

Veritas[™] Cluster Server Administrator's Guide

AIX

5.1

Veritas Cluster Server Administrator's 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.

Product version: 5.1

Document version: 5.1.0

Legal Notice

Copyright © 2009 Symantec Corporation. All rights reserved.

Symantec, the Symantec Logo, Veritas and Veritas Storage Foundation are trademarks or registered trademarks of Symantec Corporation or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation/reverse engineering. No part of this document may be reproduced in any form by any means without prior written authorization of Symantec Corporation and its licensors, if any.

THE DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. SYMANTEC CORPORATION SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

The Licensed Software and Documentation are deemed to be commercial computer software as defined in FAR 12.212 and subject to restricted rights as defined in FAR Section 52.227-19 "Commercial Computer Software - Restricted Rights" and DFARS 227.7202, "Rights in Commercial Computer Software or Commercial Computer Software Documentation", as applicable, and any successor regulations. Any use, modification, reproduction release, performance, display or disclosure of the Licensed Software and Documentation by the U.S. Government shall be solely in accordance with the terms of this Agreement.

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 maintenance 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 Web-based support that provides rapid response and up-to-the-minute information
- Upgrade assurance that delivers automatic software upgrade protection
- Global support that is available 24 hours a day, 7 days a week
- Advanced features, including Account Management Services

For information about Symantec's Maintenance Programs, you can visit our Web site at the following URL:

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

Contacting Technical Support

Customers with a current maintenance 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 non-technical support Web page at the following URL:

customercare.symantec.com

Customer service

Customer Care information is available at the following URL:

customercare.symantec.com

Customer Service is available to assist with 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 maintenance 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

Documentation feedback

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:

sfha_docs@symantec.com

Maintenance agreement resources

If you want to contact Symantec regarding an existing maintenance agreement, please contact the maintenance 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

Additional enterprise services

Symantec offers a comprehensive set of services that allow you to maximize your investment in Symantec products and to develop your knowledge, expertise, and global insight, which enable you to manage your business risks proactively.

Enterprise services that are available include the following:

Symantec Early Warning Solutions	These solutions provide early warning of cyber attacks, comprehensive threat analysis, and countermeasures to prevent attacks before they occur.
Managed Security Services	These services remove the burden of managing and monitoring security devices and events, ensuring rapid response to real threats.
Consulting Services	Symantec Consulting Services provide on-site technical expertise from Symantec and its trusted partners. Symantec Consulting Services offer a variety of prepackaged and customizable options that include assessment, design, implementation, monitoring, and management capabilities. Each is focused on establishing and maintaining the integrity and availability of your IT resources.
Educational Services	Educational Services provide a full array of technical training, security education, security certification, and awareness communication programs.

To access more information about Enterprise services, please visit our Web site at the following URL:

www.symantec.com

Select your country or language from the site index.

Contents

Section I	Clustering concepts and terminology	
Chapter 1	Introducing Veritas Cluster Server	
	What is a VCS cluster?	18
	Can my application be clustered?	20
	Physical components of VCS	22
	Logical components of VCS	23
	Putting the pieces together	35
Chapter 2	About cluster topologies	
	Basic failover configurations	38
	Advanced failover configurations	42
	Cluster topologies and storage configurations	45
Chapter 3	VCS configuration concepts	
	About configuring VCS	52
	About the VCS configuration language	52
	About the main.cf file	53
	The types.cf file	55
	About VCS attributes	57
	About VCS keywords and reserved words	61
	VCS environment variables	62
Section II	Administration-Putting VCS to work	
Chapter 4	About the VCS user privilege model	
	About VCS user privileges and roles	72
	How administrators assign roles to users	75
	User privileges for OS user groups for clusters running in secure mode ...	76
	About VCS privileges for users with multiple roles	77

Chapter 5	Administering the cluster from Cluster Manager (Java console)	
	About the Cluster Manager (Java Console)	80
	Getting started	81
	Reviewing components of the Java Console	84
	About Cluster Monitor	86
	About Cluster Explorer	92
	Accessing additional features of the Java Console	104
	Administering Cluster Monitor	112
	Administering user profiles	117
	Administering service groups	121
	Administering resources	143
	Administering systems	163
	Administering clusters	165
	Executing commands	166
	Editing attributes	167
	Querying the cluster configuration	169
	Setting up VCS event notification using the Notifier wizard	170
	Administering logs	173
	Administering VCS Simulator	177
Chapter 6	Administering the cluster from the command line	
	About administering VCS from the command line	180
	Installing a VCS license	185
	Administering LLT	187
	Starting VCS	190
	Stopping VCS	192
	Logging on to VCS	196
	Managing VCS configuration files	198
	Managing VCS users from the command line	207
	Querying VCS	210
	Administering service groups	217
	Administering agents	223
	Administering resources	224
	Administering systems	231
	Administering clusters	232
	Enabling and disabling Security Services	237
	Administering resource types	239
	Using the -wait option in scripts	241
	Running HA fire drills	243
	Administering simulated clusters from the command line	244

Chapter 7	Configuring applications and resources in VCS	
	About configuring resources and applications	246
	About VCS bundled agents	246
	Configuring the RemoteGroup agent	251
	Configuring NFS service groups	254
	Configuring the Coordination Point agent	263
	Testing resource failover using HA fire drills	263
Chapter 8	Predicting VCS behavior using VCS Simulator	
	About VCS Simulator	266
	Simulator ports	266
	Administering VCS Simulator from the Java Console	267
	Administering VCS Simulator from the command line	274
Section III	VCS communication and operations	
Chapter 9	About communications, membership, and data protection in the cluster	
	About cluster communications	282
	About cluster membership	286
	About membership arbitration	289
	About data protection	307
	Examples of VCS operation with I/O fencing	308
	About cluster membership and data protection without I/O fencing	318
	Examples of VCS operation without I/O fencing	320
	Summary of best practices for cluster communications	325
Chapter 10	Administering I/O fencing	
	About administering I/O fencing	328
	About the vxfsentsthdw utility	328
	About the vxfsenadm utility	336
	About the vxfsenclearpre utility	340
	About the vxfsenswap utility	341
	Administering the coordination point server	349
Chapter 11	Controlling VCS behavior	
	About VCS behavior on resource faults	362
	Controlling VCS behavior at the service group level	365
	Controlling VCS behavior at the resource level	373
	Changing agent file paths and binaries	386

VCS behavior on loss of storage connectivity	387
Service group workload management	390
Sample configurations depicting workload management	393

Chapter 12 The role of service group dependencies

About service group dependencies	410
Service group dependency configurations	415
Group Dependency FAQs	423
Linking service groups	424
VCS behavior with service group dependencies	424

Section IV Administration-Beyond the basics

Chapter 13 VCS event notification

About VCS event notification	430
Components of VCS event notification	432
VCS events and traps	435
Monitoring aggregate events	443
Detecting complementary events	444
Configuring notification	444

Chapter 14 VCS event triggers

About VCS event triggers	446
Using event triggers	446
List of event triggers	447

Section V Cluster configurations for disaster recovery

Chapter 15 Connecting clusters—Creating global clusters

How VCS global clusters work	466
VCS global clusters: The building blocks	467
Prerequisites for global clusters	474
About setting up global clusters	476
Setting up a global cluster	478
When a cluster faults	489
Setting up a fire drill	492
Multi-tiered application support using the RemoteGroup agent in a global environment	499
Test scenario for a multi-tiered environment	501

Chapter 16	Administering global clusters from Cluster Manager (Java console)	
	About global clusters	508
	Adding a remote cluster	509
	Deleting a remote cluster	513
	Administering global service groups	516
	Administering global heartbeats	520
Chapter 17	Administering global clusters from the command line	
	About administering global clusters from the command line	526
	Global querying	526
	Administering global service groups	533
	Administering resources	535
	Administering clusters in global clusters	536
	Administering heartbeats	538
Chapter 18	Setting up replicated data clusters	
	About replicated data clusters	540
	How VCS replicated data clusters work	541
	Setting up a replicated data cluster configuration	542
	Migrating a service group	545
	Setting up a fire drill	546
Chapter 19	Setting up campus clusters	
	About campus cluster configuration	548
	VCS campus cluster requirements	548
	Typical VCS campus cluster setup	549
	How VCS campus clusters work	550
	Setting up a campus cluster configuration	554
	About fire drill in campus clusters	557
	About the DiskGroupSnap agent	557
	Running a fire drill in a campus cluster	558
Section VI	Troubleshooting and performance	
Chapter 20	VCS performance considerations	
	How cluster components affect performance	564
	How cluster operations affect performance	568
	Scheduling class and priority configuration	574
	Monitoring CPU usage	578

VCS agent statistics	579
About VXFEN tunable parameters	582

Chapter 21 Troubleshooting and recovery for VCS

Logging	586
Troubleshooting the VCS engine	588
Troubleshooting LLT	590
Troubleshooting GAB	592
Troubleshooting VCS startup	593
Troubleshooting service groups	595
Troubleshooting resources	597
Troubleshooting I/O fencing	600
Troubleshooting notification	613
Troubleshooting VCS configuration backup and restore	613
Troubleshooting and recovery for global clusters	615
Troubleshooting licensing	618

Section VII Appendixes

Appendix A VCS user privileges—administration matrices

About administration matrices	623
Administration matrices	624

Appendix B VCS commands: quick reference

About this quick reference for VCS commands	631
VCS command line reference	632

Appendix C Cluster and system states

Remote cluster states	636
System states	638

Appendix D VCS attributes

About attributes and their definitions	644
Resource attributes	644
Resource type attributes	650
Service group attributes	663
System attributes	680
Cluster attributes	688
Heartbeat attributes (for global clusters)	697

Appendix E	Configuring VCS for partitions	
	About VCS support for WPARs	700
	Configuring VCS in WPARs	702
	Installing the application	707
	Verifying the WPAR configuration	710
	Maintenance tasks	710
	Troubleshooting information	711
	About VCS support for Live Partition Mobility	711
	About configuring failovers among physical and virtual servers	711
	Configuring for failovers—a typical setup	712
Appendix F	Accessibility and VCS	
	About accessibility in VCS	714
	Navigation and keyboard shortcuts	714
	Support for accessibility settings	715
	Support for assistive technologies	715
Glossary		717
Index		721

Clustering concepts and terminology

- [Chapter 1, “Introducing Veritas Cluster Server” on page 17](#)
- [Chapter 2, “About cluster topologies” on page 37](#)
- [Chapter 3, “VCS configuration concepts” on page 51](#)

Introducing Veritas Cluster Server

- What is a VCS cluster?
- Can my application be clustered?
- Physical components of VCS
- Logical components of VCS
- Putting the pieces together

What is a VCS cluster?

Veritas Cluster Server (VCS) from Symantec connects multiple, independent systems into a management framework for increased availability. Each system, or node, runs its own operating system and cooperates at the software level to form a cluster. VCS links commodity hardware with intelligent software to provide application failover and control. When a node or a monitored application fails, other nodes can take predefined actions to take over and bring up services elsewhere in the cluster.

How VCS detects failure

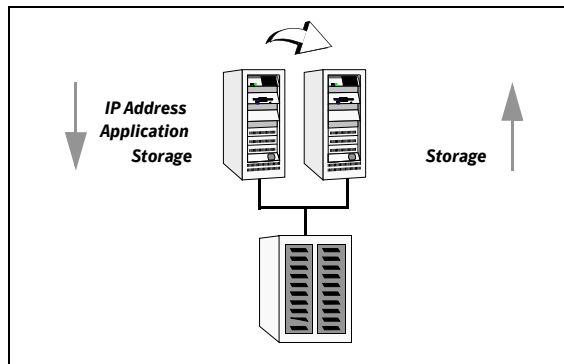
VCS detects failure of an application by issuing specific commands, tests, or scripts to monitor the overall health of an application. VCS also determines the health of underlying resources supporting the application, such as file systems and network interfaces.

VCS uses a redundant network heartbeat to discriminate between the loss of a system and the loss of communication between systems. VCS also uses membership coordination and data protection for detecting failure on a node and on fencing.

See [“About cluster control, communications, and membership”](#) on page 29.

How VCS ensures application availability

When VCS detects an application or node failure, VCS brings application services up on a different node in a cluster. VCS virtualizes IP addresses and system names, so client systems continue to access the application and are unaware of which server they use.



For example, in a 2-node cluster consisting of db-server1 and db-server2, a virtual address may be called db-server. Clients access db-server and are unaware of which physical server hosts the db-server.

About switchover and failover

Switchover and failover are the processes of bringing up application services on a different node in a cluster.

Switchover	A switchover is an orderly shutdown of an application and its supporting resources on one server and a controlled startup on another server.
Failover	A failover is similar to a switchover, except the ordered shutdown of applications on the original node may not be possible, so the services are started on another node.

Can my application be clustered?

Most applications can be placed under cluster control provided the following guidelines are met:

- Defined start, stop, and monitor procedures.
- Ability to restart in a known state.
- Ability to store required data on shared disks.
- Adherence to license requirements and host name dependencies.

Defined start, stop, and monitor procedures

The application to be clustered must have defined procedures for starting, stopping, and monitoring.

- | | |
|-------------------|---|
| Start procedure | <p>The application must have a command to start it and all resources it may require. VCS brings up the required resources in a specific order, then brings up the application using the defined start procedure.</p> <p>For example, to start an Oracle database, VCS must know which Oracle utility to call, such as sqlplus. VCS must also know the Oracle user, instance ID, Oracle home directory, and the pfile.</p> |
| Stop procedure | <p>An individual instance of the application must be capable of being stopped without affecting other instances.</p> <p>For example, killing all HTTPd processes on a Web server is unacceptable because it would also stop other Web servers.</p> <p>If VCS cannot stop an application cleanly, it may call for a more forceful method, like a kill signal. After a forced stop, a clean-up procedure may be required for various process- and application-specific items that may be left behind. These items include shared memory segments or semaphores.</p> |
| Monitor procedure | <p>The application must have a monitor procedure that determines if the specified application instance is healthy. The application must allow individual monitoring of unique instances.</p> <p>For example, the monitor procedure for a Web server connects to the specified server and verifies that it serves Web pages. In a database environment, the monitoring application can connect to the database server and perform SQL commands to verify read and write access to the database.</p> <p>The closer a test comes to matching what a user does, the better the test is in discovering problems. You should balance the level of monitoring between ensuring that the application is up and minimizing monitor overhead.</p> |

Ability to restart the application in a known state

When the application is taken offline, it must close out all tasks, store data properly on shared disk, and exit. Stateful servers must not keep that state of clients in memory. States should be written to shared storage to ensure proper failover.

