probing of the resource.
  If the processes are found running even after the wait period, then these processes are killed.

Monitor

The monitor function monitors the states of the WebSphere MQ Queue Managers running on all nodes within the cluster.

The monitor function can monitor the following WebSphereMQ Queue Manager components:

- Queue Manager

- Channel Initiator

- Command Server (If the CommandServer attribute is set to 1)

The function performs the following tasks:

- The first level check searches for all system processes that must be running for a WebSphere MQ Queue Manager. If the first level check does not find these processes running on the node, the check exits immediately, and reports the Queue Manager as offline.

  The agent for WebSphere MQ also supports Intelligent Monitoring Framework (IMF) in the first level check. IMF enables intelligent resource monitoring. The agent for WebSphere MQ is IMF-aware and uses the asynchronous monitoring framework (AMF) kernel driver for resource state change notifications. See "How the agent supports intelligent resource monitoring" on page 44.

  You can use the MonitorFreq key of the IMF attribute to specify the frequency at which the agent invokes the monitor function. See "MonitorFreq" on page 46.

- The second level check, if configured, determines the status of the WebSphereMQ Queue Manager.

  The second level check executes the `runmqsc` command and pings the Queue Manager to see if the manager is up and running. This check ensures that the processes are truly available for MQ Queue processing.

Note: The attribute used to configure the second level check and its frequency depends on the software versions of VCS and WebSphere MQ agent you have installed: For VCS 5.1 SP1 with WebSphereMQ agent version 5.1.9.0, use the LevelTwoMonitorFreq attribute. For VCS 5.1 or earlier with WebSphereMQ agent 5.1.8.0 or earlier, use the SecondLevelMonitor attribute

---

- Depending upon the MonitorProgram attribute, the monitor function can perform a customized check using a user-supplied monitoring utility. For details about executing a custom monitor program:

  See "Executing a customized monitoring program" on page 41.

- When the WebSphere MQ resource is offline and the agent detects the queue manager processes as running, but the second level monitor check fails, the agent cleans these processes.

Clean

In case of a failure or after an unsuccessful attempt to online or offline WebSphere MQ Queue Manager, the clean function removes any Queue Manager processes remaining in the system.

The function performs the following tasks:

- Attempts to gracefully shut down the WebSphere MQ Queue Manager.

- If a graceful shutdown fails, the clean function looks for all the processes running for the WebSphere MQ Queue Manager, and cleans the processes.

- The clean function executes the IBM supplied utility, amqiclen to clean the IPC resources that are associated with the WebSphere MQ Queue Manager.

- If the CommandServer attribute is set to 1 for WebSphere MQ version 6.0 or later, the clean function kills the Command Server processes associated with the WebSphere MQ Queue Manager.

Note: For information about the additional functions of the agent for WebSphere MQ when IMF is enabled: See "Agent functions for the IMF functionality" on page 45.

Installing, upgrading, and removing the agent for WebSphere MQ

This chapter includes the following topics:

- Before you install the Veritas agent for WebSphere MQ
- About the ACC library
- Installing the ACC library
- Installing the agent in a VCS environment
- Installing the agent in VCS One environment
- Removing the agent in a VCS environment
- Removing the agent in VCS One environment
- Removing the ACC library
- Upgrading the agent in a VCS environment
- Upgrading the agent in a VCS One environment

Before you install the Veritas agent for WebSphere MQ

You must install the Veritas agent for WebSphere MQ on all the systems that will host WebSphere MQ Queue Manager service groups.

Ensure that you meet the following prerequisites to install the agent for WebSphere MQ.

For VCS, do the following:

■ Install and configure Veritas Cluster Server.
For more information on installing and configuring Veritas Cluster Server, refer to the *Veritas Cluster Server Installation Guide*.

■ Remove any previous version of this agent.
To remove the agent,
See "Removing the agent in a VCS environment" on page 26.

■ Install the latest version of ACC Library.
To install or update the ACC Library package, locate the library and related documentation in the agentpack tarball:
See "Installing the ACC library" on page 19.

Note: All non-global zones must be booted and in a running state at the time of package installation or un-installation. If the non-global zones are not booted, you may need to reinstall the package manually after booting the non-global zones.

For VCS One, do the following:

■ Install and configure Veritas Cluster Server One.
For more information on installing and configuring Veritas Cluster Server One, refer to the *Veritas Cluster Server One Installation Guide*.

■ Remove any previous version of this agent.
To remove the agent,
See "Removing the agent in VCS One environment" on page 28.

Prerequisites for enabling i18n support

Perform the following steps to enable i18n support to the agent:

■ Install ACCLib version 5.1.2.0 or later.
See "Installing the ACC library" on page 19.

■ For VCS 5.0 and earlier releases, copy the latest ag_i18n_inc.pm module from the following location on the agent pack disc.

Note: Review the readme.txt for instructions to copy this module.

VCS 5.0	cd1/*platform*/*arch_dist*/vcs/application/i18n_support/5.0
VCS 4.1	cd1/*platform*/*arch_dist*/vcs/application/i18n_support/4.1
VCS 4.0	cd1/*platform*/*arch_dist*/vcs/application/i18n_support/4.0

where *arch_dist* takes the following values:
'sol_sparc' for Solaris SPARC
'sol_x64' for Solaris x64
'generic' for HP-UX and Linux

Note: *arch_dist* is not applicable to AIX.

About the ACC library

The operations of a VCS agent depend on a set of Perl modules known as the ACC library. The library must be installed on each system in the cluster that runs the agent. The ACC library contains common, reusable functions that perform tasks, such as process identification, logging, and system calls.

Instructions to install or remove the ACC library on a single system in the cluster are given in the following sections. The instructions assume that the agent's tar file has already been extracted.

Installing the ACC library

Install the ACC library on each system in the cluster that runs an agent that depends on the ACC library.

To install the ACC library

1 Log in as superuser.

2 Download ACC Library.

   You can download either the complete Agent Pack tarball or the individual ACCLib tarball from the Symantec Operations Readiness Tools (SORT) site (https://sort.symantec.com/agents).

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

AIX   *cd1*/aix/vcs/application/acc_library/*version*_library/pkgs

HP-UX   *cd1*/hpux/generic/vcs/application/acc_library/*version*_library/pkgs

Linux   *cd1*/linux/generic/vcs/application/acc_library/*version*_library/rpms

Solaris   *cd1*/solaris/*dist_arch*/vcs/application/acc_library/*version*_library/pkgs

where *dist_arch* is sol_sparc or sol_x64.

4 If you downloaded the individual ACCLib tarball, navigate to the pkgs directory (for AIX, HP-UX, and Solaris), or rpms directory (for Linux).

5 Install the package. Enter **Yes** if asked to confirm overwriting of files in the existing package.

AIX   `# installp -ac -d VRTSacclib.bff VRTSacclib`

HP-UX   `# swinstall -s ‘pwd‘ VRTSacclib`

Linux   `# rpm -i \`
                `VRTSacclib-VersionNumber-GA_GENERIC.noarch.rpm`

Solaris   `# pkgadd -d VRTSacclib.pkg`

Installing the agent in a VCS environment

Install the agent for WebSphere MQ on each node in the cluster.

Note: The agent package VRTSmq6 includes the Veritas agents for WebSphere MQ and WebSphere MQ FTE. So, the following procedure to install the agent for WebSphere MQ installs the agent for WebSphere MQ FTE also.

To install the agent in a VCS environment

1 Download the agent from the Symantec Operations Readiness Tools (SORT) site: https://sort.symantec.com/agents.

You can download either the complete Agent Pack tarball or an individual agent tarball.

2 Uncompress the file to a temporary location, say /tmp.

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

AIX
```
cd1/aix/vcs/application/webspheremq_agent/
vcs_version/version_agent/pkgs
```

HP-UX
```
cd1/hpux/generic/vcs/application/webspheremq_agent/
vcs_version/version_agent/pkgs
```

Linux
```
cd1/linux/generic/vcs/application/webspheremq_agent/
vcs_version/version_agent/rpms
```

Solaris
```
cd1/solaris/dist_arch/vcs/application/webspheremq_agent/
vcs_version/version_agent/pkgs
```

where, *dist_arch* is sol_x64 or sol_sparc

If you downloaded the individual agent tarball, navigate to the pkgs directory (for AIX, HP-UX, and Solaris), or rpms directory (for Linux).

4 Log in as superuser.

5 Install the package.

AIX
```
# installp -ac -d VRTSmq6.rte.bff VRTSmq6.rte
```

HP-UX
```
# swinstall -s 'pwd' VRTSmq6
```

Linux
```
# rpm -ihv \
VRTSmq6-AgentVersion-GA_GENERIC.noarch.rpm
```

Solaris
```
# pkgadd -d . VRTSmq6
```

6 After installing the agent package, you must import the agent type configuration file. See "Importing the agent types files in a VCS environment" on page 35.

Installing the agent in VCS One environment

You must install the agent for on all the client systems of the VCS One cluster that will host the service group. You can install the agent for using the installagpack program or using the command line interface (CLI).

The installation of the agent packs involves the following phases:

Installing the agent packages	See "Installing the agent packages using the installer" on page 22.
Adding the agent resource type definitions	See "Adding the agent resource type definitions to the Policy Master Server on UNIX" on page 24.
	See "Adding the agent resource type definitions to the Policy Master Server on Windows" on page 26.

Note: The installagpack program supports only the -addtypes, -rmtypes, -responsefile, and -rsh options. Symantec recommends that you do not use any of the other options from the installagpack command help output.

Installing the agent packages using the installer

You can install the agent packages on one or more client systems of a specific platform type.