Commercial databases such as Oracle, Sybase, or SQL Server are good examples of well-written, crash-tolerant applications. On any client SQL request, the client is responsible for holding the request until it receives acknowledgement from the server. When the server receives a request, it is placed in a special *redo* log file. The database confirms that the data is saved before it sends an acknowledgement to the client. After a server crashes, the database recovers to the last-known committed state by mounting the data tables and applying the redo logs. This returns the database to the time of the crash. The client resubmits any outstanding client requests that are unacknowledged by the server, and all others are contained in the redo logs.

If an application cannot recover gracefully after a server crashes, it cannot run in a cluster environment. The takeover server cannot start up because of data corruption and other problems.

External data storage

The application must be capable of storing all required data and configuration information on shared disks. The exception to this rule is a true *shared nothing* cluster.

See “[Shared nothing cluster](#)” on page 47.

To meet this requirement, you may need specific setup options or soft links. For example, a product may only install in `/usr/local`. This limitation requires one of the following options: linking `/usr/local` to a file system that is mounted from the shared storage device or mounting file system from the shared device on `/usr/local`.

The application must also store data to disk rather than maintaining it in memory. The takeover system must be capable of accessing all required information. This requirement precludes the use of anything inside a single system inaccessible by the peer. NVRAM accelerator boards and other disk caching mechanisms for performance are acceptable, but must be done on the external array and not on the local host.

Licensing and host name issues

The application must be capable of running on all servers that are designated as potential hosts. This requirement means strict adherence to licensing requirements and host name dependencies. Changing host names can lead to significant management issues when multiple systems have the same host name after an outage. Custom scripting to modify a system host name on failover is not recommended. Symantec recommends you configure applications and licensing to run properly on all hosts.

Physical components of VCS

A VCS cluster comprises of systems that are connected with a dedicated communications infrastructure. VCS refers to a system that is part of a cluster as a node.

Each cluster has a unique cluster ID. Redundant cluster communication links connect systems in a cluster.

Nodes

VCS nodes host the service groups (managed applications). Each system is connected to networking hardware, and usually also to storage hardware. The systems contain components to provide resilient management of the applications, and start and stop agents.

Nodes can be individual systems, or they can be created with domains or partitions on enterprise-class systems. Individual cluster nodes each run their own operating system and possess their own boot device. Each node must run the same operating system within a single VCS cluster.

Clusters can have from 1 to 32 nodes. Applications can be configured to run on specific nodes within the cluster.

Shared storage

Storage is a key resource of most applications services, and therefore most service groups. A managed application can only be started on a system that has access to its associated data files. Therefore, a service group can only run on all systems in the cluster if the storage is shared across all systems. In many configurations, a storage area network (SAN) provides this requirement.

You can use I/O fencing technology for data protection. I/O fencing blocks access to shared storage from any system that is not a current and verified member of the cluster.

See “[About cluster topologies](#)” on page 37.

Networking

Networking in the cluster is used for the following purposes:

- Communications between the cluster nodes and the customer systems.
- Communications between the cluster nodes.

Logical components of VCS

VCS is comprised of several components that provide the infrastructure to cluster an application.

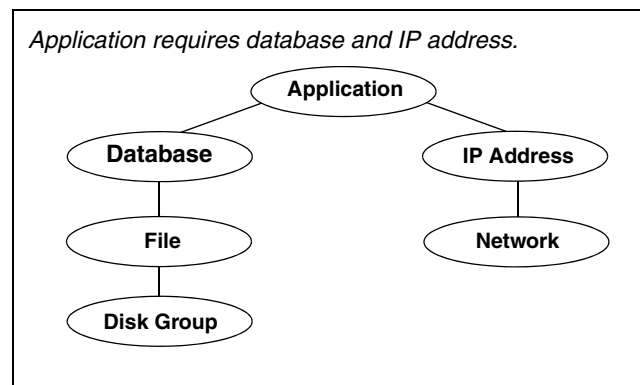
Resources and resource dependencies

Resources are hardware or software entities that make up the application. Resources include disk groups and file systems, network interface cards (NIC), IP addresses, and applications.

Resource dependencies indicate resources that depend on each other because of application or operating system requirements. Resource dependencies are graphically depicted in a hierarchy, also called a tree, where the resources higher up (parent) depend on the resources lower down (child).

[Figure 1-1](#) shows the hierarchy for a database application.

Figure 1-1 Sample resource dependency graph



Resource dependencies determine the order in which resources are brought online or taken offline. For example, a disk group must be imported before volumes in the disk group start, and volumes must start before file systems are

mounted. Conversely, file systems must be unmounted before volumes stop, and volumes must stop before disk groups are deported.

A parent is brought online after each child is brought online, and so on up the tree, until finally the application is started. Conversely, to take a managed application offline, you stop resources beginning at the top of the hierarchy. In this example, the application is stopped first, followed by the database application. Next the IP address and file systems can be stopped concurrently. These resources do not have any resource dependency between them, and so on down the tree.

Child resources must be online before parent resources can be brought online. Parent resources must be taken offline before child resources can be taken offline. If resources do not have parent-child interdependencies, they can be brought online or taken offline concurrently.

Categories of resources

Different types of resources require different levels of control. In VCS there are three categories of resources:

- **On-Off.** VCS starts and stops On-Off resources as required. For example, VCS imports a disk group when required, and deports it when it is no longer needed.
- **On-Only.** VCS starts On-Only resources, but does not stop them. For example, VCS requires NFS daemons to be running to export a file system. VCS starts the daemons if required, but does not stop them if the associated service group is taken offline.
- **Persistent.** These resources cannot be brought online or taken offline. For example, a network interface card cannot be started or stopped, but it is required to configure an IP address. A Persistent resource has an operation value of None. VCS monitors Persistent resources to ensure their status and operation. Failure of a Persistent resource triggers a service group failover.

Resource types

VCS defines a resource type for each resource it manages. For example, the NIC resource type can be configured to manage network interface cards. Similarly, all IP addresses can be configured using the IP resource type.

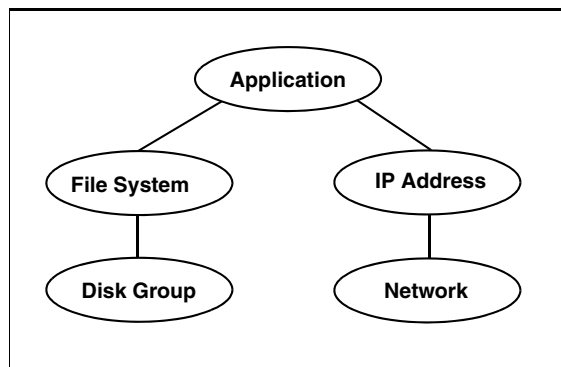
VCS includes a set of predefined resources types. For each resource type, VCS has a corresponding agent, which provides the logic to control resources.

See “[About agents in VCS](#)” on page 27.

Service groups

A service group is a virtual container that contains all the hardware and software resources that are required to run the managed application. Service groups allow VCS to control all the hardware and software resources of the managed application as a single unit. When a failover occurs, resources do not fail over individually— the entire service group fails over. If there is more than one service group on a system, a group may fail over without affecting the others.

Figure 1-2 Typical database service group



A single node may host any number of service groups, each providing a discrete service to networked clients. If the server crashes, all service groups on that node must be failed over elsewhere.

Service groups can be dependent on each other. For example a finance application may be dependent on a database application. Because the managed application consists of all components that are required to provide the service, service group dependencies create more complex managed applications. When you use service group dependencies, the managed application is the entire dependency tree.

See “[About service group dependencies](#)” on page 410.

Types of service groups

VCS service groups fall in three main categories: *failover*, *parallel*, and *hybrid*.

Failover service groups

A failover service group runs on one system in the cluster at a time. Failover groups are used for most applications that do not support multiple systems to simultaneously access the application's data.

Parallel service groups

A parallel service group runs simultaneously on more than one system in the cluster. A parallel service group is more complex than a failover group. Parallel service groups are appropriate for applications that manage multiple application instances running simultaneously without data corruption.

Hybrid service groups

A hybrid service group is for replicated data clusters and is a combination of the failover and parallel service groups. It behaves as a failover group *within* a system zone and a parallel group *across* system zones.

See “[About system zones](#)” on page 367.

A hybrid service group cannot fail over across system zones. VCS allows a switch operation on a hybrid group only if both systems are within the same system zone. If there are no systems within a zone for failover, VCS invokes the nofailover trigger on the lowest numbered node. Hybrid service groups adhere to the same rules governing group dependencies as do parallel groups.

See “[About service group dependencies](#)” on page 410.

See “[nofailover event trigger](#)” on page 452.

About the ClusterService group

The ClusterService group is a special purpose service group, which contains resources that are required by VCS components. The group contains resources for:

- Cluster Management Console
 - Notification
 - wide-area connector (WAC) process, which is used in global clusters
- The ClusterService group can fail over to any node despite restrictions such as frozen. The ClusterService group the first service group to come online and cannot be autodisabled. The group comes online on the first node that goes in the running state. The VCS engine discourages taking the group offline manually.

About the cluster UUID

When you install VCS using the product installer, the installer generates a universally unique identifier (UUID) for the cluster. This value is the same across all the nodes in the cluster. The value is defined in the CID attribute.

See [“Cluster attributes”](#) on page 688.

If do not use the product installer to install VCS, you must run the `uuidconfig.pl` utility to configure the UUID for the cluster.

See [“Configuring and unconfiguring the cluster UUID value”](#) on page 232.

About agents in VCS

Agents are multi-threaded processes that provide the logic to manage resources. VCS has one agent per resource type. The agent monitors all resources of that type; for example, a single IP agent manages all IP resources.

When the agent is started, it obtains the necessary configuration information from VCS. It then periodically monitors the resources, and updates VCS with the resource status.

The action to bring a resource online or take it offline differs significantly for each resource type. For example, bringing a disk group online requires importing the disk group. But, bringing a database online requires starting the database manager process and issuing the appropriate startup commands.

VCS monitors resources when they are online *and* offline to ensure they are not started on systems on which they are not supposed to run. For this reason, VCS starts the agent for any resource that is configured to run on a system when the cluster is started. If no resources of a particular type are configured, the agent is

not started. For example, if there are no Oracle resources in your configuration, the Oracle agent is not started on the system.

Certain agents can identify when an application has been intentionally shut down outside of VCS control.

For agents that support this functionality, if an administrator intentionally shuts down an application outside of VCS control, VCS does not treat it as a fault. VCS sets the service group state as offline or partial, depending on the state of other resources in the service group.

This feature allows administrators to stop applications without causing a failover. The feature is available for V51 agents.

See also [“VCS behavior for resources that support the intentional offline functionality”](#) on page 372.

Agent functions

Agents carry out specific functions on resources. The functions an agent performs are called *entry points*. For details on any of the following agent functions, see the *Veritas Cluster Server Agent Developer's Guide*.

- **Online**—Brings a specific resource ONLINE from an OFFLINE state.
- **Offline**—Takes a resource from an ONLINE state to an OFFLINE state.
- **Monitor**—Tests the status of a resource to determine if the resource is online or offline. The function runs at the following times:
 - During initial node startup, to probe and determine status of all resources on the system.
 - After every online and offline operation.
 - Periodically, to verify that the resource remains in its correct state. Under normal circumstances, the monitor entry point is run every 60 seconds when a resource is online. The entry point is run every 300 seconds when a resource is expected to be offline.
- **Clean**—Cleans up after a resource fails to come online, fails to go offline, or fails while in an ONLINE state. The clean entry point is designed to clean up after an application. The function ensures that the host system is returned to a valid state. For example, the clean function may remove shared memory segments or IPC resources that are left behind by a database.
- **Action**—Performs actions that can be completed in a short time and which are outside the scope of traditional activities such as online and offline. Some agents have predefined action scripts that you can run by invoking the action function.
- **Info**—Retrieves specific information for an online resource.

The retrieved information is stored in the resource attribute `ResourceInfo`. This function is invoked periodically by the agent framework when the resource type attribute `InfoInterval` is set to a non-zero value. The `InfoInterval` attribute indicates the period after which the info function must be invoked. For example, the Mount agent may use this function to indicate the space available on the file system.

Agent classifications

Bundled agents

Bundled agents are packaged with VCS. They include agents for Disk, Mount, IP, and various other resource types.

See the *Veritas Cluster Server Bundled Agents Reference Guide*.

Enterprise agents

Enterprise agents control third party applications and are licensed separately. These include agents for Oracle, Sybase, and DB2. Contact your sales representative for more information.

Custom agents

Custom agents are agents that customers or by Symantec consultants develop. Typically, agents are developed because the user requires control of an application that the current bundled or enterprise agents do not support.

See the *Veritas Cluster Server Agent Developer's Guide*.

About the VCS agent framework

The VCS agent framework is a set of common, predefined functions that are compiled into each agent. These functions include the ability to connect to the VCS engine (HAD) and to understand common configuration attributes. The agent framework frees the developer from developing functions for the cluster; the developer can instead focus on controlling a specific resource type.

For more information on developing agents, see the *Veritas Cluster Server Agent Developer's Guide*.