Note: To install the VCS One client for managing VMware ESX Servers, download the tar ball for Red Hat Enterprise Linux 4 (RHEL 4) x86 (32-bit) or RHEL 5 x86_64

Perform the following steps to install the agent packages using the installer

1   On the Policy Master system, download the complete Agent Pack tarball or the individual agent tarball from the Symantec Operations Readiness Tools (SORT) site: https://sort.symantec.com/agents.

2   Uncompress the file to a temporary location, say /tmp.

3   If you downloaded the complete Agent Pack tarball, navigate to the following directory containing the installer for the VCS One agents, for the platform running in your environment:

AIX	cd1/aix/vcsone/*vcsone_version*
HP-UX	cd1/hpux/hpux*os_version*/vcsone/*vcsone_version*
	Where *os_version* is the HP-UX version.
Linux	cd1/linux/*dist_arch*/vcsone/*vcsone_version*
	Where *dist* is the Linux distribution and *arch* is the architecture.
Solaris	cd1/solaris/*dist_arch*/vcsone/*vcsone_version*
	Where, *dist_arch* is 'sol_sparc' or 'sol_x64'.

4   Enter the following command to start the agent pack installation:

    ```
    # ./installagpack [-rsh]
    ```

    You can use the -rsh option if rsh and rcp are used for communication
    between systems instead of the default ssh and scp. This option requires that
    systems be preconfigured such that the rsh commands between systems
    execute without prompting for passwords or confirmations.

5   Enter the name of the client systems where you want to install the agents.

6   Choose whether to install all the agents or any specific agent. Follow the
    installer prompt to specify your option.

7   Review the output as the installation program installs the agent packages.

    You can view installation logs in the /var/VRTS/install/logs directory.

Installing the agent package using the CLI

You can install the desired agent package using the CLI, on one or more client
systems of a specific platform type.

Perform the following steps to install the agent packages using CLI

1   On the Policy Master system, download the complete Agent Pack tarball or
    the individual agent tarball from the Symantec Operations Readiness Tools
    (SORT) site: https://sort.symantec.com/agents.

2   Uncompress the file to a temporary location, say /tmp.

3 If you downloaded the complete Agent Pack tarball, navigate to the following directory containing the installer for the VCS One agents, for the platform running in your environment:

AIX   # **cd1/aix/vcsone/vcsone_version/pkgs**

HP-UX   # **cd1/hpux/hpux*os_version*/vcsone/vcsone_version/depot**

Linux   # **cd1/linux/*dist_arch*/vcsone/vcsone_version/rpms**

Where, *dist* is the Linux distribution and *arch* is the architecture

Solaris   # **cd1/solaris/*dist_arch*/vcsone/vcsone_version/pkgs**

Where *dist_arch* is 'sol_sparc' or 'sol_x64'

4 Type the following command on each client system to install the agent. Answer the prompt accordingly:

AIX   # **installp -ac -d . VRTSmq6.rte**

HP-UX   # **swinstall -s `pwd` VRTSmq6**

Linux   # **rpm -ivh VRTSmq6_rpm_filename**

Solaris   # **pkgadd -d . VRTSmq6**

Adding the agent resource type definitions to the Policy Master Server on UNIX

You must add the agent resource type definitions to the Policy Master database configuration. You can perform this task from any client system in the VCS One cluster.

Note: You must add the agent resource type definitions only one time per platform type.

To add the agent resource types to the policy master database configuration

1   Set up RSH or SSH communications between the client system and the policy master system.

    For information on configuring SSH for remote communication, refer to the *Veritas Cluster Server One Installation Guide*.

2   Make sure that the PM daemon is running.

    ```
    # /opt/VRTSvcsone/bin/haclus -display
    ```

    The output should show ClusterState is RUNNING.

3   Access the temporary location where you downloaded the tar ball and depending on the platform type, navigate to the directory containing the agent installer:

AIX	cd1/aix/vcsone/*vcsone_version*
HP-UX	cd1/hpux/hpux*os_version*/vcsone/*vcsone_version*
	Where *os_version* is the HP-UX version.
Linux	cd1/linux/*dist_arch*/vcsone/*vcsone_version*
	Where *dist* is the Linux distribution and *arch* is the architecture.
Solaris	cd1/solaris/*dist_arch*/vcsone/*vcsone_version*
	Where *dist_arch* is the sol_sparc or sol_x64.

4   Enter the command to start the agent pack installer for adding resource types to the Policy Master configuration database. Use the -addtypes option:

    ```
    # ./installagpack -addtypes
    ```

5   When the installer prompts, enter the virtual IP address of the Policy Master.

6   Review the output as the installer verifies communication with the Policy Master system.

7   Choose whether to add the type definitions for all the agents or for specific agents. Follow the installer prompts to add the type definitions.

8   Review the output as the installer adds the agent types to the PM database configuration and copies the appropriates types.xml files to the PM system.

    You can view installation logs in the /var/VRTS/install/logs directory.

Adding the agent resource type definitions to the Policy Master Server on Windows

After you have installed the agent package, you must add the agent resource type definitions to the Policy Master database configuration. You must perform this task from the Policy Master Server.

Note: You must add the agent resource type definitions only one time per platform type.

To add the agent resource types to the Policy Master Server on Windows, perform the following steps from the Policy Master Server command prompt

1   Create a temporary directory on the Policy Master Server, to add the type definitions.

    C:\> mkdir addtypes_tmp

2   Change your working directory to the temporary directory created in step 1.

    C:\> chdir addtypes_tmp

3   Copy the agent's type xml file in to the temporary directory.

4   Convert this type xml file into type cmd file.

    C:\addtypes_tmp> haconf -xmltocmd *type_xml_filename*.xml

5   Rename the *type_xml_filename*.xml.cmd file to *type_xml_filename*.bat

6   Run the batch file.

    C:\addtypes_tmp> *type_xml_filename*.bat >log.txt 2>&1

7   Review the log.txt file for any errors.

8   Verify whether the type has been successfully added to the Policy Master Server.

    C:\addtypes_tmp> hatype -list -platform *platform_name*

Removing the agent in a VCS environment

You must uninstall the agent for WebSphere MQ from a cluster while the cluster is active.

Warning: The agent package VRTSmq6 includes the Veritas agents for WebSphere MQ and WebSphere MQ FTE. So, the following procedure to remove the agent for WebSphere MQ removes the agent for WebSphere MQ FTE also.

To uninstall the agent in a VCS environment

1   Log in as a superuser.

2   Set the cluster configuration mode to read/write by typing the following command from any node in the cluster:

```
# haconf -makerw
```

3   Remove all WebSphere MQ Queue Manager resources from the cluster. Use the following command to verify that all resources have been removed:

```
# hares -list Type=WebSphereMQ6
```

4   Remove the agent type from the cluster configuration by typing the following command from any node in the cluster:

```
# hatype -delete WebSphereMQ6
```

Removing the agent's type file from the cluster removes the include statement for the agent from the main.cf file, but the agent's type file is not removed from the cluster configuration directory. You can remove the agent's type file later from the cluster configuration directory.

5   Save these changes. Then set the cluster configuration mode to read-only by typing the following command from any node in the cluster:

```
# haconf -dump -makero
```

6   Use the platform's native software management program to remove the agent for WebSphere MQ from each node in the cluster.

Execute the following command to uninstall the agent:

AIX	# installp -u VRTSmq6.rte
HP-UX	# swremove VRTSmq6
Linux	# rpm -e VRTSmq6
Solaris	# pkgrm VRTSmq6

Removing the agent in VCS One environment

Removing the agent package involves removing the agent files from each client system where it was installed.

You can remove the packages using the agent pack installer or the command line.

See "Removing the agent packages using the installer" on page 28.

See "Removing the agent package using CLI" on page 29.

After removing the agent packages you can remove the agent type definition from the Policy Master system.

See "Removing the agent type definition from the Policy Master system on UNIX" on page 30.

See "Removing the agent type definition from the Policy Master sytem on Windows" on page 30.

Removing the agent packages using the installer

You can remove all the agent packages or the desired agent package using the uninstallagpack program.

Note: The uninstallagpack program supports only the -responsefile and -rsh options. Symantec recommends that you do not use any of the other options from the `uninstallagpack` command help output.

To remove the agent packages from the client systems

1   Freeze the service groups that hosts the application, on the system from which you want to remove the agent package.

    # hagrp -freeze <*groupname*>

2   Stop the agent on all client systems before you remove the agent package from the system.

    # haagent -stop -notransition <*AgentName*> -sys <*system_name*>

3   Ensure that the agent operations are stopped on all the cluster systems.

    # haagent -display <*AgentName*>

4 Access the temporary location where you downloaded the Agent Pack and navigate to the directory containing the package for the platform running in your environment:

AIX	`cd1/aix/vcsone/`*`vcsone_version`*
HP-UX	`cd1/hpux/hpux`*`os_version`*`/vcsone/`*`vcsone_version`* Where *os_version* is the HP-UX version.
Linux	`cd1/linux/`*`dist_arch`*`/vcsone/`*`vcsone_version`* Where *dist* is the Linux distribution and *arch* is the architecture.
Solaris	`cd1/solaris/`*`dist_arch`*`/vcsone/`*`vcsone_version`* Where *dist_arch* is the sol_sparc or sol_x64.

5 Start the uninstallagpack program.

```
# ./uninstallagpack [-rsh]
```

6 Enter the name of the client systems on which you want to uninstall the agent pack. The names must be separated by spaces.

7 Choose whether to remove all the agent packages or a specific agent package. Follow the installer prompt to remove the agent package.

8 Review the output as the program verifies the agent pack that you installed and removes the agent packages.

You can view logs in the /var/VRTS/install/logs directory.

Removing the agent package using CLI

You can remove a desired agent package using the CLI.

Note: You must remove this agent package from each client system in the cluster.