About cluster control, communications, and membership

Cluster communications ensure that VCS is continuously aware of the status of each system's service groups and resources. They also enable VCS to recognize which systems are active members of the cluster, which have joined or left the cluster, and which have failed.

About the high availability daemon (HAD)

The VCS high availability daemon (HAD) runs on each system. Also known as the VCS engine, HAD is responsible for:

- building the running cluster configuration from the configuration files
- distributing the information when new nodes join the cluster
- responding to operator input
- taking corrective action when something fails.

The engine uses agents to monitor and manage resources. It collects information about resource states from the agents on the local system and forwards it to all cluster members.

The local engine also receives information from the other cluster members to update its view of the cluster. HAD operates as a *replicated state machine* (RSM). The engine running on each node has a completely synchronized view of the resource status on each node. Each instance of HAD follows the same code path for corrective action, as required.

The RSM is maintained through the use of a purpose-built communications package. The communications package consists of the protocols *Low Latency Transport* (LLT) and *Group Membership Services/Atomic Broadcast* (GAB).

See “[About inter-system cluster communications](#)” on page 283.

The hashadow process monitors HAD and restarts it when required.

About the HostMonitor daemon

VCS also starts HostMonitor daemon when the VCS engine comes up. The VCS engine creates a VCS resource VCS_{hm} of type HostMonitor and a VCS_{hmg} service group. The VCS engine does not add these objects to the main.cf file. Do not modify or delete these components of VCS. VCS uses the HostMonitor daemon to monitor the resource utilization of CPU and Swap. VCS reports to the engine log if the resources cross the threshold limits that are defined for the resources.

You can control the behavior of the HostMonitor daemon using the HostMonLogLvl attribute.

See “[Cluster attributes](#)” on page 688.

About Group Membership Services/Atomic Broadcast (GAB)

The Group Membership Services/Atomic Broadcast protocol (GAB) is responsible for cluster membership and cluster communications.

- Cluster Membership

GAB maintains cluster membership by receiving input on the status of the heartbeat from each node by LLT. When a system no longer receives heartbeats from a peer, it marks the peer as DOWN and excludes the peer from the cluster. In VCS, memberships are sets of systems participating in the cluster. VCS has the following types of membership:

- A regular membership includes systems that communicate with each other across one or more network channels.
- A jeopardy membership includes systems that have only one private communication link.
- Daemon Down Node Alive (DDNA) is a condition in which the VCS high availability daemon (HAD) on a node fails, but the node is running. In a DDNA condition, VCS does not have information about the state of service groups on the node. So, VCS places all service groups that were online on the affected node in the autodisabled state. The service groups that were online on the node cannot fail over. Manual intervention is required to enable failover of autodisabled service groups. The administrator must release the resources running on the affected node, clear resource faults, and bring the service groups online on another node.
- Cluster Communications
GAB's second function is reliable cluster communications. GAB provides guaranteed delivery of point-to-point and broadcast messages to all nodes. The VCS engine uses a private IOCTL (provided by GAB) to tell GAB that it is alive.

About Low Latency Transport (LLT)

VCS uses private network communications between cluster nodes for cluster maintenance. The Low Latency Transport functions as a high-performance, low-latency replacement for the IP stack, and is used for all cluster communications. Symantec recommends two independent networks between all cluster nodes. These networks provide the required redundancy in the communication path and enable VCS to discriminate between a network failure and a system failure. LLT has two major functions.

- Traffic Distribution
LLT distributes (load balances) internode communication across all available private network links. This distribution means that all cluster communications are evenly distributed across all private network links (maximum eight) for performance and fault resilience. If a link fails, traffic is redirected to the remaining links.
- Heartbeat

LLT is responsible for sending and receiving heartbeat traffic over network links. The Group Membership Services function of GAB uses this heartbeat function to determine cluster membership.

About the I/O fencing module

The I/O fencing module implements a quorum-type functionality to ensure that only one cluster survives a split of the private network. I/O fencing also provides the ability to perform SCSI-3 persistent reservations on failover. I/O fencing offers complete protection against data corruption by nodes that are assumed to be excluded from cluster membership.

See “[About the I/O fencing algorithm](#)” on page 308.

About security services

VCS uses the Symantec Product Authentication Service to provide secure communication between cluster nodes and clients, including the Java and the Web consoles. VCS uses digital certificates for authentication and uses SSL to encrypt communication over the public network.

In secure mode:

- VCS uses platform-based authentication.
- VCS does not store user passwords.
- All VCS users are system and domain users and are configured using fully-qualified user names. For example, administrator@vcsdomain. VCS provides a single sign-on mechanism, so authenticated users need not sign on each time to connect to a cluster.

VCS requires a system in your enterprise to be configured as a *root broker*. Additionally, all nodes in the cluster must be configured as *authentication brokers*.

- A root broker serves as the main registration and certification authority; it has a self-signed certificate and can authenticate other brokers. The root broker may be a system in the cluster. Symantec recommends having a single root broker per domain, typically a datacenter, acting as root broker for all products using Symantec Product Authentication Services. The root broker is only used during initial creation of an authentication broker.
- Authentication brokers serve as intermediate registration and certification authorities. An authentication broker authenticates users, such as a login to the cluster, or services, such as daemons running on application nodes, but cannot authenticate other brokers. Authentication brokers have certificates that are signed by the root. Each node in VCS serves as an authentication broker.

Security credentials for the authentication broker must be obtained from the root broker.

For secure communication, VCS components acquire credentials from the authentication broker that is configured on the local system. The acquired certificate is used during authentication and is presented to clients for the SSL handshake.

VCS and its components specify the account name and the domain in the following format:

■ **HAD Account**

```
name = _HA_VCS_(systemname)
domain = HA_SERVICES@(fully_qualified_system_name)
```

■ **CmdServer**

```
name = _CMDSERVER_VCS_(systemname)
domain = HA_SERVICES@(fully_qualified_system_name)
```

For instructions on how to set up Security Services while setting up the cluster, see the *Veritas Cluster Server Installation Guide*.

You can also enable and disable Security Services manually

See “[Enabling and disabling Security Services](#)” on page 237.

Components for administering VCS

VCS provides the following components to administer clusters:

Veritas Cluster Server Management Console

A Web-based graphical user interface for monitoring and administering the cluster.

- Install VCS Management Console on cluster nodes to manage a single cluster.
- Install VCS Management Console on a management server outside the cluster to manage multiple clusters.
See the *Veritas Cluster Server Management Console Implementation Guide* for more information.

Cluster Manager (Java console)

A cross-platform Java-based graphical user interface that provides complete administration capabilities for your cluster. The console runs on any system inside or outside the cluster, on any operating system that supports Java.

See “[Administering the cluster from Cluster Manager \(Java console\)](#)” on page 79.

VCS command line interface (CLI)

The VCS command-line interface provides a comprehensive set of commands for managing and administering the cluster.

See “[Administering the cluster from the command line](#)” on page 179.

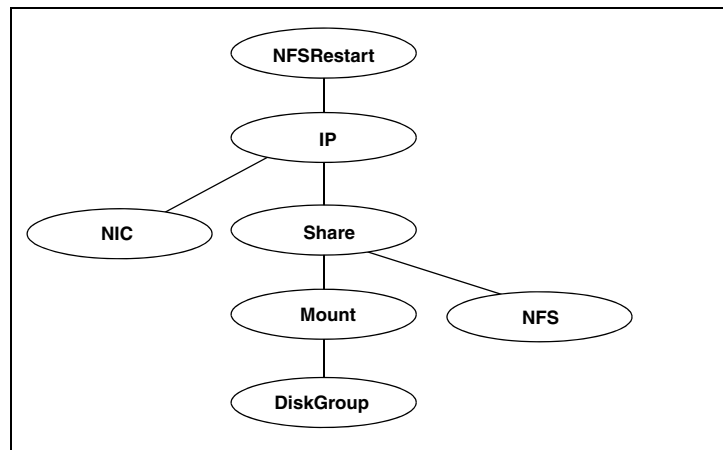
Putting the pieces together

In this example, a two-node cluster exports an NFS file system to clients. Both nodes are connected to shared storage, which enables them to access the directories being shared. A single service group, NFS_Group, fails over between System A and System B, as necessary.

The VCS engine, HAD, reads the configuration file, determines what agents are required to control the resources in the service group, and starts the agents. HAD uses resource dependencies to determine the order in which to bring the resources online. VCS issues online commands to the corresponding agents in the correct order.

The following figure shows the dependency graph for the service group NFS_Group.

Figure 1-3 Dependency graph for the sample NFS group



VCS starts the agents for disk group, mount, share, NFS, NIC, and IP on all systems that are configured to run NFS_Group. The resource dependencies are configured as:

- The /home file system (configured as a Mount resource), requires the disk group (configured as a DiskGroup resource) to be online before mounting.
- The NFS export of the home file system (Share) requires the file system to be mounted and the NFS daemons (NFS) be running.
- The high availability IP address, nfs_IP, requires the file system (Share) to be shared and the network interface (NIC) to be up.
- The NFSRestart resource requires the IP address to be up.

- The NFS daemons and the disk group have no child dependencies, so they can start in parallel.
- The NIC resource is a persistent resource and does not require starting.

The service group can be configured to start automatically on either node in the preceding example. It can then move or fail over to the second node on command or automatically if the first node fails. On failover or relocation, VCS starts offline the resources beginning at the top of the graph and start them on the second node beginning at the bottom.

About cluster topologies

- [Basic failover configurations](#)
- [Advanced failover configurations](#)
- [Cluster topologies and storage configurations](#)

Basic failover configurations

This section describes basic failover configurations, including asymmetric, symmetric, and N-to-1.

Asymmetric or Active/Passive configuration

In an asymmetric configuration, an application runs on a primary, or master, server. A dedicated redundant server is present to take over on any failure. The redundant server is not configured to perform any other functions. In the following illustration, a database application is moved, or failed over, from the master to the redundant server.

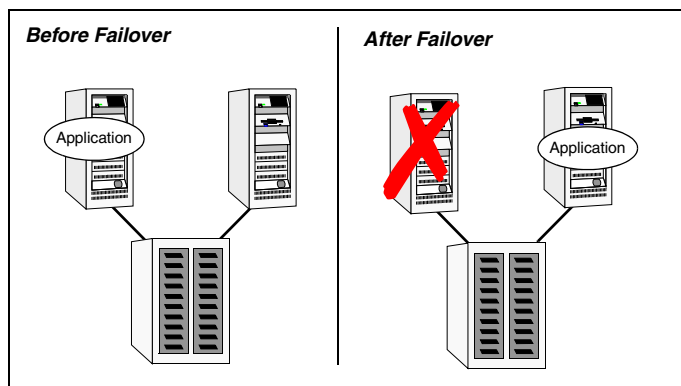


Figure 2-1 Asymmetric failover

This configuration is the simplest and most reliable. The redundant server is on stand-by with full performance capability. If other applications are running, they present no compatibility issues.

Symmetric or Active/Active configuration

In a symmetric configuration, each server is configured to run a specific application or service and provide redundancy for its peer. In this example, each server runs one application service group. When a failure occurs, the surviving server hosts both application groups.

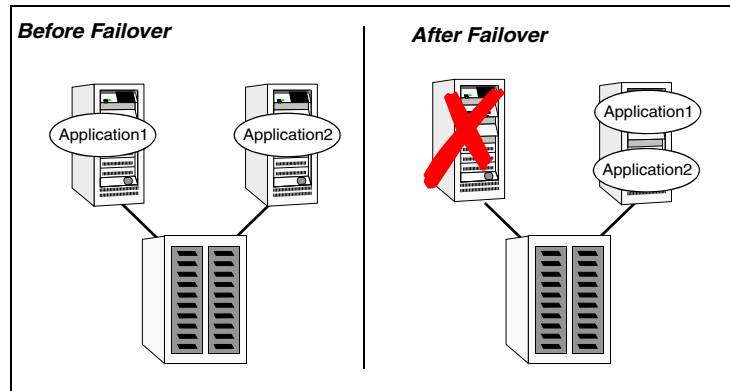


Figure 2-2 Symmetric failover

Symmetric configurations appear more efficient in terms of hardware utilization. In the asymmetric example, the redundant server requires only as much processor power as its peer. On failover, performance remains the same. In the symmetric example, the redundant server requires adequate processor power to run the existing application and the new application it takes over.

Further issues can arise in symmetric configurations when multiple applications running on the same system do not co-exist properly. Some applications work well with multiple copies started on the same system, but others fail. Issues can also arise when two applications with different I/O and memory requirements run on the same system.

N-to-1 configuration

An N-to-1 failover configuration reduces the cost of hardware redundancy and still provides a potential, dedicated spare. In an asymmetric configuration there is no performance penalty. There are no issues with multiple applications running on the same system; however, the drawback is the 100 percent redundancy cost at the server level.

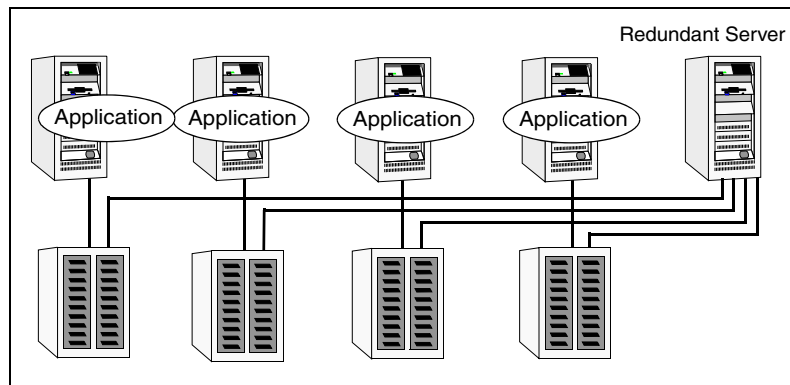


Figure 2-3 N-to-1 configuration

An N-to-1 configuration is based on the concept that multiple, simultaneous server failures are unlikely; therefore, a single redundant server can protect multiple active servers. When a server fails, its applications move to the redundant server. For example, in a 4-to-1 configuration, one server can protect four servers. This configuration reduces redundancy cost at the server level from 100 percent to 25 percent. In this configuration, a dedicated, redundant server is cabled to all storage and acts as a spare when a failure occurs.

The problem with this design is the issue of *failback*. When the failed server is repaired, all services that are hosted on the server must be failed back to the server. The failback action frees the spare server and restores redundancy to the cluster.

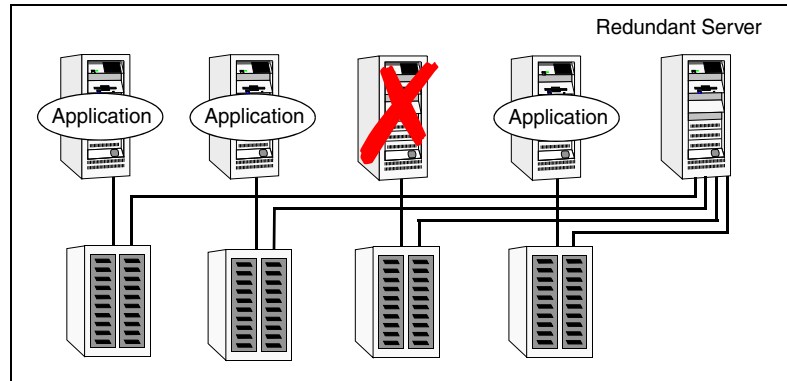


Figure 2-4 N-to-1 failover requiring failback

Most shortcomings of early N-to-1 cluster configurations were caused by the limitations of storage architecture. Typically, it was impossible to connect more than two hosts to a storage array without complex cabling schemes and their inherent reliability problems, or resorting to expensive arrays with multiple controller ports.

Advanced failover configurations

This section describes advanced failover configurations for VCS.

N + 1 configuration

With the capabilities introduced by storage area networks (SANs), you cannot only create larger clusters, but can connect multiple servers to the same storage.

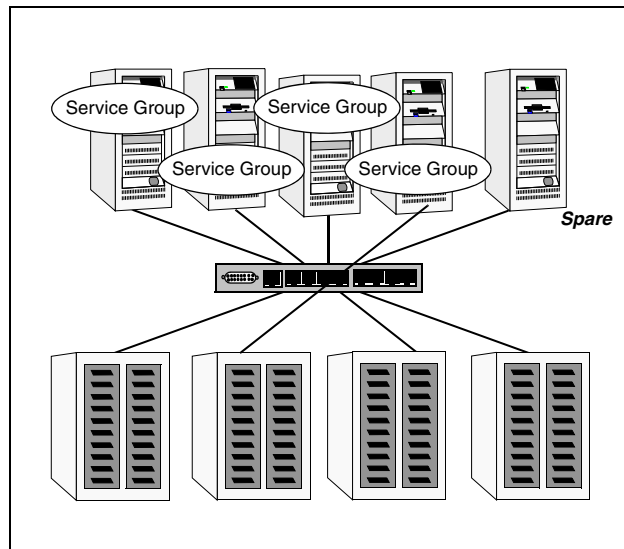


Figure 2-5 N+1 configuration

A dedicated, redundant server is no longer required in the configuration. Instead of N-to-1 configurations, you can use an *N+1* configuration. In advanced N+1 configurations, an extra server in the cluster is spare capacity only.

When a server fails, the application service group restarts on the spare. After the server is repaired, it becomes the spare. This configuration eliminates the need for a second application failure to fail back the service group to the primary system. Any server can provide redundancy to any other server.

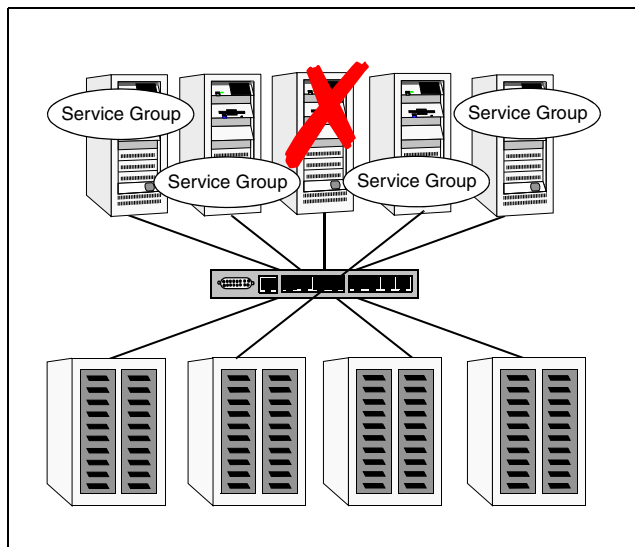
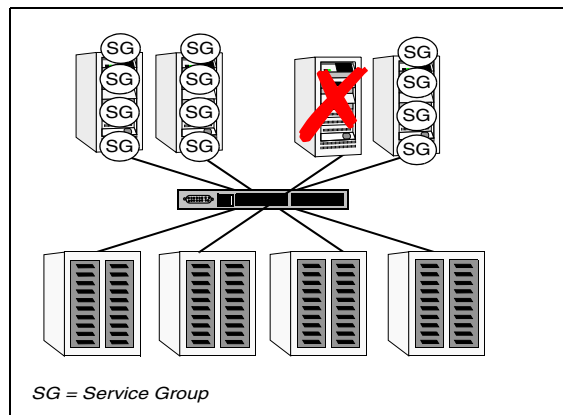


Figure 2-6 N+1 failover

N-to-N configuration

An N-to-N configuration refers to multiple service groups running on multiple servers, with each service group capable of being failed over to different servers. For example, consider a four-node cluster with each node supporting three critical database instances.

Figure 2-7 N-to-N configuration



If any node fails, each instance is started on a different node, ensuring that no single node becomes overloaded. This configuration is a logical evolution of N + 1: it provides cluster *standby capacity* instead of a *standby server*.

N-to-N configurations require careful testing to ensure that all applications are compatible. Applications must also have complete control of where service groups fail when an event occurs.

Cluster topologies and storage configurations

This section describes commonly-used cluster topologies, along with the storage configuration used to support the topologies.

Basic shared storage cluster

In this configuration, a single cluster shares access to a storage device, typically over a SAN. An application can only be started on a node with access to the required storage. For example, in a multi-node cluster, any node that is designated to run a specific database instance must have access to the storage where the database's tablespaces, redo logs and control files are stored. Shared disk architecture is also the easiest to implement and maintain. When a node or application fails, all data that is required to start on another node is stored on the shared disk.

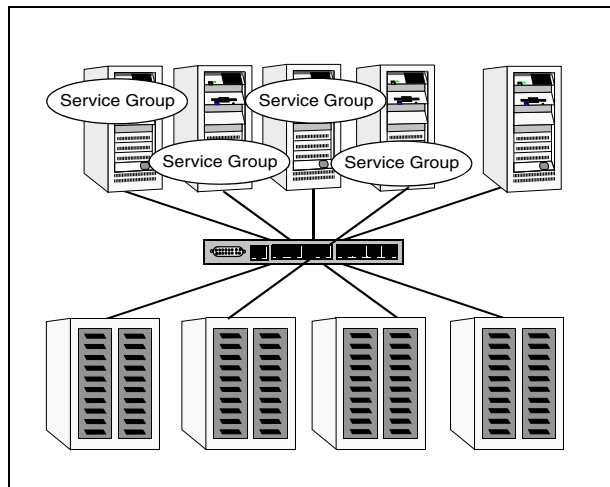
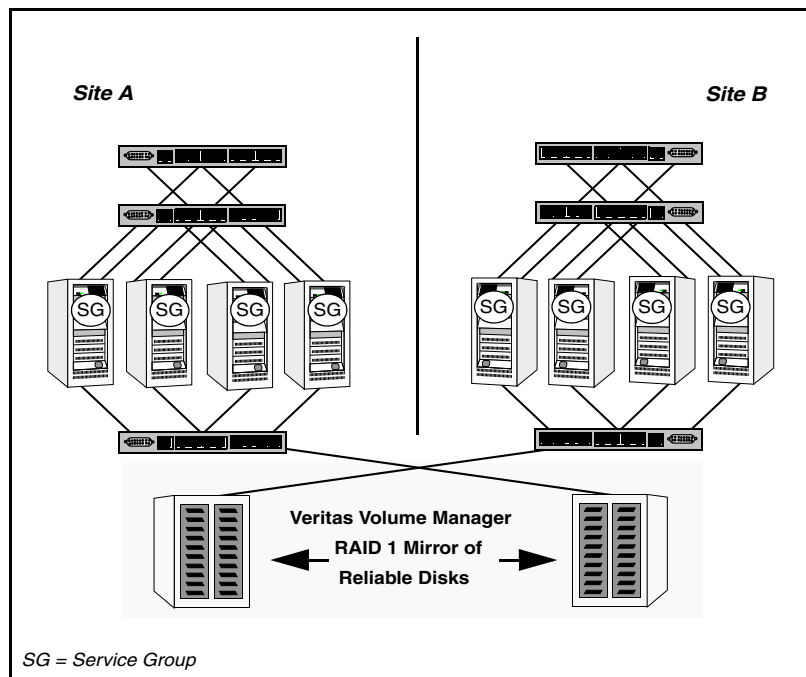


Figure 2-8 Shared disk architecture for basic cluster

Campus, or Metropolitan, shared storage cluster

In a campus environment, VCS and Veritas Volume Manager are used to create a cluster that spans multiple datacenters or buildings. Instead of a single storage array, data is mirrored between arrays using Veritas Volume Manager. This configuration provides synchronized copies of data at both sites. This procedure is identical to mirroring between two arrays in a datacenter, only now it is spread over a distance.

Figure 2-9 Campus shared storage cluster



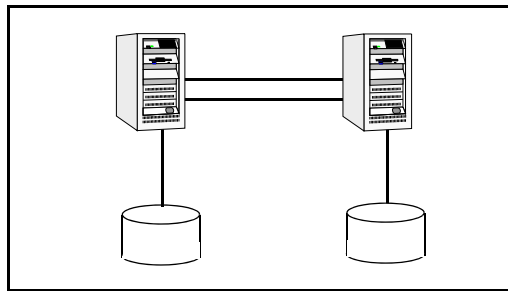
A campus cluster requires two independent network links for heartbeat, two storage arrays each providing highly available disks, and public network connectivity between buildings on same IP subnet. If the campus cluster setup resides on different subnets with one for each site, then use the VCS DNS agent to handle the network changes or issue the DNS changes manually.

See [“How VCS campus clusters work”](#) on page 550.

Shared nothing cluster

Systems in shared nothing clusters do not share access to disks; they maintain separate copies of data. VCS shared nothing clusters typically have read-only data stored locally on both systems. For example, a pair of systems in a cluster that includes a critical Web server, which provides access to a backend database. The Web server runs on local disks and does not require data sharing at the Web server level.

Figure 2-10 Shared nothing cluster

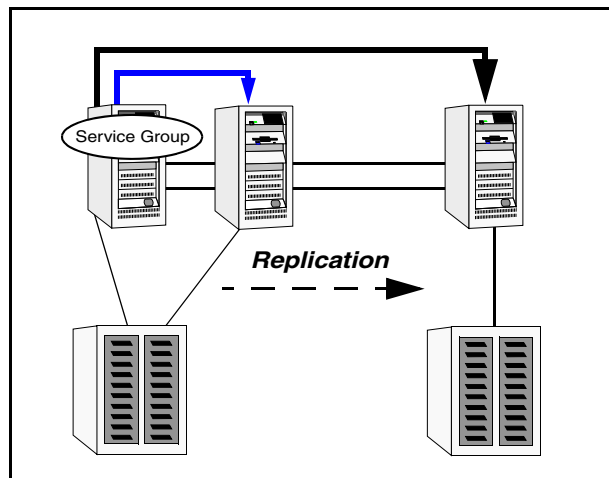


Replicated data cluster

In a replicated data cluster there is no shared disk. Instead, a data replication product synchronizes copies of data between nodes. Replication can take place at the application, host, and storage levels. Application-level replication products, such as Oracle DataGuard, maintain consistent copies of data between systems at the SQL or database levels. Host-based replication products, such as Veritas Volume Replicator, maintain consistent storage at the logical volume level. Storage- or array-based replication maintains consistent copies of data at the disk or RAID LUN level.

The following illustration shows a hybrid shared storage/replicated data cluster, in which different failover priorities are assigned to nodes according to particular service groups.

Figure 2-11 Shared storage replicated data cluster



Replicated data clusters can also be configured without the ability to fail over locally, but this configuration is not recommended.

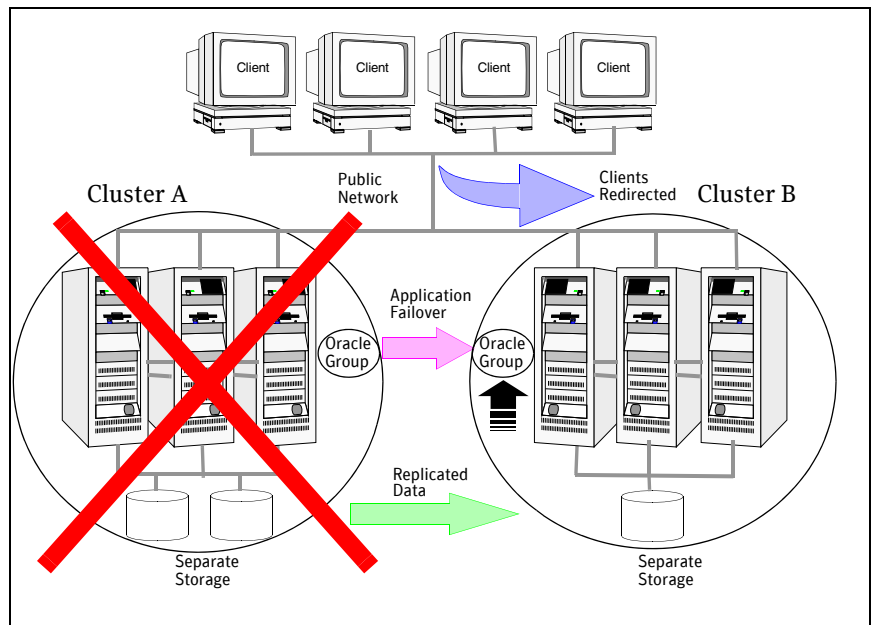
See [“How VCS replicated data clusters work”](#) on page 541.