To remove the agent for from a client system

◆ Type the following command on each client system to remove the agent. Answer prompts accordingly:

AIX   # `installp -u VRTSmq6`

HP-UX   # `swremove VRTSmq6`

Linux   # `rpm -e VRTSmq6`

Solaris   # `pkgrm VRTSmq6`

Removing the agent type definition from the Policy Master system on UNIX

After you remove the agent packages, you can remove the agent type definitions for agents you removed, from the Policy Master system.

To remove the agent type definition from the Policy Master system on UNIX

1 Navigate to the following directory on the client system.

   # `cd /opt/VRTS/install`

2 Run the following command to remove the agent type definition from the Policy Master system:

   # `./installagpack -rmtypes`

3 When the installer prompts, enter the virtual IP address of the Policy Master.

4 Choose whether to remove the type definitions for all the agents or for specific agents. Follow the installer prompts to remove the type definitions.

   You can view logs in the /var/VRTS/install/logs directory.

Removing the agent type definition from the Policy Master sytem on Windows

After you remove the agent packages, you can remove the agent type definitions for agents you removed, from the Policy Master system.

To remove the agent type definition from the Policy Master system on Windows

◆ Run the following command from the Policy Master Server command prompt.

C:\> hatype -delete *agentname*_i.e._*typename* -platform *platformname*

Removing the ACC library

Perform the following steps to remove the ACC library.

To remove the ACC library

1 Ensure that all agents that use ACC library are removed.

2 Run the following command to remove the ACC library package.

AIX	# installp -u VRTSacclib
HP-UX	# swremove VRTSacclib
Linux	# rpm -e VRTSacclib
Solaris	# pkgrm VRTSacclib

Upgrading the agent in a VCS environment

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

Note: The agent package VRTSmq6 includes the Veritas agents for WebSphere MQ and WebSphere MQ FTE. Hence, both the agents will be upgraded as the result of upgrading the package. So, perform the following steps for the agent for WebSphere MQ FTE as well.

To upgrade the agent in a VCS environment

1 Persistently freeze the service groups that host the application.

    # hagrp -freeze *GroupName* -persistent

2 Stop the cluster services forcibly.

    # hastop -all -force

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

    # ps -ef | grep WebSphereMQ6

4 Uninstall the agent package from all the nodes. Use the platform's native software management program to remove the agent for WebSphere MQ from each node in the cluster.

Execute the following command to uninstall the agent:

AIX	# installp -u VRTSmq6.rte
HP-UX	# swremove VRTSmq6
Linux	# rpm -e VRTSmq6
Solaris	# pkgrm VRTSmq6

5 Install the new agent on all the nodes.

See

6 Copy the new WebSphereMQ6Types.cf file from the agent's conf directory, to the VCS conf directory /etc/VRTSvcs/conf/config.

VCS version	Operating system	Agent types file
VCS 4.x	■ AIX ■ HP-UX ■ Linux ■ Solaris	/etc/VRTSvcs/conf/sample_WebSphereMQ6/ WebSphereMQ6Types.cf
VCS 5.x	■ AIX ■ HP-UX ■ Linux	/etc/VRTSagents/ha/conf/WebSphereMQ6/ WebSphereMQ6Types.cf
VCS 5.0	■ Solaris SPARC and x64	/etc/VRTSagents/ha/conf/WebSphereMQ6/ WebSphereMQ6Types50.cf
VCS 5.1	■ Solaris SPARC and x64	/etc/VRTSagents/ha/conf/WebSphereMQ6/ WebSphereMQ6Types51.cf

Note: If you upgraded the VCS version to VCS 5.1 SP1 and the WebSphere MQ agent version to 5.1.9.0 (or later), and if you had enabled detail monitoring in the previous version, then do the following: Set the value of the LevelTwoMonitorFreq attribute to the same value as that of the SecondLevelMonitor attribute.

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

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

8 Start the cluster services.

```
# hastart
```

9 Start the agent on all nodes, if not started.

```
# haagent -start WebSphereMQ6 -sys SystemName
```

10 Unfreeze the service groups once all the resources come to an online steady state.

```
# hagrp -unfreeze GroupName -persistent
```

Upgrading the agent in a VCS One environment

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

> **Note:** The agent package VRTSmq6 includes the Veritas agents for WebSphere MQ and WebSphere MQ FTE. Hence, both the agents will be upgraded as the result of upgrading the package. So, perform the following steps for the agent for WebSphere MQ FTE as well.

To upgrade the agent with minimal disruption, in a VCS One environment

1 Freeze service groups that hosts the application.

```
# hagrp -freeze -propagate GroupName
```

2 Stop the clients forcibly. Execute the following command from the Policy Master.

```
# hastop -client -sys SystemName -force
```

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

```
# ps -ef | grep WebSphereMQ6
```

4 Uninstall the agent package from all the nodes. Type the following command on each client system to remove the agent. Answer prompts accordingly:

AIX   # **installp -u VRTSmq6**

HP-UX   # **swremove VRTSmq6**

Linux   # **rpm -e VRTSmq6**

Solaris   # **pkgrm VRTSmq6**

5 Install the new agent on all the nodes in the cluster.

See "Installing the agent in VCS One environment" on page 21.

6 Add the agent types, using the installagpack program.

See "Adding the agent resource type definitions to the Policy Master Server on UNIX" on page 24.

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

8 Start the clients.

hastart -client

9 Start the agent on all nodes, if not started.

haagent -start WebSphereMQ6 -sys *SystemName*

10 Unfreeze the service groups.

hagrp -unfreeze -propagate *GroupName*

Configuring the agent for WebSphere MQ

This chapter includes the following topics:

- About configuring the Veritas agent for WebSphere MQ
- Importing the agent types files in a VCS environment
- WebSphere MQ agent attributes
- Executing a customized monitoring program

About configuring the Veritas agent for WebSphere MQ

After installing the Veritas agent for WebSphere MQ, you must import the agent type configuration file. After importing this file, you can create and configure WebSphere MQ Queue Manager resources. Before you configure a resource, review the attributes table that describes the resource type and its attributes.

To view the sample agent type definition and service groups configuration.

See "About sample configurations for the agent for WebSphere MQ" on page 67.

Importing the agent types files in a VCS environment

To use the agent for WebSphere MQ, you must import the agent types file into the cluster.

To import the agent types file using the Veritas Cluster Server graphical user interface

1   Start the Veritas Cluster Manager and connect to the cluster on which the agent is installed.

2   Click **File > Import Types**.

3   In the Import Types dialog box, select the following file:

VCS version	Operating System	Agent types file
VCS 4.x	■ AIX ■ HP-UX ■ Linux ■ Solaris	/etc/VRTSvcs/conf/sample_WebSphereMQ6/ WebSphereMQ6Types.cf
VCS 5.x	■ AIX ■ HP-UX ■ Linux	/etc/VRTSagents/ha/conf/WebSphereMQ6/ WebSphereMQ6Types.cf
VCS 5.0	■ Solaris SPARC and x64	/etc/VRTSagents/ha/conf/WebSphereMQ6/ WebSphereMQ6Types50.cf
VCS 5.1	■ Solaris SPARC and x64	/etc/VRTSagents/ha/conf/WebSphereMQ6/ WebSphereMQ6Types51.cf

4   Click **Import**.

5   Save the VCS configuration.

The WebSphere MQ Queue Manager agent type is now imported to the VCS engine.

You can now create WebSphere MQ Queue Manager resources. For additional information about using the VCS GUI, refer to the *Veritas Cluster Server User's Guide*.

To import the agent types file using the Veritas Cluster Server command line interface (CLI), perform the following steps.

1   Log on to any one of the systems in the cluster as the superuser.

2   Create a temporary directory.

```
# mkdir ./temp

# cd ./temp
```

3   Copy the sample file Types.cf.

VCS 4.x	■ AIX ■ HP-UX ■ Linux ■ Solaris	`/etc/VRTSvcs/conf/sample_WebSphereMQ6/` `WebSphereMQ6Types.cf`
VCS 5.x	■ AIX ■ HP-UX ■ Linux	`/etc/VRTSagents/ha/conf/WebSphereMQ6/` `WebSphereMQ6Types.cf`

VCS 5.0	■ Solaris
	  SPARC
	  and x64
		`/etc/VRTSagents/ha/conf/WebSphereMQ6/`
		`WebSphereMQ6Types50.cf`

VCS 5.1	■ Solaris
	  SPARC
	  and x64
		`/etc/VRTSagents/ha/conf/WebSphereMQ6/`
		`WebSphereMQ6Types51.cf`

4 Create a dummy main.cf file:

```
# echo 'include "WebSphereMQ6Types.cf"' > main.cf
```

5 Create the WebSphere MQ resource type as follows:

```
# hacf -verify .

# haconf -makerw

# sh main.cmd

# haconf -dump
```

The WebSphere MQ Queue Manager agent type is now imported to the VCS engine.

You can now create WebSphere MQ Queue Manager resources. For additional information about using the VCS CLI, refer to the *Veritas Cluster Server User's Guide*.

WebSphere MQ agent attributes

Refer to the required and optional attributes while configuring the agent for WebSphere MQ Queue Manager.

Table 3-1 shows the required attributes for configuring a WebSphere MQ Queue Manager.