Global cluster

A global cluster links clusters at separate locations and enables wide-area failover and disaster recovery.

Local clustering provides local failover for each site or building. Campus and replicated cluster configurations offer protection against disasters affecting limited geographic regions. Large scale disasters such as major floods, hurricanes, and earthquakes can cause outages for an entire city or region. In such situations, you can ensure data availability by migrating applications to sites located considerable distances apart.

Figure 2-12 Global cluster



In a global cluster, if an application or a system fails, the application is migrated to another system within the same cluster. If the entire cluster fails, the application is migrated to a system in another cluster. Clustering on a global level also requires replicating shared data to the remote site.

See [“How VCS global clusters work”](#) on page 466.

VCS configuration concepts

- [About configuring VCS](#)
- [About the VCS configuration language](#)
- [About the main.cf file](#)
- [The types.cf file](#)
- [About VCS attributes](#)
- [About VCS keywords and reserved words](#)
- [VCS environment variables](#)

About configuring VCS

Configuring VCS means conveying to the VCS engine the definitions of the cluster, service groups, resources, and resource dependencies. VCS uses two configuration files in a default configuration:

- `main.cf`—Defines the cluster, including services groups and resources.
- `types.cf`—Defines the resource types.

By default, both files reside in the following directory:

`/etc/VRTSvcs/conf/config`

Additional files similar to `types.cf` may be present if agents have been added, such as `OracleTypes.cf`.

In a VCS cluster, the first system to be brought online reads the configuration file and creates an internal (in-memory) representation of the configuration. Systems brought online after the first system derive their information from systems running in the cluster.

You must stop the cluster while you are modifying the files from the command line. Changes made by editing the configuration files take effect when the cluster is restarted. The node on which the changes were made should be the first node to be brought back online.

About the VCS configuration language

The VCS configuration language specifies the makeup of service groups and their associated entities, such as resource types, resources, and attributes. These specifications are expressed in configuration files, whose names contain the suffix `.cf`.

There are several ways to generate configuration files:

- Use the web-based Cluster Management Console
- Use Cluster Manager (Java Console).
- Use the command-line interface.
- If VCS is not running, use a text editor to create and modify the files.

About the main.cf file

The format of the main.cf file comprises include clauses and definitions for the cluster, systems, service groups, and resources. The main.cf file also includes service group and resource dependency clauses.

- **Include clauses**—Include clauses incorporate additional configuration files into main.cf. These additional files typically contain type definitions, including the types.cf file. Typically, custom agents add type definitions in their own files.

```
include "types.cf"
```

“[About including multiple .cf files in main.cf](#)” on page 55.

- **Cluster definition**—Defines the attributes of the cluster, including the cluster name and the names of the cluster users.

```
cluster demo (
  UserNames = { admin = cDRpdxPmHzpS }
)
```

See “[Cluster attributes](#)” on page 688.

- **System definition**—Lists the systems designated as part of the cluster. The system names must match the name returned by the command `uname -a`. Each service group can be configured to run on a subset of systems defined in this section.

```
system Server1
system Server2
```

- **Service group definition**—Service group definitions in main.cf comprise the attributes of a particular service group.

```
group NFS_group1 (
  SystemList = { Server1, Server2 }
  AutoStartList = { Server1 }
)
```

See “[Service group attributes](#)” on page 663.

See “[About the SystemList attribute](#)” on page 54.

- **Resource definition**—Defines each resource used in a particular service group. Resources can be added in any order and the utility hacf arranges the resources alphabetically the first time the configuration file is run.

```
DiskGroup DG_shared1 (
  DiskGroup = shared1
)
```

- **Resource dependency clause**—Defines a relationship between resources. A dependency is indicated by the keyword `requires` between two resource names.

```
IP_resource requires NIC_resource
```

See “[Resources and resource dependencies](#)” on page 23.

- Service group dependency clause—To configure a service group dependency, place the keyword `requires` in the service group declaration of the `main.cf` file. Position the dependency clause before the resource dependency specifications and after the resource declarations.

```
group_x requires group_y
```

See “[About service group dependencies](#)” on page 410.

About the SystemList attribute

The `SystemList` attribute designates all systems on which a service group can come online. By default, the order of systems in the list defines the priority of systems used in a failover. For example, the following definition configures `SystemA` to be the first choice on failover, followed by `SystemB` and then `SystemC`.

```
SystemList = { SystemA, SystemB, SystemC }
```

System priority may also be assigned explicitly in the `SystemList` attribute by assigning numeric values to each system name. For example:

```
SystemList = { SystemA=0, SystemB=1, SystemC=2 }
```

If you do not assign numeric priority values, VCS assigns a priority to the system without a number by adding 1 to the priority of the preceding system. For example, if the `SystemList` is defined as follows, VCS assigns the values `SystemA = 0`, `SystemB = 2`, `SystemC = 3`.

```
SystemList = { SystemA, SystemB=2, SystemC },
```

Note that a duplicate numeric priority value may be assigned in some situations:

```
SystemList = { SystemA, SystemB=0, SystemC }
```

The numeric values assigned are `SystemA = 0`, `SystemB = 0`, `SystemC = 1`.

To avoid this situation, do not assign any numbers or assign different numbers to each system in `SystemList`.

Initial configuration

When VCS is installed, a basic `main.cf` configuration file is created with the cluster name, systems in the cluster, and a Cluster Manager user named *admin* with the password *password*.

The following is an example of the `main.cf` for cluster demo and systems `SystemA` and `SystemB`.

```
include "types.cf"
cluster demo (
  UserNames = { admin = cDRpdxPmHzpS }
)
system SystemA
system SystemB
```

About including multiple .cf files in main.cf

You may choose include several configuration files in the main.cf file. For example:

```
include "applicationtypes.cf"
include "listofsystems.cf"
include "applicationgroup.cf"
```

If you include other .cf files in main.cf, the following considerations apply:

- Resource type definitions must appear before the definitions of any groups that use the resource types.

In the following example, the applicationgroup.cf file includes the service group definition for an application. The service group includes resources whose resource types are defined in the file applicationtypes.cf. In this situation, the applicationtypes.cf file must appear first in the main.cf file. For example:

```
include "applicationtypes.cf"
include "applicationgroup.cf"
```

- If you define heartbeats outside of the main.cf file and include the heartbeat definition file, saving the main.cf file results in the heartbeat definitions getting added directly to the main.cf file.

The types.cf file

The types.cf file describes standard resource types to the VCS engine; specifically, the data required to control a specific resource.

The following example illustrates a DiskGroup resource type definition for AIX.

```
type DiskGroup (
    static keylist SupportedActions = { "license.vfd",
        disk.vfd", numdisks }
    static int OnlineRetryLimit = 1
    static str ArgList[] = { DiskGroup, StartVolumes,
        StopVolumes, MonitorOnly, MonitorReservation,
        tempUseFence }
    str DiskGroup
    str StartVolumes = 1
    str StopVolumes = 1
    static int NumThreads = 1
    boolean MonitorReservation = 0
    temp str tempUseFence = "INVALID"
    boolean PanicSystemOnDGLoss = 0
)
```

The types definition performs two important functions:

- Defines the type of values that may be set for each attribute.

In the DiskGroup example, the NumThreads and OnlineRetryLimit attributes are both classified as int, or integer. The DiskGroup, StartVolumes and StopVolumes attributes are defined as str, or strings. See “[Attribute data types](#)” on page 57.

- Defines the parameters passed to the VCS engine through the ArgList attribute. The line static str ArgList[] = { xxx, yyy, zzz } defines the order in which parameters are passed to the agents for starting, stopping, and monitoring resources.

For another example, review the following main.cf and types.cf representing an IP resource:

main.cf for AIX

```
IP nfs_ip1 (  
    Device = en0  
    Address = "192.168.1.201"  
)
```

types.cf for AIX

```
type IP (  
    static keylist SupportedActions = { "device.vfd", "route.vfd" }  
    static str ArgList[] = { Device, Address, NetMask, Options }  
    str Device  
    str Address  
    str NetMask  
    str Options  
)
```

The high-availability address is configured on the interface defined by the Device attribute.

The IP address is enclosed in double quotes because the string contains periods.

See “[Attribute data types](#)” on page 57.

The VCS engine passes the identical arguments to the IP agent for online, offline, clean and monitor. It is up to the agent to use the arguments it requires. All resource names must be unique in a VCS cluster.

About VCS attributes

VCS components are configured using *attributes*. Attributes contain data about the cluster, systems, service groups, resources, resource types, agent, and heartbeats if using global clusters. For example, the value of a service group's SystemList attribute specifies on which systems the group is configured and the priority of each system within the group. Each attribute has a definition and a value. Attributes also have default values assigned when a value is not specified.

Attribute data types

VCS supports the following data types for attributes.

String	A string is a sequence of characters enclosed by double quotes. A string may also contain double quotes, but the quotes must be immediately preceded by a backslash. A backslash is represented in a string as \\. Quotes are not required if a string begins with a letter, and contains only letters, numbers, dashes (-), and underscores (_). For example, a string defining a network interface such as hme0 or eth0 does not require quotes as it contains only letters and numbers. However a string defining an IP address contains periods and requires quotes, such as: 192.168.100.1
Integer	Signed integer constants are a sequence of digits from 0 to 9. They may be preceded by a dash, and are interpreted in base 10. Integers cannot exceed the value of a 32-bit signed integer: 21471183247.
Boolean	A boolean is an integer, the possible values of which are 0 (false) and 1 (true).

Attribute dimensions

VCS attributes have the following dimensions.

Scalar	<p>A scalar has only one value. This is the default dimension.</p>
Vector	<p>A vector is an ordered list of values. Each value is indexed using a positive integer beginning with zero. Use a comma (,) or a semi-colon (;) to separate values. A set of brackets ([]) after the attribute name denotes that the dimension is a vector.</p> <p>For example, an agent's ArgList is defined as:</p> <pre>static str ArgList[] = {RVG, DiskGroup, Primary, SRL, Links}</pre>
Keylist	<p>A keylist is an unordered list of strings, and each string is unique within the list. Use a comma (,) or a semi-colon (;) to separate values.</p> <p>For example, to designate the list of systems on which a service group will be started with VCS (usually at system boot):</p> <pre>AutoStartList = {SystemA; SystemB; SystemC}</pre>
Association	<p>An association is an unordered list of name-value pairs. Use a comma (,) or a semi-colon (;) to separate values.</p> <p>A set of braces ({}) after the attribute name denotes that an attribute is an association.</p> <p>For example, to designate the list of systems on which the service group is configured to run and the system's priorities:</p> <pre>SystemList = {SystemA=1, SystemB=2, SystemC=3}</pre>

Attributes and cluster objects

VCS has the following types of attributes, depending on the cluster object the attribute applies to.

Cluster attributes	<p>Attributes that define the cluster.</p> <p>For example, <code>ClusterName</code> and <code>ClusterAddress</code>.</p>
Service group attributes	<p>Attributes that define a service group in the cluster.</p> <p>For example, <code>Administrators</code> and <code>ClusterList</code>.</p>
System attributes	<p>Attributes that define the system in the cluster.</p> <p>For example, <code>Capacity</code> and <code>Limits</code>.</p>
Resource type attributes	<p>Attributes that define the resource types in VCS. These can be further classified as:</p> <ul style="list-style-type: none">■ Type-independent—Attributes that all agents (or resource types) understand. Examples: <code>RestartLimit</code> and <code>MonitorInterval</code>; these can be set for any resource type. Typically, these attributes are set for all resources of a specific type. For example, setting <code>MonitorInterval</code> for the IP resource type affects all IP resources.■ Type-dependent—Attributes that apply to a particular resource type. These attributes appear in the type definition file (<code>types.cf</code>) for the agent. Example: The <code>Address</code> attribute applies only to the IP resource type. Attributes defined in the file <code>types.cf</code> apply to all resources of a particular resource type. Defining these attributes in the <code>main.cf</code> file overrides the values in the <code>types.cf</code> file for a specific resource. For example, setting <code>StartVolumes = 1</code> for the <code>DiskGroup</code> <code>types.cf</code> defaults <code>StartVolumes</code> to <code>True</code> for all <code>DiskGroup</code> resources. Setting the value in <code>main.cf</code> overrides the value on a per-resource basis.■ Static—These attributes apply for every resource of a particular type. These attributes are prefixed with the term <code>static</code> and are not included in the resource's argument list. You can override some static attributes and assign them resource-specific values. See “Overriding resource type static attributes” on page 240.
Resource attributes	<p>Attributes that define a specific resource.</p> <p>Some of these attributes are type-independent. For example, you can configure the <code>Critical</code> attribute for any resource.</p> <p>Some resource attributes are type-dependent. For example, the <code>Address</code> attribute defines the IP address associated with the IP resource. These attributes are defined in the <code>main.cf</code> file.</p>

Attribute scope across systems: global and local attributes

An attribute whose value applies to all systems is *global* in scope. An attribute whose value applies on a per-system basis is *local* in scope. The at operator (@) indicates the system to which a local value applies.

An example of local attributes can be found in the following resource type where IP addresses and routing options are assigned per machine.

MultiNICA definition for AIX

```
MultiNICA mnic (  
  Device@sysa = { en0 = "166.98.16.103", en3 = "166.98.16.103" }  
  Device@sysb = { en0 = "166.98.16.104", en3 = "166.98.16.104" }  
  NetMask = "255.255.255.0"  
  Options = "mtu m"  
  RouteOptions@sysa = "-net 192.100.201.0 192.100.13.7"  
  RouteOptions@sysb = "-net 192.100.201.1 192.100.13.8"  
)
```

Attribute life: temporary attributes

You can define temporary attributes in the types.cf file. The values of temporary attributes remain in memory as long as VCS is running. Values of temporary attributes are not available when the cluster is restarted. These attribute values are not stored in the main.cf file.

Temporary attributes cannot be converted to permanent, and vice-versa. When you save a configuration, VCS saves temporary attributes and their default values in the file types.cf.

The scope of these attributes can be local to a node or global across all nodes in the cluster. Local attributes can be defined even when the node is not part of a cluster.

You can define and modify these attributes only while VCS is running.

See “[Adding, deleting, and modifying resource attributes](#)” on page 225.

Size limitations for VCS objects

The following VCS objects are restricted to 1024 characters.

- Service group names
- Resource names
- Resource type names
- User names
- Attribute names

VCS passwords are restricted to 255 characters. You can enter a password of maximum 255 characters.

About VCS keywords and reserved words

The following list includes the current keywords reserved for the VCS configuration language. Note they are case-sensitive.

action	false	keylist	remoteclass	static
after	firm	local	requires	stop
ArgListValues	global	offline	resource	str
before	group	online	set	system
boolean	Group	MonitorOnly	Signaled	System
cluster	hard	Name	soft	temp
Cluster	heartbeat	NameRule	start	type
condition	HostMonitor	Path	Start	Type
ConfidenceLevel	int	Probed	state	VCShm
event	IState	remote	State	VCShmg

VCS environment variables

Table 3-1 lists VCS environment variables.

See “Defining VCS environment variables” on page 65.

See also “Environment variables to start and stop VCS modules” on page 65 and

See also “Environment variables associated with the coordination point server” on page 349.

Table 3-1 VCS environment variables

Environment Variable	Definition and Default Value
PERL5LIB	Root directory for Perl executables. (applicable only for Windows) Default: Install Drive:\Program Files\VERITAS\cluster server\lib\perl5.
VCS_CONF	Root directory for VCS configuration files. Default: /etc/VRTSvcs Note: If this variable is added or modified you must reboot the system to apply the changes.
VCS_DOMAIN	The Security domain to which the VCS users belong. Symantec Product Authentication Service uses this environment variable to authenticate VCS users on a remote host. Default: Fully qualified host name of the remote host as defined in the VCS_HOST environment variable or in the .vcshost file.
VCS_DOMAINTYPE	The type of Security domain such as unixpwd, nt, nis, nisplus, ldap, or vx. Symantec Product Authentication Service uses this environment variable to authenticate VCS users on a remote host. Default: unixpwd
VCS_DIAG	Directory where VCS dumps HAD cores and FFDC data.
VCS_ENABLE_LDF	Designates whether or not log data files (LDFs) are generated. If set to a non-zero integer, LDFs are generated. If set to 0, they are not.

Table 3-1 VCS environment variables

Environment Variable	Definition and Default Value
VCS_HOME	Root directory for VCS executables. Default: /opt/VRTSvcs
VCS_HOST	VCS node on which ha commands will be run.
VCS_GAB_PORT	GAB port to which VCS connects. Default: h
VCS_GAB_TIMEOUT	Timeout in milliseconds for HAD to send heartbeats to GAB. Default: 30000 Note: If the specified timeout is exceeded, GAB kills HAD, and all active service groups on system are disabled.
VCS_GAB_RMTIMEOUT	Timeout in seconds for HAD to register with GAB. Default: 200 See “Registration monitoring” on page 572.
VCS_GAB_RMACTION	Controls the GAB behavior when VCS_GAB_RMTIMEOUT exceeds. You can set the value as follows: <ul style="list-style-type: none">■ PANIC—GAB panics the system■ SYSLOG—GAB logs an appropriate message Default: SYSLOG See “Registration monitoring” on page 572.
VCS_HAD_RESTART_TIMEOUT	Set this variable to designate the amount of time the hashadow process waits (sleep time) before restarting HAD. Default: 0
VCS_LOG	Root directory for log files and temporary files. Default: /var/VRTSvcs Note: If this variable is added or modified you must reboot the system to apply the changes.
VCS_SERVICE	Name of configured VCS service. Default: vcs Note: The specified service should be configured before starting the VCS engine (HAD). If a service is not specified, the VCS engine starts with port 14141.

Table 3-1 VCS environment variables

Environment Variable	Definition and Default Value
VCS_TEMP_DIR	<p>Directory in which temporary information required by, or generated by, hacf is stored.</p> <p>Default: /var/VRTSvcs</p> <p>This directory is created in /tmp under the following conditions:</p> <ul style="list-style-type: none">■ The variable is not set.■ The variable is set but the directory to which it is set does not exist.■ The utility hacf cannot find the default location.

Defining VCS environment variables

Define VCS environment variables in the file `vcshenv`, which is located at the path `/opt/VRTSvcs/bin/`. These variables are set for VCS when the `hastart` command is run.

To set a variable, use the syntax appropriate for the shell in which VCS starts. For example, if you use the bash shell, define variables as:

```
export VCS_GAB_TIMEOUT = 18000
export umask = 022
```

By default, files generated by VCS inherit the system's umask settings. To override the system's umask settings for files generated by VCS, define a umask value in the `vcshenv` file.

Environment variables to start and stop VCS modules

The start and stop environment variables for LLT, GAB, VxFEN, and VCS engine define the default VCS behavior to start these modules during system restart or stop these modules during system shutdown.

Note: The startup and shutdown of LLT, GAB, VxFEN, and VCS engine are inter-dependent. For a clean startup or shutdown of VCS, you must either enable or disable the startup and shutdown modes for all these modules.

In a single-node VCS cluster, you can disable the start and stop environment variables for LLT, GAB, and VxFEN.

[Table 3-2](#) lists the start and stop environment variables.

Table 3-2 Start and stop environment variables for VCS

Environment variable	Definition and default value
LLT_START	Startup mode for LLT. By default, LLT is enabled to start up after a system reboot. This environment variable is defined in the following file: <code>/etc/default/llt</code> Default: 1

Table 3-2 Start and stop environment variables for VCS

Environment variable	Definition and default value
LLT_STOP	<p>Shutdown mode for LLT. By default, LLT is enabled to shut down after a system shutdown.</p> <p>This environment variable is defined in the following file:</p> <p>/etc/default/llt</p> <p>Default: 1</p>
GAB_START	<p>Startup mode for GAB. By default, GAB is enabled to start up after a system reboot.</p> <p>This environment variable is defined in the following file:</p> <p>/etc/default/gab</p> <p>Default: 1</p>
GAB_STOP	<p>Shutdown mode for GAB. By default, GAB is enabled to shut down after a system shutdown.</p> <p>This environment variable is defined in the following file:</p> <p>/etc/default/gab</p> <p>Default: 1</p>
VXFEN_START	<p>Startup mode for VxFEN. By default, VxFEN is enabled to start up after a system reboot.</p> <p>This environment variable is defined in the following file:</p> <p>/etc/default/vxfen</p> <p>Default: 1</p>
VXFEN_STOP	<p>Shutdown mode for VxFEN. By default, VxFEN is enabled to shut down after a system shutdown.</p> <p>This environment variable is defined in the following file:</p> <p>/etc/default/vxfen</p> <p>Default: 1</p>
VCS_START	<p>Startup mode for VCS engine. By default, VCS engine is enabled to start up after a system reboot.</p> <p>This environment variable is defined in the following file:</p> <p>/etc/default/vcs</p> <p>Default: 1</p>

Table 3-2 Start and stop environment variables for VCS

Environment variable	Definition and default value
VCS_STOP	<p>Shutdown mode for VCS engine. By default, VCS engine is enabled to shut down after a system shutdown.</p> <p>This environment variable is defined in the following file:</p> <p>/etc/default/vcs</p> <p>Default: 1</p>



Administration-Putting VCS to work

- [Chapter 4, “About the VCS user privilege model” on page 71](#)
- [Chapter 5, “Administering the cluster from Cluster Manager \(Java console\)” on page 79](#)
- [Chapter 6, “Administering the cluster from the command line” on page 179](#)
- [Chapter 7, “Configuring applications and resources in VCS” on page 245](#)
- [Chapter 8, “Predicting VCS behavior using VCS Simulator” on page 265](#)

About the VCS user privilege model

- [About VCS user privileges and roles](#)
- [How administrators assign roles to users](#)
- [User privileges for OS user groups for clusters running in secure mode](#)
- [About VCS privileges for users with multiple roles](#)

About VCS user privileges and roles

Cluster operations are enabled or restricted depending on the privileges with which you log on. VCS has three privilege levels: Administrator, Operator, and Guest. VCS provides some predefined user roles; each role has specific privilege levels. For example, the role Guest has the fewest privileges, Cluster Administrator the most.

See “[VCS user privileges—administration matrices](#)” on page 623.

About VCS privilege levels

- VCS privilege levels include:
- Administrators— Can perform all operations, including configuration options on the cluster, service groups, systems, resources, and users.
 - Operators—Can perform specific operations on a cluster or a service group.
 - Guests—Can view specified objects.

About user roles in VCS

[Table 4-1](#) lists the predefined VCS user roles, with a summary of their associated privileges.

Table 4-1 User roles in VCS

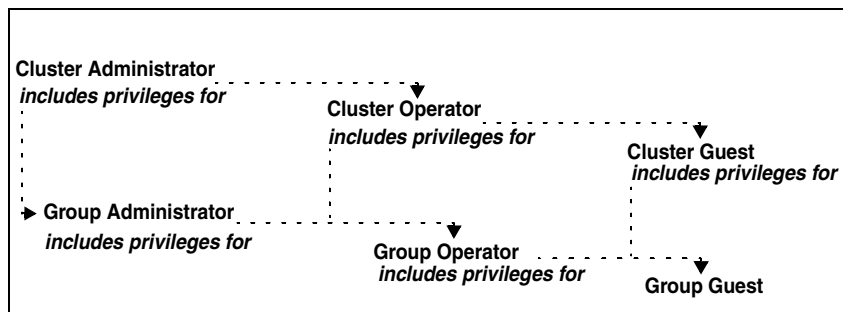
User Role	Privileges
Cluster Administrator	<p>Cluster Administrators are assigned full privileges, including making configuration read-write, creating and deleting groups, setting group dependencies, adding and deleting systems, and adding, modifying, and deleting users. All group and resource operations are allowed. Users with Cluster Administrator privileges can also change other users’ privileges and passwords.</p> <p>To stop a cluster, cluster administrators require administrative privileges on the local system.</p> <p>Note: Cluster Administrators can change their own and other users’ passwords only after changing the configuration to read/write mode.</p> <p>Cluster Administrators can create and delete resource types.</p>

Table 4-1 User roles in VCS

User Role	Privileges
Cluster Operator	<p>Cluster Operators can perform all cluster-, group-, and resource-level operations, including modifying the user's own password, bringing service groups online and taking them offline.</p> <p>Note: Cluster Operators can change their own passwords only if configuration is in read/write mode. Cluster Administrators can change the configuration to the read/write mode.</p> <p>Users with this role can be assigned Group Administrator privileges for specific service groups.</p>
Group Administrator	<p>Group Administrators can perform all service group operations on specific groups, such as bringing groups and resources online, taking them offline, and creating or deleting resources. Additionally, users can establish resource dependencies and freeze or unfreeze service groups. Note that Group Administrators cannot create or delete service groups.</p>
Group Operator	<p>Group Operators can bring service groups and resources online and take them offline. Users can also temporarily freeze or unfreeze service groups.</p>
Cluster Guest	<p>Cluster Guests have read-only access to the cluster, meaning they can view the configuration, but cannot change it. They can modify their own passwords only if the configuration is in read/write mode. They cannot add or update users. Additionally, users with this privilege can be assigned Group Administrator or Group Operator privileges for specific service groups.</p> <p>Note: By default, newly created users are assigned Cluster Guest permissions.</p>
Group Guest	<p>Group Guests have read-only access to the service group, meaning they can view the configuration, but cannot change it. The Group Guest role is available for clusters running in secure mode.</p>

About the hierarchy in VCS roles

The following illustration shows the roles and how they overlap with one another.



For example, Cluster Administrator includes privileges for Group Administrator, which includes privileges for Group Operator

User privileges for CLI commands

Users logged with administrative or root privileges are granted privileges that exceed those of Cluster Administrator, such as the ability to start and stop a cluster.

If you do not have root privileges, VCS prompts for your VCS user name and password when you execute `haxxx` commands.

You can use the `halogin` command to save the authentication information so that you do not have to enter your credentials every time you run a VCS command.

See [“Logging on to VCS”](#) on page 196.

User privileges in global clusters

VCS permits a cross-cluster online or offline operation only if the user initiating the operation has one of the following privileges:

- Group Administrator or Group Operator privileges for the group on the remote cluster
- Cluster Administrator or Cluster Operator privileges on the remote cluster

VCS permits a cross-cluster switch operation only if the user initiating the operation has the following privileges:

- Group Administrator or Group Operator privileges for the group on both clusters
- Cluster Administrator or Cluster Operator privileges on both clusters

User privileges for clusters running in secure mode

In secure mode, VCS assigns Guest privileges to all native users.

When assigning privileges for clusters running in secure mode, you must specify fully-qualified user names, in the format `username@domain`.

You cannot assign or change passwords for users using VCS when VCS is running in secure mode.

How administrators assign roles to users

To assign a role to a user, an administrator performs the following tasks:

- Add a user to the cluster, if the cluster is not running in secure mode.
- Assign a role to the user.
- Assign the user a set of objects appropriate for the role. For clusters running in secure mode, you can also add a role to an operating system user group. See [“User privileges for OS user groups for clusters running in secure mode”](#) on page 76.