Table 3-1   Required attributes

Required attributes	Description
CommandServer	Decides whether the monitor function must monitor the command server process. This attribute is applicable for WebSphere version 6.0 and later.
	If this attribute is set to 1, the agent for WebSphere MQ monitors the command server process, amqpcsea. If this process faults, the agent for WebSphere MQ restarts the process.
	If you set this attribute to 0, the agent for WebSphere MQ does not monitor the amqpcsea process.
	Type and dimension: Boolean-scalar
	Default: 0
	Example: 1
MQUser	UNIX user name of the owner of the WebSphere MQ directories and executables. The agent functions use this name to execute all WebSphere MQ commands. This user name also owns the WebSphere MQ processes.
	This user name does not have to be unique within a cluster. The login shell for this user must be Bourne, Korn, or C-shell.
	Type and dimension: string-scalar
	Default: mqm
	Example: mqusr1
MQVer	Version of the WebSphere MQ Queue Manager. Valid values are 5.3, 6.0, and 7.0.
	Type and dimension: string-scalar
	Default: 6.0
	Example: 7.0
QueueManager	Name of the WebSphere MQ Queue Manager that the cluster server manages.
	You must uniquely define this attribute for each Queue Manager within the cluster. This attribute also uniquely identifies the processes running for a specific WebSphere MQ Queue Manager.
	Type and dimension: string-scalar
	Default: ""
	Example: venus.queue.manager

Table 3-1   Required attributes *(continued)*

Required attributes	Description
ResLogLevel	The logging detail performed by the agent for the resource. Valid values are:
	ERROR: Only logs error messages.
	WARN : Logs above plus warning messages.
	INFO: Logs above plus informational messages.
	TRACE: Logs above plus trace messages. TRACE is very verbose and should only be used during initial configuration or for troubleshooting and diagnostic functions.
	Type and dimension: string-scalar
	Default: INFO
	Example: TRACE

Table 3-2 shows the optional attributes for configuring a WebSphere MQ Queue Manager.

Table 3-2   Optional attributes

Optional attribute	Description
EnvFile	The complete path of the file name to source to set the environment prior to executing WebSphere MQ programs. Symantec recommends storing the file on the shared disk. This ensures that the same file is available on each failover node. Specifying this attribute is optional. The shell environments supported are ksh, sh, and csh.
	Type and dimension: string-scalar
	Default: ""
	Example: /MQ/setEnv.sh
MonitorProgram	Absolute path name of an external, user-supplied monitor executable. For information about setting this attribute:
	See "Executing a customized monitoring program" on page 41.
	Type and dimension: string-scalar
	Default: ""
	Example 1: /ibm/mq/myMonitor.sh
	Example 2: /ibm/mq/myMonitor.sh arg1 arg2

	Table 3-2	Optional attributes *(continued)*

Optional attribute	Description
SecondLevelMonitor	Used to enable second-level monitoring. Second-level monitoring is a deeper, more thorough state check of the WebSphere MQ Queue Manager. The numeric value specifies how often the monitoring routines must run. 0 means never run the second-level monitoring routines, 1 means run routines every monitor interval, 2 means run routines every second monitor interval, and so on.
	Note: Exercise caution while setting SecondLevelMonitor to large numbers. For example, if the MonitorInterval is set to 60 seconds and the SecondLevelMonitor is set to 100, then the runmqsc command is executed every 100 minutes, which may not be as often as intended. For maximum flexibility, no upper limit is defined for SecondLevelMonitor.
	Note: The SecondLevelMonitor attribute is applicable to VCS versions earlier than VCS 5.1 SP1 with WebSphereMQ agent versions earlier than 5.1.9.0. From VCS version 5.1 SP1 with WebSphere MQ agent version 5.1.9.0 onwards, the SecondLevelMonitor attribute of the WebSphereMQ agent is deprecated. Instead, a resource type level attribute LevelTwoMonitorFreq should be used to specify the frequency of in-depth monitoring.
	Type and dimension: integer-scalar
	Default: 0
	Example: 1
LevelTwoMonitorFreq	Specifies the frequency at which the agent for this resource type must perform second-level or detailed monitoring. You can also override the value of this attribute at the resource level.
	The value indicates the number of monitor cycles after which the agent will monitor the WebSphere MQ queue manager in detail. For example, the value 5 indicates that the agent will monitor the WebSphere MQ queue manager in detail after every five online monitor intervals.
	Note: This attribute is applicable to VCS version 5.1 SP1 with WebSphere MQ agent version 5.1.9.0 or later. If the VCS version is earlier than VCS 5.1 SP1 and the WebSphere MQ agent version is earlier than 5.1.9.0, the SecondLevelMonitor attribute should be used.
	If you upgraded the VCS version to VCS 5.1 SP1 and the WebSphereMQ agent version to 5.1.9.0 (or later), and if you had enabled detail monitoring in the previous version, then do the following:
	■ Set the value of the LevelTwoMonitorFreq attribute to the same value as that of the SecondLevelMonitor attribute.
	Type and dimension: integer-scalar
	Default: 0

Table 3-2   Optional attributes *(continued)*

Optional attribute	Description
MonitorListener	Decides whether the monitor and clean function must monitor and clean the MQ listener process.
	If this attribute is set to 1, the agent for WebSphere MQ monitors and cleans the MQ listener process (runmqlsr), as part of mandatory processes.
	If you set this attribute to 0 (default value), the agent for WebSphere MQ does not monitor the runmqlsr process.
	Note: If you enable MonitorListener, you should not configure separate resource for listener process as this process is already monitored and cleaned as part of WebSphereMQ resource.
	Type and dimension: boolean-scaler
	Default: 0
	Example: 1

Note: For information about the additional attributes of the agent for WebSphereMQ when IMF is enabled: See "Attributes that enable IMF" on page 45.

Executing a customized monitoring program

The monitor function can execute a customized monitoring utility to perform an additional WebSphere MQ Queue Manager server state check.

The monitor function executes the utility specified in the MonitorProgram attribute if the following conditions are satisfied:

■ The specified utility is a valid executable file.

■ The first-level process check indicates that the WebSphere MQ Queue Manager instance is online.

■ The second-level monitor check indicates that the WebSphere MQ Queue Manager instance is online.

■ The second-level monitor check is deferred for this monitoring cycle.

The monitor function interprets the utility exit code as follows:

110 or 0	WebSphere MQ Queue Manager server instance is online
100 or 1	WebSphere MQ Queue Manager server instance is offline

| 99 | WebSphere MQ Queue Manager server instance is unknown |
| Any other value | WebSphere MQ Queue Manager server instance is unknown |

To ensure that the customized utility is always available to the agent, Symantec recommends storing the file in a shared directory that is available on an online node.

Enabling the agent for WebSphere MQ for IMF

This chapter includes the following topics:

About Intelligent Monitoring Framework

With intelligent monitoring framework (IMF), VCS supports intelligent resource monitoring in addition to the poll-based monitoring. Poll-based monitoring polls the resources periodically whereas intelligent monitoring performs asynchronous monitoring. You can enable or disable the intelligent resource monitoring functionality of the WebSphere MQ agent.

VCS process and mount-based agents use the Asynchronous Monitoring Framework (AMF) kernel driver that provides asynchronous event notifications to the agents that are enabled for Intelligent Monitoring Framework (IMF).

The WebSphere MQ agent can be enabled for IMF, provided both the following software versions are installed:

- Veritas Cluster Server (VCS) 5.1 SP1 or later
- Veritas High Availability agent for WebSphere MQ version 5.1.9.0 or later

See the *Veritas Cluster Server Administrator's Guide 5.1 Service Pack 1* for more information about:

- IMF notification module functions
- Administering the AMF kernel driver

Benefits of IMF

IMF offers the following benefits:

- Performance
  Enhances performance by reducing the monitoring of each resource at a default of 60 seconds for online resources, and 300 seconds for offline resources. IMF enables the agent to monitor a large number of resources with a minimal effect on performance.

- Faster detection
  Asynchronous notifications would detect a change in the resource state as soon as it happens. Immediate notification enables the agent to take action at the time of the event.

How the agent supports intelligent resource monitoring

When an IMF-enabled agent starts up, the agent initializes the asynchronous monitoring framework (AMF) kernel driver. After the resource is in a steady state, the agent registers with the AMF kernel driver, the details of the resource that are required to monitor the resource. For example, the agent for WebSphere MQ registers the PIDs of the WebSphere MQ Queue Manager processes with the IMF notification module. The agent's imf_getnotification function waits for any resource state changes.When the AMF kernel driver module notifies the imf_getnotification function about a resource state change, the agent framework runs the monitor agent function to ascertain the state of that resource. The agent notifies the state change to VCS, which then takes appropriate action.

See the *Veritas Cluster Server Administrator's Guide 5.1 Service Pack 1* for more information.

Agent functions for the IMF functionality

If the WebSphere MQ agent is enabled for IMF, the agent supports the following functions, in addition to the functions mentioned in WebSphere MQ agent functions.

imf_init

This function initializes the WebSphere MQ agent to interface with the AMF kernel driver, which is the IMF notification module for the agent for WebSphere MQ. This function runs when the agent starts up.

imf_getnotification

This function gets notifications about resource state changes. This function runs after the agent initializes with the AMF kernel module. This function continuously waits for notification and takes action on the resource upon notification.

imf_register

This function registers or unregisters resource entities with the AMF kernel module. This function runs for each resource after the resource goes into a steady state—online or offline.

Attributes that enable IMF

If the agent for WebSphere MQ is enabled for IMF, the agent uses the following type-level attributes in addition to the attributes described in WebSphere MQ agent attributes.

IMF

This resource type-level attribute determines whether the WebSphere MQ agent must perform intelligent resource monitoring. You can also override the value of this attribute at the resource level.

This attribute includes the following keys:

Mode

Define this attribute to enable or disable intelligent resource monitoring. Valid values are as follows:

- 0—Does not perform intelligent resource monitoring

- 1—Performs intelligent resource monitoring for offline resources and performs poll-based monitoring for online resources

- 2—Performs intelligent resource monitoring for online resources and performs poll-based monitoring for offline resources

- 3—Performs intelligent resource monitoring for both online and for offline resources.

Note: The agent for WebSphere MQ supports intelligent resource monitoring for online resources only. Hence, Mode should be set to either 0 or 2.

Type and dimension: integer-association

Default: 0

MonitorFreq

This key value specifies the frequency at which the agent invokes the monitor agent function. The value of this key is an integer.

Default: 1

You can set this key to a non-zero value for cases where the agent requires to perform both poll-based and intelligent resource monitoring.

If the value is 0, the agent does not perform poll-based process check monitoring.

After the resource registers with the AMF kernel driver, the agent calls the monitor agent function as follows:

- After every (MonitorFreq x MonitorInterval) number of seconds for online resources

- After every (MonitorFreq x OfflineMonitorInterval) number of seconds for offline resources

RegisterRetryLimit

If you enable intelligent resource monitoring, the agent invokes the imf_register agent function to register the resource with the AMF kernel driver.

The value of the RegisterRetryLimit key determines the number of times the agent must retry registration for a resource. If the agent cannot register the resource within the limit that is specified, then intelligent monitoring is disabled until the resource state changes or the value of the Mode key changes.

Default: 3.

IMFRegList

An ordered list of attributes whose values are registered with the IMF notification module.

Type and dimension: string-vector

Default: No default value

Note: The attribute values can be overriden at the resource level.

Before you enable the agent for IMF

Before you enable the WebSphere MQ agent for IMF, ensure that the AMF kernel module is loaded and AMF is configured. For details, see the 'Administering the AMF kernel driver' section of the *Veritas Cluster Server Administrator's Guide 5.1 SP1*. For details about the commands you can use to configure AMF, use the `amfconfig -h` command.

Enabling the agent for IMF

In order to enable the WebSphere MQ agent for IMF, you must make the following configuration changes to the attributes of the agent:

- AgentFile: Set the AgentFile attribute to Script51Agent
- IMF Mode: Set the IMF Mode attribute to 2
- IMFRegList: Update the IMFRegList attribute

The following sections provide more information on the commands you can use to make these configuration changes, depending on whether VCS is in a running state or not.

> **Note:** If you have upgraded VCS from an earlier version to version 5.1 SP1, and you already have WebSphere MQ agent version 5.1.9.0 installed, ensure that you run the following commands to create appropriate symbolic links:
>
> ```
> # cd /opt/VRTSagents/ha/bin/WebSphereMQ6
> ```
>
> ```
> # ln -s /opt/VRTSamf/imf/imf_getnotification imf_getnotification
> ```
>
> ```
> # ln -s /opt/VRTSagents/ha/bin/WebSphereMQ6/monitor imf_register
> ```

If VCS is in a running state

To enable the WebSphere MQ resource for IMF when VCS is in a running state:

1   Make the VCS configuration writable.

```
# haconf -makerw
```

2   Run the following command to update the AgentFile attribute.

```
# hatype -modify WebSphereMQ6 AgentFile
/opt/VRTSvcs/bin/Script51Agent
```

3   Run the following command to update the IMF attribute.

```
# hatype -modify WebSphereMQ6 IMF Mode num MonitorFreq num
RegisterRetryLimit num
```

For example, to enable intelligent monitoring of online resources, with the MonitorFreq key set to 5, and the RegisterRetryLimit key is set to 3, run the following command:

```
# hatype -modify WebSphereMQ6 IMF Mode 2 MonitorFreq 5
RegisterRetryLimit 3
```

> **Note:** The valid values for the Mode key of the IMF attribute are 0 (disabled) and 2 (online monitoring).

4   Run the following command to update the IMFRegList attribute:

```
# hatype -modify WebSphereMQ6 IMFRegList QueueManager MQUser MQVer
```

5   Save the VCS configuration.

```
# haconf -dump -makero
```

6   If the WebSphere MQ agent is running, restart the agent.

For information on the commands you can use to restart the agent, see Restarting the agent.

Restarting the agent

To restart the agent:

1   Run the following command to stop the agent forcefully:

```
# haagent -stop WebSphereMQ6 -force -sys <system>
```

Note: Stopping the agent forcefully eliminates the need to take the resource offline.

2   Run the following command to start the agent:

```
# haagent -start WebSphereMQ6 -sys <system>.
```

If VCS is not in a running state

To change the WebSphereMQ6 type definition file when VCS is not in a running state:

1   Update the AgentFile attribute.

```
static str AgentFile = "/opt/VRTSvcs/bin/Script51Agent"
```

2   Update the IMF attribute.

The valid values for the Mode key of the IMF attribute are 0 (disabled) and 2 (online monitoring).

```
static int IMF{} = { Mode=num, MonitorFreq=num,
RegisterRetryLimit=num }
```

For example, to update the IMF attribute such that the Mode key is set to 2, the MonitorFreq key is set to 5, and the RegisterRetryLimit key is set to 3:

```
static int IMF{} = { Mode=2, MonitorFreq=5, RegisterRetryLimit=3
}
```

3   Update the IMFRegList attribute.

```
static str IMFRegList[] = { QueueManager, MQUser, MQVer }
```

Disabling intelligent resource monitoring

To disable intelligent resource monitoring

1   Make the VCS configuration writable.

    ```
    # haconf -makerw
    ```

2   To disable intelligent resource monitoring for all the resources of a certain type, run the following command:

    ```
    # hatype -modify WebSphereMQ6 IMF -update Mode 0
    ```

3   To disable intelligent resource monitoring for a specific resource, run the following command:

    ```
    # hares -override resource_name IMF
    ```

    ```
    # hares -modify resource_name IMF -update Mode 0
    ```

4   Save the VCS configuration.

    ```
    # haconf -dump -makero
    ```

Troubleshooting the configuration for IMF

If you face problems with the IMF configuration or functionality, consider the following:

- Ensure that the following attributes are configured with appropriate values.

  - AgentFile

  - IMF

  - IMFRegList
    If IMFRegList is not configured correctly, the WebSphere MQ resources that have been registered for IMF get unregistered every time the monitor function is run.

- If you have configured the required attributes to enable the WebSphere MQ agent for IMF, but the agent is still not IMF-enabled, restart the agent. The imf_init function runs only when the agent starts up, so when you restart the agent, imf_init runs and initializes the WebSphere MQ agent to interface with the AMF kernel driver.

- You can run the following command to check the value of the MonitorMethod attribute and to verify that a resource is registered for IMF.

    ```
    # hares -value resource MonitorMethod system
    ```

The MonitorMethod attribute specifies the monitoring method that the agent uses to monitor the resource:

■ Traditional—Poll-based resource monitoring

■ IMF—Intelligent resource monitoring

■ You can use the `amfstat` to see a list of registered PIDs for a WebSphereMQ6 resource.
Following is a sample output for the Queue Manager 'testQM'.
The `ps -ef` command output shows the Queue Manager process.

```
# ps -ef | grep -i testQM
mqm 10085 10075   1 13:38:30 ?  0:00 amqzlaa0 -mTestQM
                                        -fip0
mqm 10080 10075   1 13:38:28 ?  0:00 amqzmur0 -m TestQM
mqm 10083 10075   1 13:38:28 ?  0:00 /opt/mqm/bin/
                                        amqzdmaa -m TestQM
mqm 10076 10075   1 13:38:27 ?  0:00 /opt/mqm/bin/
                                        amqzfuma -m TestQM
mqm 10088 10081   1 13:38:30 ?  0:00 amqfqpub -mTestQM
mqm 10086 10084   0 13:38:30 ?  0:00 /opt/mqm/bin/
                                        runmqchi -m TestQM
                                        -q SYSTEM.CHANNEL.
                                        INITQ -r
mqm 10089 10088   1 13:38:30 ?  0:00 amqfcxba -m TestQM
mqm 10077 10075   1 13:38:27 ?  0:00 amqzmuc0 -m TestQM
mqm 10082 10075   0 13:38:28 ?  0:00 /opt/mqm/bin/
                                        amqrrmfa -m TestQM
                                        -t2332800 -s2592000
                                        -p2592000 -g5184000
                                        -c3600
mqm 10084 10075   1 13:38:29 ?  0:00 /opt/mqm/bin/
                                        amqzmgr0 -m TestQM
mqm 10075     1   1 13:38:27 ?  0:00 amqzxma0 -m TestQM
mqm 10081 10075   1 13:38:28 ?  0:00 amqzmuf0 -m TestQM
mqm 10087 10084   1 13:38:30 ?  0:00 /opt/mqm/bin/
                                        amqpcsea TestQM
```

The `amfstat` command shows the Queue Manager PIDs monitored by the WebSphereMQ agent.

```
# amfstat
AMF Status Report
```

```
Registered Reapers (3):
======================
 RID    PID     MONITOR TRIGG   REAPER
163    16417   7       0       WebSphereMQ6

Process ONLINE Monitors (7):
============================
 RID    R_RID   PID     GROUP
165    163     10082   TestQM
166    163     10086   TestQM
167    163     10083   TestQM
168    163     10076   TestQM
169    163     10075   TestQM
170    163     10077   TestQM
171    163     10080   TestQM
```

■ Run the following command to set the ResLogLevel attribute to TRACE. When you set ResLogLevel to TRACE, the agent logs messages in the WebSphereMQ6_A.log file.

```
# hares -modify Queue1 ResLogLevel TRACE
```

■ Run the following command to view the content of the AMF in-memory trace buffer.

```
# amfconfig -p dbglog
```

■ If you have upgraded to VCS version 5.1 SP1, from an earlier VCS version, and if you already have WebSphere MQ agent version 5.1.9.0 installed, ensure that the appropriate symbolic links have been created. You can run the following commands to create appropriate symbolic links:

- ■ `# cd /opt/VRTSagents/ha/bin/WebSphereMQ6`

- ■ `# ln -s /opt/VRTSamf/imf/imf_getnotification imf_getnotification`

- ■ `# ln -s /opt/VRTSagents/ha/bin/WebSphereMQ6/monitor imf_register`

For more troubleshooting information, see the following:

■ See "*Troubleshooting the agent for WebSphere MQ*" on page 61.