For example, an administrator may assign a user the Group Administrator role for specific service groups. Now, the user has privileges to perform operations on the specific service groups.

You can manage users and their privileges from the command line or from the graphical user interface.

See [“Managing VCS users from the command line”](#) on page 207

See [“Administering user profiles”](#) on page 117.

User privileges for OS user groups for clusters running in secure mode

For clusters running in secure mode, you can assign privileges to native users individually or at an operating system (OS) user group level.

For example, you may decide that all users that are part of the OS Administrators group get administrative privileges to the cluster or to a specific service group. Assigning a VCS role to a user group assigns the same VCS privileges to all members of the user group, unless you specifically exclude individual users from those privileges.

When you add a user to an OS user group, the user inherits VCS privileges assigned to the user group.

Assigning VCS privileges to an OS user group involves adding the user group in one (or more) of the following attributes:

- AdministratorGroups—for a cluster or for a service group.
- OperatorGroups—for a cluster or for a service group.

For example, user Tom belongs to an OS user group: OSUserGroup1. You can assign VCS privileges to user Tom in the following ways:

To assign privileges	At an individual level, configure attribute	To the OS user group, configure attribute
Cluster Administrator	cluster (Administrators = {tom@domain})	cluster (AdministratorGroups = {OSUserGroup1@domain})
Cluster Operator	cluster (Operators = {tom@domain})	cluster (OperatorGroups = {OSUserGroup1@domain})
Cluster Guest	Cluster (Guests = {tom@domain})	
Group Administrator	group <i>group_name</i> (Administrators = {tom@domain})	group <i>group_name</i> (AdministratorGroups = {OSUserGroup1@domain})
Group Operator	group <i>group_name</i> (Operators = {tom@domain})	group <i>group_name</i> (OperatorGroups = {OSUserGroup1@domain})
Group Guest	Cluster (Guests = {tom@domain})	

About VCS privileges for users with multiple roles

Table 4-2 describes how VCS assigns privileges to users with multiple roles. The scenarios describe user Tom who is part of two OS user groups: OSUserGroup1 and OSUserGroup2.

Table 4-2 VCS privileges for users with multiple roles

Situation and rule	Roles assigned in the VCS configuration	Privileges that VCS grants Tom
Situation: Multiple roles at an individual level. Rule: VCS grants highest privileges (or a union of all the privileges) to the user.	Tom: Cluster Administrator Tom: Group Operator	Cluster Administrator.
Situation: Roles at an individual and OS user group level (secure clusters only). Rule: VCS gives precedence to the role granted at the individual level.	Tom: Group Operator OSUserGroup1: Cluster Administrator	Group Operator
Situation: Different roles for different OS user groups (secure clusters only). Rule: VCS grants the highest privilege (or a union of all privileges of all user groups) to the user.	OSUserGroup1: Cluster Administrators OSUserGroup2: Cluster Operators	Cluster Administrator
Situation: Roles at an individual and OS user group level (secure clusters only). Rule: VCS gives precedence to the role granted at the individual level. You can use this behavior to exclude specific users from inheriting VCS privileges assigned to their OS user groups.	OSUserGroup1: Cluster Administrators OSUserGroup2: Cluster Operators Tom: Group Operator	Group Operator

Administering the cluster from Cluster Manager (Java console)

- [About the Cluster Manager \(Java Console\)](#)
- [Getting started](#)
- [Reviewing components of the Java Console](#)
- [About Cluster Monitor](#)
- [About Cluster Explorer](#)
- [Accessing additional features of the Java Console](#)
- [Administering Cluster Monitor](#)
- [Administering user profiles](#)
- [Administering service groups](#)
- [Administering resources](#)
- [Administering systems](#)
- [Administering clusters](#)
- [Executing commands](#)
- [Editing attributes](#)
- [Querying the cluster configuration](#)
- [Setting up VCS event notification using the Notifier wizard](#)
- [Administering logs](#)
- [Administering VCS Simulator](#)

About the Cluster Manager (Java Console)

Cluster Manager (Java Console) offers complete administration capabilities for your cluster. Use the different views in the Java Console to monitor clusters and VCS objects, including service groups, systems, resources, and resource types. Many of the operations supported by the Java Console are also supported by the command line interface and Cluster Management Console.

The console enables or disables features depending on whether the features are supported in the cluster that the console is connected to. For example, the Cluster Shell icon is grayed out when you connect to recent versions of VCS. But the icon is enabled when you connect to earlier versions of a VCS cluster.

You can download the console from <http://go.symantec.com/vcsmc>.

Symantec also offers the Veritas Cluster Server (VCS) Management Console to manage clusters. Refer to the *Veritas Cluster Server Management Console Implementation Guide* for installation, upgrade, and configuration instructions.

To download the most current version of VCS Management Console, go to <http://go.symantec.com/vcsmc>.

Disability compliance

Cluster Manager (Java Console) for VCS provides disabled individuals access to and use of information and data that is comparable to the access and use provided to non-disabled individuals, including:

- Alternate keyboard sequences for specific operations.
See “[Accessibility and VCS](#)” on page 713.
- High-contrast display settings.
- Support of third-party accessibility tools. Note that Symantec has not tested screen readers for languages other than English.
- Text-only display of frequently viewed windows.

Getting started

- Make sure you have the current version of Cluster Manager (Java Console) installed. If you have a previous version installed, upgrade to the latest version. Cluster Manager (Java Console) is compatible with earlier versions of VCS.
- Cluster Manager (Java Console) is supported on:
 - AIX 5.2 and 5.3
 - Windows XP and Windows 2003

Note: Make sure you are using an operating system version that supports JRE 1.5.

- Verify the configuration has a user account. A user account is established during VCS installation that provides immediate access to Cluster Manager. If a user account does not exist, you must create one.
See [“Adding a user”](#) on page 117.
- On UNIX systems, you must set the display for Cluster Manager.
See [“Setting the display on UNIX systems”](#) on page 81.
- Start Cluster Manager.
See [“Starting Cluster Manager \(Java Console\)”](#) on page 83.
- Add a cluster panel.
See [“Configuring a new cluster panel”](#) on page 112.
- Log on to a cluster.
See [“Logging on to and off of a cluster”](#) on page 114.
- Make sure you have adequate privileges to perform cluster operations.
See [“About the VCS user privilege model”](#) on page 71.

Setting the display on UNIX systems

The UNIX version of the Cluster Manager (Java Console) requires an X-Windows desktop. Setting the display is not required on Windows workstations.

To set the display

- 1 Type the following command to grant the system permission to display on the desktop:

`xhost +`

- 2 Configure the shell environment variable `DISPLAY` on the system where Cluster Manager will be launched. For example, if using Korn shell, type the following command to display on the system `myws`:

```
export DISPLAY=myws:0
```

Using Java Console with secure shell

You can use Java Console with secure shell (SSH) using X11 forwarding, or Port forwarding. Make sure that SSH is correctly configured on the client and the host systems.

To use x11 forwarding

- 1 In the `ssh` configuration file, set `ForwardX11` to `yes`.
ForwardX11 yes
- 2 Log on to the remote system and start an X clock program that you can use to test the forward connection.

```
xclock &.
```

Do not set the `DISPLAY` variable on the client. X connections forwarded through a secure shell use a special local display setting.

To use port forwarding

In this mode the console connects to a specified port on the client system. This port is forwarded to port 14141 on the VCS server node.

- 1 In the `ssh` configuration file, set `GatewayPorts` to `yes`.
GatewayPorts yes
- 2 From the client system, forward a port (*client_port*) to port 14141 on the VCS server.

```
$ssh -L client_port:server_host:14141 server_host
```

You may not be able set `GatewayPorts` in the configuration file if you use `openSSH`. In this case use the `-g` option in the command.

```
$ssh -g -L client_port:server_host:14141 server_host
```

- 3 Open another window on the client system and start the Java Console.
\$/opt/VRTSvcs/bin/hagui
- 4 Add a cluster panel in the Cluster Monitor. When prompted, enter the name of client system as the host and the *client_port* as the port. Do not enter `localhost`.

Starting Cluster Manager (Java Console)

You can run the Java Console on Windows or UNIX systems.

To start the Java Console on Windows systems

From the Start menu, click **Start>All Programs>Symantec>Veritas Cluster Server>Veritas Cluster Manager - Java Console**.

To start the Java Console on UNIX systems

After establishing a user account and setting the display, type the following command to start Cluster Manager:

```
/opt/VRTSvcS/bin/hagui
```

The command `hagui` will work across firewalls only if the following outgoing server ports are open:

- VCS Engine: 14141
- Command server: 14150
- Secure cluster : 2821

Reviewing components of the Java Console

Cluster Manager (Java Console) offers two windows, Cluster Monitor and Cluster Explorer, from which most tasks are performed. Use Cluster Manager to manage, configure, and administer the cluster while VCS is running (online). The Java Console also enables you to use VCS Simulator. Use this tool to simulate operations and generate new configuration files (main.cf and types.cf) while VCS is offline. VCS Simulator enables you to design configurations that imitate real-life scenarios without test clusters or changes to existing configurations.

See “[Administering VCS Simulator](#)” on page 177.

Icons in the Java Console

The Java Console uses the following icons to communicate information about cluster objects and their states.

See “[Cluster and system states](#)” on page 635.

Table 5-1 Icons in Cluster Manager (Java Console)








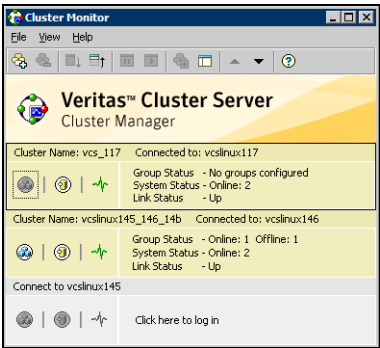
Icon	Description
	Cluster
	System
	Service Group
	Resource Type
	Resource
	OFFLINE
	Faulted (in UP BUT NOT IN CLUSTER MEMBERSHIP state)

Table 5-1 Icons in Cluster Manager (Java Console)

Icon	Description
	Faulted (in EXITED state)
	PARTIAL
	Link Heartbeats (in UP and DOWN states)
	UP AND IN JEOPARDY
	FROZEN
	AUTODISABLED
	UNKNOWN
	ADMIN_WAIT
	Global Service Group (requires the VCS Global Cluster Option)
	Remote Cluster in RUNNING state (requires the VCS Global Cluster Option)
	Remote Cluster in EXITING, EXITED, INIT, INQUIRY, LOST_CONN, LOST_HB, TRANSITIONING, or UNKNOWN state.

About Cluster Monitor

After starting Cluster Manager, the first window that appears is Cluster Monitor. This window includes one or more panels displaying general information about actual or simulated clusters. Use Cluster Monitor to log on to and off of a cluster, view summary information on various VCS objects, customize the display, use VCS Simulator, and exit Cluster Manager.



Cluster monitor toolbar

The Cluster Monitor toolbar contains the following buttons.



From left to right:



New Cluster. Adds a new cluster panel to Cluster Monitor.



Delete Cluster. Removes a cluster panel from Cluster Monitor.



Expand. Expands the Cluster Monitor view.



Collapse. Collapses the Cluster Monitor view.



Stop. Pauses cluster panel scrolling.



Start. Resumes scrolling.



Login. Log on to the cluster shown in the cluster panel.



Show Explorer. Launches an additional window of Cluster Explorer after logging on to that cluster.



Move Cluster Panel Up. Moves the selected cluster panel up.



Move Cluster Panel Down. Moves the selected cluster panel down.



Help. Access online help.

Cluster monitor panels

To administer a cluster, add a cluster panel or reconfigure an existing cluster panel in Cluster Monitor. Each panel summarizes the status of the connection and components of a cluster.

Monitoring the cluster connection with Cluster Monitor

The right pane of a panel in Cluster Monitor displays the status of the connection to a cluster. An inactive panel will appear grey until the user logs on and connects to the cluster. To alter the connection to a cluster, right-click a panel to access a menu.

- The menu on an active panel enables you to log off a cluster.
- The menu on an inactive panel enables you to log on to a cluster, configure the cluster, and delete the cluster from Cluster Monitor.

Menus are enabled when the Cluster Monitor display appears in the default expanded view. If you activate a menu on a collapsed scrolling view of Cluster Monitor, the scrolling stops while accessing the menu.

If the system to which the console is connected goes down, a message notifies you that the connection to the cluster is lost. Cluster Monitor tries to connect to another system in the cluster according to the number of Failover retries set in the Connectivity Configuration dialog box. The panels flash until Cluster Monitor is successfully connected to a different system. If the failover is unsuccessful, a message notifies you of the failure and the panels turn grey.

Monitoring VCS objects with Cluster Monitor

Cluster Monitor summarizes the state of various objects in a cluster and provides access to in-depth information about these objects in Cluster Explorer. The right pane of a Cluster Monitor panel displays the connection status (online, offline, up, or down) of service groups, systems, and heartbeats. The left pane of a Cluster Monitor panel displays three icons representing service groups, systems, and heartbeats. The colors of the icons indicate the state of the cluster; for example:

- A flashing red slash indicates Cluster Manager failed to connect to the cluster and will attempt to connect to another system in the cluster.
- A flashing yellow slash indicates Cluster Manager is experiencing problems with the connection to the cluster.

Pointing to an icon accesses the icon's ScreenTip, which provides additional information on the specific VCS object.

To review detailed information about VCS objects in Cluster Explorer, Logs, and Command Center, right-click a panel to access a menu. Menus are enabled when the Cluster Monitor display appears in the default expanded view. If you activate a menu on a collapsed scrolling view of Cluster Monitor, the scrolling stops while accessing the menu.

Expanding and collapsing the Cluster Monitor display

Cluster Monitor supports two views: expanded (default) and collapsed. The expanded view shows all cluster panels. The collapsed view shows one cluster panel at a time as the panels scroll upward.

Operations enabled for the expanded view of cluster panels, such as viewing menus, are also enabled on the collapsed view after the panels stop scrolling.