■ For information about asynchronous monitoring, attributes that enable asynchronous monitoring, and instructions on using some AMF CLI: *Veritas Cluster Server Administrator's Guide 5.1 Service Pack 1*

■ *Veritas Cluster Server Release Notes 5.1 Service Pack 1*

Sample IMF configurations

An example of a type definition file for a WebSphere MQ agent that is IMF-enabled is as follows.

In this example, the IMF-related attributes are set to the following values:

AgentFile	opt/VRTSvcs/bin/Script51Agent
IMF{}	{ Mode=2, MonitorFreq=5, RegisterRetryLimit=3 }
IMFRegList[]	{ QueueManager, MQUser, MQVer }
LevelTwoMonitorFreq	25

```
type WebSphereMQ6 (
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/WebSphereMQ6"
    static str AgentFile = "/opt/VRTSvcs/bin/Script51Agent"
    static int LevelTwoMonitorFreq = 25
    static str ArgList[] = { ResLogLevel, State, IState, QueueManager,
                            CommandServer, MQUser, MQVer, EnvFile,
                            SecondLevelMonitor, MonitorProgram,
                            MonitorListener }
    static boolean AEPTimeout = 1
    static int IMF{} = { Mode=2, MonitorFreq=5, RegisterRetryLimit=3 }
    static str IMFRegList[] = { QueueManager, MQUser, MQVer }
    str ResLogLevel = INFO
    str QueueManager
    boolean CommandServer = 0
    str MQUser = mqm
    str MQVer = "6.0"
    str EnvFile
    int SecondLevelMonitor
    str MonitorProgram
    boolean MonitorListener = 0
)
```

A sample resource configuration from the `/etc/VRTSvcs/conf/config/main.cf` file is as follows:

```
WebSphereMQ6 Queue1 (
                QueueManager = Queue1
                CommandServer = 1
                MQVer = "7.0"
```

```
                    MonitorListener = 1
                    )
```

Known issues

This release of the agent for WebSphere MQ has the following known issue:

Problem

An error message might appear when you run the `hares -offline` command to take a resource offline.

Description

When a resource is taken offline, it is unregistered from the AMF module. However, the imf_register function attempts to unregister the resource again. This results in the following error message from the engine log.

```
VCS ERROR V-16-2-13710 Resource(Queue1) - imf_register entry point
failed with exit code(1)
```

The following message is logged in the agent log:

```
V-16-55000-10209 Commandline [/opt/VRTSamf/bin/amfregister -u
-rWebSphereMQ6 -g Queue1 ] provided a non-zero exit code --This does
not necessarily indicate a problem ... (Perl's OS error variable
prior to the command-pipe close was [], and after the close was []
) VCSagentFW:messageEngineLog:[AMF amfregisterNOTICEIgnoring the
group unregister request; group named \"Queue1\" not found]
```

Workaround

It is safe to ignore this error message.

Configuring the service groups for WebSphere MQ

This chapter includes the following topics:

- Before configuring the service groups for WebSphere MQ
- Configuring service groups for WebSphere MQ Queue Managers

Before configuring the service groups for WebSphere MQ

Before you configure the WebSphere MQ Queue Manager service group, you must:

- Verify that VCS is installed and configured on all nodes in the cluster where you will configure the service group.
  Refer to the *Veritas Cluster Server Installation Guide* for more information.

- Verify that the Veritas agent for WebSphere MQ is installed on all nodes in the cluster.
  See "Installing the agent in a VCS environment" on page 20.

Configuring service groups for WebSphere MQ Queue Managers

You can cluster WebSphere MQ Queue Managers in a clustered environment, and you can use the Veritas agent for WebSphere MQ to manage these components.

Configuring a WebSphere MQ resource

In a clustered environment, you can configure a WebSphere MQ resource using the following methods:

■ Active-passive configuration

The active-passive configuration is an easier method of configuration. This method limits the configuration to one service group running a WebSphere MQ Queue Manager on a particular node at one time.

■ Active-active configuration

The active-active configuration allows multiple service groups running WebSphere MQ Queue Managers on a particular node simultaneously. This configuration incurs additional complexity in configuration and maintenance.

Active-passive configuration

Use this configuration only where you need WebSphere MQ Queue Managers in a clustered environment.

On the node that hosts the service group, perform the following steps:

To configure a WebSphere MQ Queue Manager using active-passive configuration

1   Ensure that a file system is located on a shared disk.

   This file system must be in the same service group in which the WebSphere MQ is to be created.

2   If required, copy the WebSphere MQ default files from the local copy in to the /var/mqm directory. This directory is a WebSphere MQ configuration item that is not changeable.

3   Mount the file system at the /var/mqm directory.

4   Use the WebSphere MQ tools to create the WebSphere MQ Queue Manager. Refer to the WebSphere MQ documentation for details.

5   Define this WebSphere MQ Queue Manager as a resource in the service group.

   See "Sample service group configurations" on page 71.

   You can now create additional Queue Managers on the same node on which the service group is currently online.

   Ensure that you always define the additional Queue Manager as a cluster server resource in the same service group where other Queue Managers are defined.

Active-active configuration

In an active-active configuration, you can configure each WebSphere MQ Queue Manager in a separate service group and each Queue Manager can fail over independent of each other.

This configuration is complex to implement and maintain. However, this configuration provides the flexibility that some applications may require. This method also supports many-to-one and many-to-many cluster configurations.

On the node that hosts the service group to which the WebSphere MQ Queue Manager belongs, perform the following steps:

To configure a WebSphere MQ Queue Manager using active-active configuration

1   Use the WebSphere MQ tools to create the WebSphere MQ Queue Managers that you require. Refer to the WebSphere MQ documentation for details.

2   Create a file system for each WebSphere MQ on the shared disk. Add each file system to a separate service group.

See Figure A-1 on page 72.

3   Move the log directory from the /var/mqm/log/QueueManager directory to a directory on each file system. Ensure that you copy the sub-directories also.

If a period occurs in the name of the Queue Manager, replace the period with !. For example, if the queue name is venus.veritas and the filesystem is /mq/venus, execute the following commands:

```
# mkdir /mq/venus/log
# cp -rp /var/mqm/log/venus!veritas /mq/venus/log
```

4   Remove the QueueManager directory:

```
# rm -r /var/mqm/log/venus!veritas
```

5   Create a symbolical link between the /var/mqm/log/QueueManager directory and the directory on the file system on which you copied the data in step 3.

Ensure that the permissions for all the copied files, directories, and symbolic links are the same as the original files and are owned by "mqm:mqm".

For example:

```
# ln -s /mq/venus/log /var/mqm/log/venus!veritas
```

```
# chown mqm:mqm  /var/mqm/log/venus!veritas
```

6 Move the qmgr directory from the /var/mqm/qmgr/QueueManager directory to a directory on the shared file system that you created in step 1.

7 Ensure that you copy the sub-directories also.

If a period occurs in the name of the Queue Manager, replace the period with !. For example, if the queue name is venus.veritas and the filesystem is /mq/venus, execute the following commands:

```
# mkdir /mq/venus/qmgrs
# cp -rp /var/mqm/qmgrs/venus!veritas /mq/venus/qmgrs
```

8 Remove the QueueManager directory:

```
# rm -r /var/mqm/qmgrs/venus!veritas
```

9 Create a symbolical link between the /var/mqm/qmgrs/QueueManager directory and the directory on the file system on which you copied the data in step 6.

Ensure that the permissions for all the copied files, directories, and symbolic links are the same as the original files and are owned by "mqm:mqm".

For example:

```
# ln -s /mq/venus/qmgrs /var/mqm/qmgrs/venus!veritas

chown mqm:mqm  /var/mqm/qmgrs/venus!veritas
```

10 Define the Queue Managers as resources in separate service groups.

See Figure A-1 on page 72.

The WebSphere MQ can run on many nodes in the cluster. These nodes are defined in the SystemList attribute. On all such nodes, perform the following steps:

■ Create a symbolical link between the /var/mqm/log/QueueManager and the directory in which the logs were copied in step 3.

■ Create a symbolical link between the /var/mqm/qmgr/QueueManager and the directory in which the qmgr directory was copied in step 6.

■ Add the following lines at the end of the /var/mqm/mqs.ini file using a text editor:

```
QueueManager:
   Name=QueueManager
```

```
Prefix=/var/mqm
Directory=QueueManager
```

If all these nodes are to handle the queues, then copy the /var/mqm/mqs.ini file from the first node to all other nodes.

Follow these steps whenever you want to add new WebSphere MQ Queue Managers in the cluster.

Configuring a WebSphere MQ listener

A WebSphere MQ Queue Manager uses a Listener to listen for requests on a specific IP address. You can configure a Listener resource in the cluster using a bundled application agent. An example listener resource configuration is shown as follows. In this example, the virtual IP address is set to 1.2.3.4 and the Queue Manager name is venus.veritas.

See "WebSphere MQ agent attributes" on page 37.

You can enable the MonitorListener attribute, if the listener is configured to start automatically when the queue manager starts.

You can enable the MonitorListener attribute, to start Listener when WebSphere MQ resource is online.

You can replace these values with the virtual IP address and Queue Manager name defined within the cluster.

```
Application was4WSMQ_listen
(
   User        = mqm
   StartProgram  = "/opt/mqm/bin/runmqlsr -t tcp
                    -i 1.2.3.4 -m venus.veritas &"
   StopProgram   = "/opt/mqm/bin/endmqlsr -m venus.veritas"
   MonitorProcesses  = {"/opt/mqm/bin/runmqlsr -t tcp
                        -i 1.2.3.4 -m venus.veritas" }
)
```

For details about the WebSphere MQ listener, refer to the WebSphere MQ documentation.

Troubleshooting the agent for WebSphere MQ

This chapter includes the following topics:

- Using the correct software and operating system versions
- Meeting prerequisites
- Configuring WebSphere MQ Queue Manager resources
- Starting the WebSphere MQ Queue Manager instance outside a cluster
- Monitoring WebSphere MQ Queue Manager processes
- Stopping WebSphere MQ Queue Manager processes forcefully
- Reviewing error log files

Using the 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:

See "Supported software" on page 12.

Meeting prerequisites

Before installing the agent for WebSphere MQ, double check that you meet the prerequisites.

For example, you must install the ACC library on VCS before installing the agent for WebSphere MQ.

See "Before you install the Veritas agent for WebSphere MQ" on page 17.

Note: For information about the prerequisites for IMF and for other IMF-related troubleshooting information: See "Troubleshooting the configuration for IMF" on page 50.

Configuring WebSphere MQ Queue Manager resources

Before using WebSphere MQ Queue Manager resources, ensure that you configure the resources properly. For a list of attributes used to configure all WebSphere MQ Queue Manager resources, refer to the agent attributes.

Starting the WebSphere MQ Queue Manager instance outside a cluster

If you face problems while working with a resource, you must disable the resource within the cluster framework. A disabled resource is not under the control of the cluster framework, and so you can test the WebSphere MQ Queue Manager instance independent of the cluster framework. Refer to the cluster documentation for information about disabling a resource.

You can then restart the WebSphere MQ Queue Manager instance outside the cluster framework.

Note: Use the same parameters that the resource attributes define within the cluster framework while restarting the resource outside the cluster framework.

A sample procedure to start a WebSphere MQ instance outside the cluster framework, is illustrated as follows.

To restart the WebSphere MQ Queue Manager outside the framework

1   Log in to the WebSphere MQ Queue Manager as an MQUser.

```
# su - MQUser
```

2   Start the WebSphere MQ Queue Manager.

```
# strmqm QueueManagerName
```

If the WebSphere MQ Queue Manager works properly outside the cluster framework, you can then attempt to implement the Queue Manager within the cluster framework.

Monitoring WebSphere MQ Queue Manager processes

The agent for WebSphere MQ monitors the following processes:

MQ 5.3	`"amqhasmx X_QUEUE_MANAGER_X( |\$)",`
	`"amqzllp0 .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzlaa0 .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqrrmfa .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"runmqchi .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzdmaa .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzfuma .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzxma0 .*-m *X_QUEUE_MANAGER_X( |\$)",`
MQ 6.0 and later	`"amqrrmfa .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"runmqchi .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzdmaa .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzfuma .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzxma0 .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzmuc0 .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzmur0 .*-m *X_QUEUE_MANAGER_X( |\$)",`

Stopping WebSphere MQ Queue Manager processes forcefully

As per IBM recommendations, when an attempt to gracefully stop the Queue Manager fails, the agent for WebSphere MQ kills the processes in the following order:

MQ 5.3
```
"amqhasmx X_QUEUE_MANAGER_X( |\$)",
"amqzllp0 .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqzlaa0 .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqrrmfa .*-m *X_QUEUE_MANAGER_X( |\$)",
"runmqchi .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqzdmaa .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqzfuma .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqzxma0 .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqpcsea .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqharmx .*-m *X_QUEUE_MANAGER_X( |\$)",
"runmqlsr .*-m *X_QUEUE_MANAGER_X( |\$)",
```

MQ 6.0
```
"amqzmuc0 .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqzxma0 .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqzfuma .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqzlaa0 .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqzlsa0 .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqzmgr0 .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqzmur0 .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqrmppa .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqrrmfa .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqzdmaa .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqpcsea *X_QUEUE_MANAGER_X( |\$)",
"amqhasmx X_QUEUE_MANAGER_X( |\$)",
"amqzllp0 .*-m *X_QUEUE_MANAGER_X( |\$)",
"runmqchi .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqharmx .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqfqpub .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqfcxba .*-m *X_QUEUE_MANAGER_X( |\$)",
"amqcrsta .*-m *X_QUEUE_MANAGER_X( |\$)",
"runmqsc *X_QUEUE_MANAGER_X( |\$)",
"runmqlsr .*-m *X_QUEUE_MANAGER_X( |\$)",
```

MQ 7.0	`"amqzmuc0 .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzxma0 .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzfuma .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzlaa0 .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzlsa0 .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzmuf0 .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzmur0 .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzmgr0 .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqfqpub .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqfcxba .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqrmppa .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqcrsta .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqcrs6b .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqrrmfa .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqzdmaa .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqpcsea *X_QUEUE_MANAGER_X( |\$)",`
	`"runmqtrm .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"runmqdlq .*X_QUEUE_MANAGER_X( |\$)",`
	`"runmqchi .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"runmqlsr .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqxssvn .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"amqztrcn .*-m *X_QUEUE_MANAGER_X( |\$)",`
	`"runmqsc *X_QUEUE_MANAGER_X( |\$)",`

Reviewing error log files

If you face problems while using WebSphere MQ Queue Manager or the agent for
WebSphere MQ, use the log files described in this section to investigate the
problems.

Using WebSphere MQ log files

If a WebSphere MQ Queue Manager is facing problems, you can access the server
log files to further diagnose the problem. The WebSphere MQ Queue Manager log
files are located in the *<Queue Manager Home>*/qmgrs/*<Queue Manager
Name>*/errors directory.

Reviewing cluster log files

In case of problems while using the agent for WebSphere MQ, you can also access
the engine log file for more information about a particular resource. The engine
log files are located at the following location:

■ The VCS engine log file is /var/VRTSvcs/log/engine_A.log.

- The VCS One engine log file is /var/VRTSvcsone/log/engine_A.log.
- The VCS One client log file is /var/VRTSvcsone/log/vcsoneclientd_A.log.

Using trace level logging

The ResLogLevel attribute controls the level of logging that is written in a cluster log file for each WebSphere MQ Queue Manager resource. You can set this attribute to TRACE, which enables very detailed and verbose logging.

If you set ResLogLevel to TRACE, a very high volume of messages are produced. Symantec recommends that you localize the ResLogLevel attribute for a particular resource.

To localize ResLogLevel attribute for a resource

1   Identify the resource for which you want to enable detailed logging.

2   Localize the ResLogLevel attribute for the identified resource:

    # hares -local *Resource_Name* ResLogLevel

3   Set the ResLogLevel attribute to TRACE for the identified resource:

    # hares -modify *Resource_Name* ResLogLevel TRACE -sys *SysA*

4   Note the time before you begin to operate the identified resource.

5   Test the identified resource. The function reproduces the problem that you are attempting to diagnose.

6   Note the time when the problem is reproduced.

7   Set the ResLogLevel attribute back to INFO for the identified resource:

    # hares -modify *Resource_Name* ResLogLevel INFO -sys *SysA*

8   Review the contents of the log file. Use the time noted in Step 4 and Step 6 to diagnose the problem.

    You can also contact Symantec support for more help.

Sample Configurations

This appendix includes the following topics:

■ About sample configurations for the agent for WebSphere MQ

■ Sample agent type definition for WebSphere MQ

■ Sample configuration when IMF is enabled

■ Sample configuration in a VCS environment

■ Sample configuration in a VCS One environment

■ Sample service group configurations

About sample configurations for the agent for WebSphere MQ

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

Sample agent type definition for WebSphere MQ

After importing the agent types into the cluster, if you save the configuration on your system disk using the `haconf -dump` command, you can find the WebSphereMQ6Types.cf file in the /etc/VRTSvcs/conf/config cluster configuration directory.

Examples of agent type definition files for different versions of VCS are as follows:

For VCS 4.x

```
type WebSphereMQ6
(
    static str ArgList[] = { ResLogLevel, State, IState,
                             QueueManager, CommandServer, MQUser,
                             MQVer, EnvFile, SecondLevelMonitor,
                             MonitorProgram, MonitorListerner }
    str ResLogLevel = INFO
    str QueueManager
    boolean CommandServer = 1
    str MQUser = mqm
    str MQVer = "6.0"
    str EnvFile
    int SecondLevelMonitor
    str MonitorProgram
    boolean MonitorListerner = 0
)
```

For VCS 5.x

```
type WebSphereMQ6
(
    static str AgentFile = "/opt/VRTSvcs/bin/Script50Agent"
    static str AgentDirectory =
"/opt/VRTSagents/ha/bin/WebSphereMQ6"
    static str ArgList[] = { ResLogLevel, State, IState,
                             QueueManager, CommandServer, MQUser,
                             MQVer, EnvFile, SecondLevelMonitor,
                             MonitorProgram, MonitorListerner }
    str ResLogLevel = INFO
    str QueueManager
    boolean CommandServer = 1
    str MQUser = mqm
    str MQVer = "6.0"
    str EnvFile
    int SecondLevelMonitor
    str MonitorProgram
    boolean MonitorListerner = 0
)
```

VCS One

After installing the agent, go to the /etc/VRTSagents/ha/conf/WebSphereMQ6/ directory to view the WebSphereMQ6Types.platform.xml agent definition file.

Sample configuration when IMF is enabled

For a sample configuration when the WebSphere MQagent is configured for IMF:
See

Sample configuration in a VCS environment

An excerpt from the main.cf file that includes a WebSphere MQ resource follows.

```
group WASMQ_Sol_x64 (
       SystemList = { system_A = 0, system_B = 1 }
       )
        DiskGroup DG_OPT (
               DiskGroup = WAS
               )
        DiskGroup DG_VAR (
               DiskGroup = WAS
               )

       Mount Mount_OPT (
               MountPoint = "/opt/mqm"
               BlockDevice = "/dev/vx/dsk/WAS/MQ_Opt"
               FSType = vxfs
               FsckOpt = "-y"
               )

       Mount Mount_VAR (
               MountPoint = "/var/mqm"
               BlockDevice = "/dev/vx/dsk/WAS/MQ_Vol"
               FSType = vxfs
               FsckOpt = "-y"
               )

       Volume Volum_OPT (
               Volume = MQ_Opt
               DiskGroup = WAS
               )
```

```
 Volume Volume_VAR (
        Volume = MQ_Vol
        DiskGroup = WAS
        )


WebSphereMQ6 WASMQ (
        QueueManager = MQ1
        CommandServer = 1
        MQVer = "7.0"
        SecondLevelMonitor = 5
        MonitorProgram = "/ibm/mq/myMonitor.sh"
        MonitorListener = 1
        )


Mount_OPT requires Volum_OPT

Mount_VAR requires Volume_VAR

Volum_OPT requires DG_OPT

Volume_VAR requires DG_VAR

WASMQ requires Mount_OPT

WASMQ requires Mount_VAR


// resource dependency tree
//      group WASMQ_Sol_x64
//      {
//      WebSphereMQ6 WASMQ
//          {
//          Mount Mount_OPT
//              {
//              Volume Volum_OPT
//                  {
//                  DiskGroup DG_OPT
//                  }
//              }
//          Mount Mount_VAR
//              {
//              Volume Volume_VAR
```

```
//                    {
//                    DiskGroup DG_VAR
//                    }
//              }
//         }
//     }
```

Sample configuration in a VCS One environment

To view a sample VCS One configuration file (main.xml) with an MQ Listener and a WebSphere MQ Queue Manager, go to the /etc/VRTSagents/ha/conf/WebSphereMQ6/ directory.

Sample service group configurations

Figure A-1 shows a sample service group that shows two WebSphere MQ Queue Manager resources.

This simple configuration also requires a Mount and a Disk Group resource.

Figure A-1   Sample Service group configuration
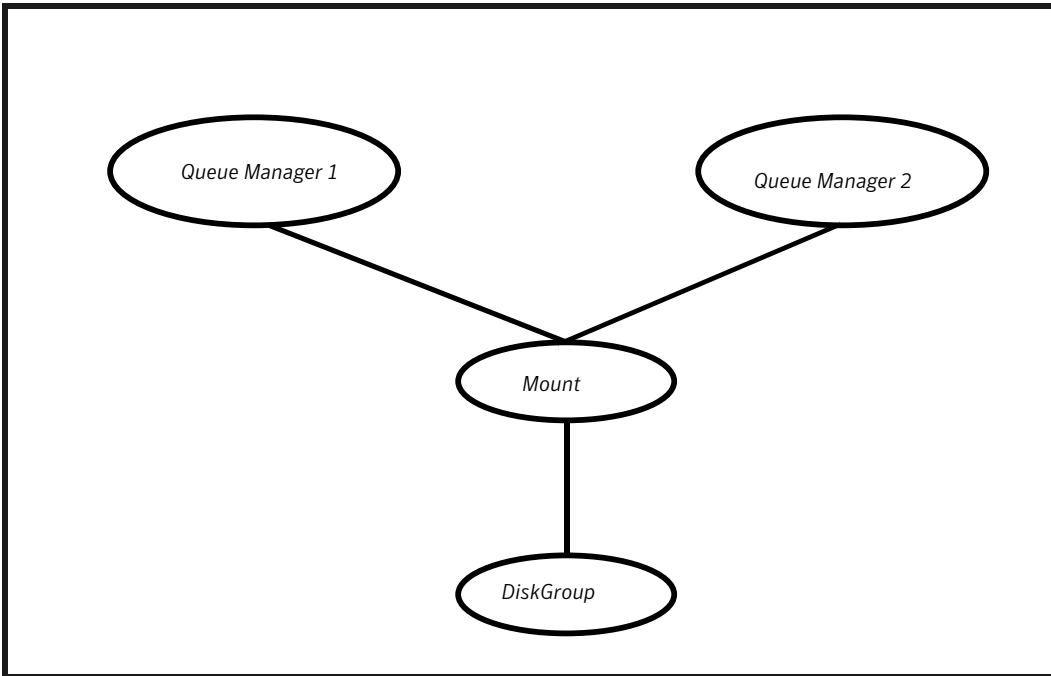
Figure A-2 shows a sample service group that includes two WebSphere MQ Queue Manager resources with associated listeners.

In this example, each resource depends on a listener, which in turn depends on an IP and a Mount resource. This configuration applies to WebSphere MQ when listeners are used to provide remote services to application clients.

Figure A-2   Sample Service group configuration with listeners
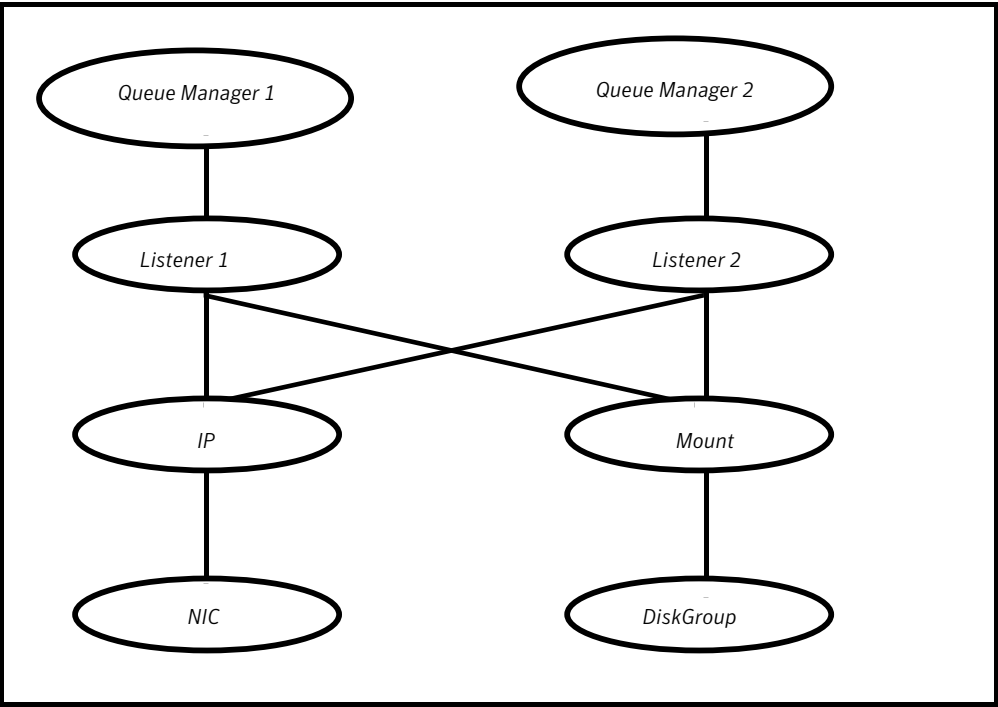
Appendix B

Changes introduced in previous releases

This appendix includes the following topics:

■ Changes introduced in previous releases

Changes introduced in previous releases

The enhancements in the previous releases of Veritas agent for WebSphere MQ agent are as follows:

■ The SecondLevelMonitor attribute of the Veritas agent for WebSphere MQ is deprecated if both the following software versions are installed:

■ Veritas Cluster Server (VCS) 5.1 SP1

■ Veritas agent for WebSphere MQ version 5.1.9.0

In such a case, instead of the SecondLevelMonitor attribute, you must use the resource type level attribute 'LevelTwoMonitorFreq' to specify the frequency of in-depth monitoring. For more information: See Table 3-2 on page 39.

■ Added Intelligent Monitoring Framework (IMF) capability and support for intelligent resource monitoring.

■ Added support for Fast First Level Monitor (FFLM). The agent maintains PID files based on search patterns to expedite the first level process based monitoring check.

■ From this release of the WebSphere MQ agent (version 5.1.9.0) onwards, the agent does not source the user's profile while executing commands. You can use the EnvFile attribute to set up the required environment. For more information: See Table 3-2 on page 39.

- When the WebSphereMQ resource is offline and the agent detects the queue manager processes as running, but the second level monitor check fails, the agent cleans these processes.

- Added new attribute MonitorListener from WebSphereMQ6 agent version 5.1.7.0.
  If this attribute is enabled, the agent monitors and cleans MQ listener process (runmqlsr) along with other set of mandatory processes.

- As per IBM recommendations, updated the list of processes that may be killed during clean entry point.

- Added support for VCS 5.1 on AIX, Linux, and Solaris.

- Added support for VCS One 5.0.

- Added support for AIX 6.1 on pSeries.

- Added support for SUSE Linux Enterprise Server 9.

- Added support for VCS One 2.0.

- Added support for WebSphere MQ version 6.0 and 7.0.

- Added support for Solaris x86 for VCS 4.1 and 5.0

- Added support for Internationalization (i18n).

- Added support for ACC library 5.0 that is compliant with VCS and VCS One.

- Added support for zones on Solaris 10 in a VCS environment.

- Added support for the First Failure Data Capture (FFDC) feature.

- Removed the following attributes:

  - MQImmediateTimeout

  - MQPreemptiveTimeout

  - FirstMonitorDelay

  - VProLogLevel

- Added the following attributes:

  - MQVer

  - MonitorProgram

  - ResLogLevel

  - CommandServer

- Added support for monitoring the Command Server for WebSphere MQ version 6.0 and later. The WebSphere MQ enables remote administration of queue

managers. To facilitate this functionality, a Command Server can run within a WebSphere MQ Queue Manager. This Command Server executes commands sent to the Queue Manager.

Index