To collapse the Cluster Monitor view

On the **View** menu, click **Collapse**.

or

Click **Collapse** on the Cluster Monitor toolbar.

To expand the Cluster Monitor view

On the **View** menu, click **Expand**.

or

Click **Expand** on the Cluster Monitor toolbar.

To pause a scrolling cluster panel

Click the cluster panel.

or

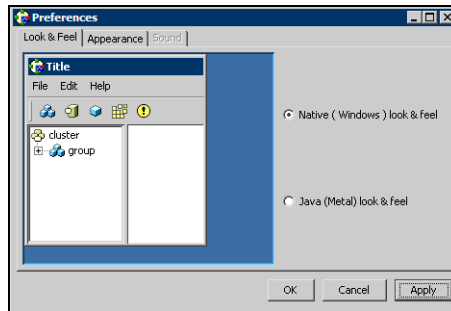
Click **Stop** on the Cluster Monitor toolbar.

Customizing the Cluster Manager display

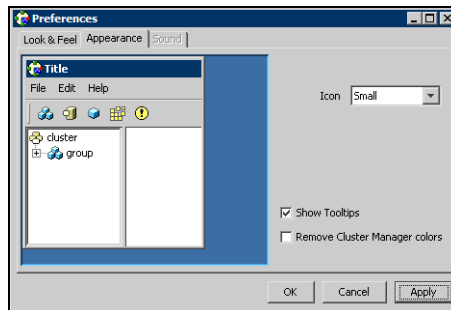
Customize the Cluster Manager to display objects according to your preference.

To customize the Cluster Manager display

- 1 From Cluster Monitor, click **Preferences** on the **File** menu. If you are using a Windows system, proceed to step 2. Otherwise, proceed to step 3.
- 2 In the **Look & Feel** tab (for Windows systems):

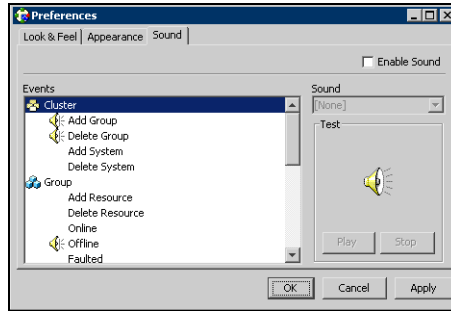


- Click **Native (Windows or Motif) look & feel** or **Java (Metal) look & feel**.
 - Click **Apply**.
- 3 In the **Appearance** tab:



- Click the color (applies to Java (Metal) look & feel).
- Click an icon size.
- Select the **Show Tooltips** check box to enable ToolTips.
- Select the **Remove Cluster Manager colors** check box to alter the standard color scheme.
- Click **Apply**.

4 In the **Sound** tab:

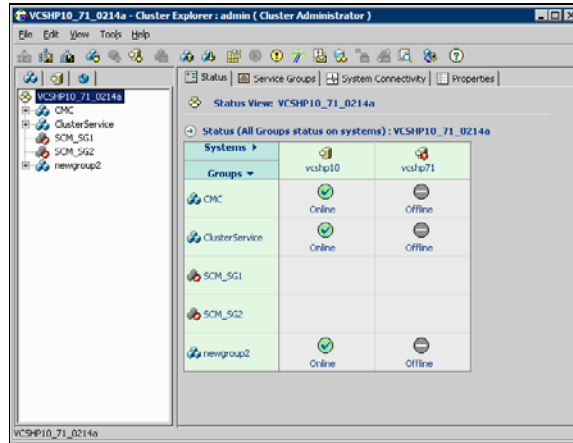


Note: This tab requires a properly configured sound card.

- Select the **Enable Sound** check box to associate sound with specific events.
 - Click an event from the **Events** configuration tree.
 - Click a sound from the **Sounds** list box.
 - To test the selected sound, click **Play**.
 - Click **Apply**.
 - Repeat these steps to enable sound for other events.
- 5 After you have made your final selection, click **OK**.

About Cluster Explorer

Cluster Explorer is the main window for cluster administration. From this window, you can view the status of VCS objects and perform various operations.



The display is divided into three panes. The top pane includes a toolbar that enables you to quickly perform frequently used operations. The left pane contains a configuration tree with three tabs: Service Groups, Systems, and Resource Types. The right pane contains a panel that displays various views relevant to the object selected in the configuration tree.

To access Cluster Explorer

- 1 Log on to the cluster.
- 2 Click anywhere in the active Cluster Monitor panel.
or
Right-click the selected Cluster Monitor panel and click Explorer View from the menu.

Cluster Explorer toolbar

The Cluster Explorer toolbar contains 18 buttons. Available operations are described below. Note: Some buttons may be disabled depending on the type of cluster (local or global) and the privileges with which you logged on to the cluster.



From left to right:



Open Configuration. Modifies a read-only configuration to a read-write file. This enables you to modify the configuration.



Save Configuration. Writes the configuration to disk.



Save and Close Configuration. Writes the configuration to disk as a read-only file.



Add Service Group. Displays the Add Service Group dialog box.



Add Resource. Displays the Add Resource dialog box.



Add System. Displays the Add System dialog box.



Manage systems for a Service Group. Displays the System Manager dialog box.



Online Service Group. Displays the Online Service Group dialog box.



Offline Service Group. Displays the Offline Service Group dialog box.



Show Command Center. Enables you to perform many of the same VCS operations available from the command line.



Show Shell Command Window. Enables you to launch a non-interactive shell command on cluster systems, and to view the results on a per-system basis.



Show the Logs. Displays alerts and messages received from the VCS engine, VCS agents, and commands issued from the console.



Launch Configuration Wizard. Enables you to create VCS service groups.



Launch Notifier Resource Configuration Wizard. Enables you to set up VCS event notification.



Remote Group Resource Configuration Wizard. Enables you to configure resources to monitor a service group in a remote cluster.



Add/Delete Remote Clusters. Enables you to add and remove global clusters.



Configure Global Groups. Enables you to convert a local service group to a global group, and vice versa.



Query. Enables you to search the cluster configuration according to filter criteria.



Virtual Fire Drill. Checks whether a resource can fail over to another node in the cluster. Requires agents that support running virtual fire drills.



Show Cluster Explorer Help. Enables you to access online help.

Cluster Explorer configuration tree

The Cluster Explorer configuration tree is a tabbed display of VCS objects.

- The **Service Groups** tab lists the service groups in the cluster. Expand each service group to view the group's resource types and resources.
- The **Systems** tab lists the systems in the cluster.
- The **Types** tab lists the resource types in the cluster

Cluster Explorer view panel

The right pane of the Cluster Explorer includes a view panel that provides detailed information about the object selected in the configuration tree. The information is presented in tabular or graphical format. Use the tabs in the view panel to access a particular view. The console enables you to “tear off” each view to appear in a separate window.

- Click any object in the configuration tree to access the Status View and Properties View.
- Click a cluster in the configuration tree to access the Service Group view, System Connectivity view, and Remote Cluster Status View (for global clusters only).
- Click a service group in the configuration tree to access the Resource view.

To create a tear-off view

On the **View** menu, click **Tear Off**, and click the appropriate view from the menu.

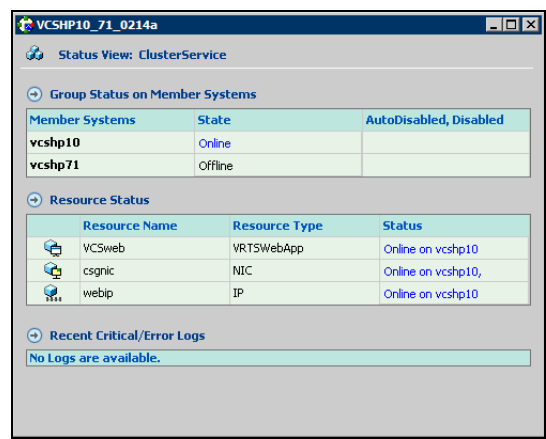
or

Right-click the object in the configuration tree, click **View**, and click the appropriate view from the menu.

Status view

The Status View summarizes the state of the object selected in the configuration tree. Use this view to monitor the overall status of a cluster, system, service group, resource type, and resource.

For example, if a service group is selected in the configuration tree, the Status View displays the state of the service group and its resources on member systems. It also displays the last five critical or error logs. Point to an icon in the status table to open a ScreenTip about the relevant VCS object.



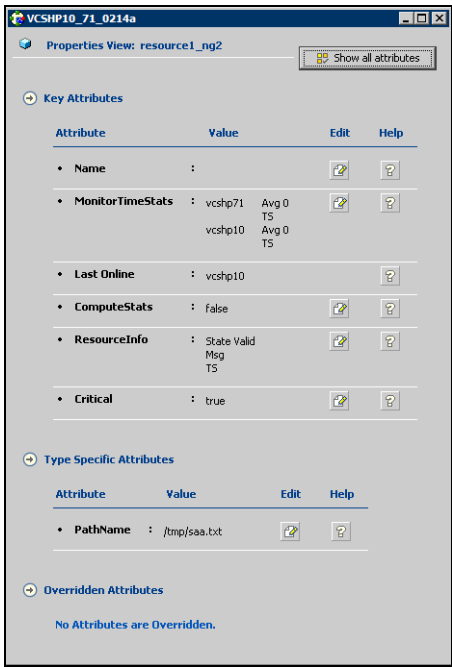
For global clusters, this view displays the state of the remote clusters. For global groups, this view shows the status of the groups on both local and remote clusters.

To access the Status view

- 1 From Cluster Explorer, click an object in the configuration tree.
- 2 In the view panel, click the **Status** tab.

Properties view

The Properties View displays the attributes of VCS objects. These attributes describe the scope and parameters of a cluster and its components.



To view information on an attribute, click the attribute name or the icon in the **Help** column of the table.

See “[VCS attributes](#)” on page 643”

By default, this view displays key attributes of the object selected in the configuration tree. The Properties View for a resource displays key attributes of the resource and attributes specific to the resource types. It also displays attributes whose values have been overridden.

See “[Overriding resource type static attributes](#)” on page 152.

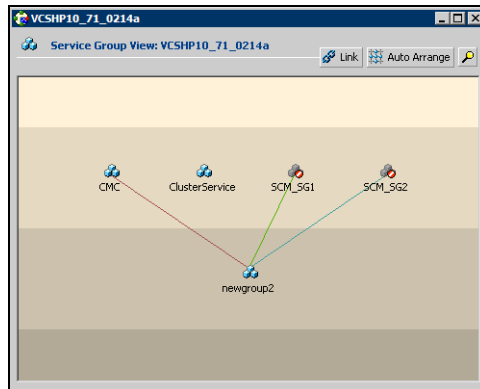
To view all attributes associated with the selected VCS object, click **Show all attributes**.

To access the properties view

- 1 From Cluster Explorer, click a VCS object in the configuration tree.
- 2 In the view panel, click the **Properties** tab.

Service Group view

The Service Group view displays the service groups and their dependencies in a cluster. Use the graph and ScreenTips in this view to monitor, create, and disconnect dependencies. To view the ScreenTips, point to a group icon for information on the type and state of the group on the cluster systems, and the type of dependency between the service groups.



The line between two service groups represents a dependency, or parent-child relationship. In VCS, parent service groups depend on child service groups. A service group can function as a parent and a child.

See “[About service group dependencies](#)” on page 410.

The color of the link between service groups indicates different types of dependencies.

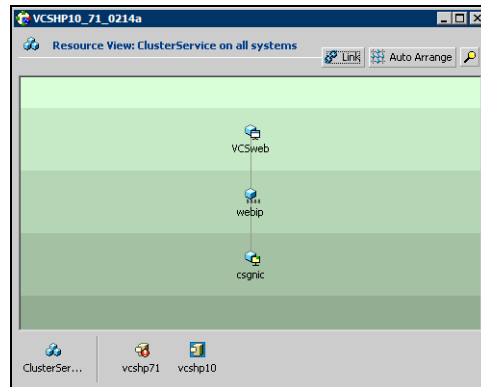
- A blue link indicates a soft dependency.
- A red link indicates a firm dependency.
- A green link indicates a hard dependency typically used with VVR in disaster recovery configurations.

To access the Service Group view

- 1 From Cluster Explorer, click a cluster in the configuration tree.
- 2 In the view panel, click the **Service Groups** tab.

Resource view

The Resource view displays the resources in a service group. Use the graph and ScreenTips in this view to monitor the dependencies between resources and the status of the service group on all or individual systems in a cluster.



In the graph, the line between two resources represents a dependency, or parent-child relationship. Resource dependencies specify the order in which resources are brought online and taken offline. During a failover process, the resources closest to the top of the graph must be taken offline before the resources linked to them are taken offline. Similarly, the resources that appear closest to the bottom of the graph must be brought online before the resources linked to them can come online.

- A resource that depends on other resources is a parent resource. The graph links a parent resource icon to a child resource icon below it. Root resources (resources without parents) are displayed in the top row.
- A resource on which the other resources depend is a child resource. The graph links a child resource icon to a parent resource icon above it.
- A resource can function as a parent and a child.

Point to a resource icon to display ScreenTips about the type, state, and key attributes of the resource. The state of the resource reflects the state on a specified system (local).

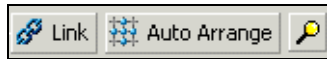
In the bottom pane of the Resource view, point to the system and service group icons to display ScreenTips about the service group status on all or individual systems in a cluster. Click a system icon to view the resource graph of the service group on the system. Click the service group icon to view the resource graph on all systems in the cluster.

To access the Resource view

- 1 From Cluster Explorer, click the service groups tab in the configuration tree.
- 2 Click a service group in the configuration tree.
- 3 In the view panel, click the **Resources** tab.

Moving and linking icons in Service Group and Resource views

The Link and Auto Arrange buttons are available in the top right corner of the Service Group or Resource view:



Click **Link** to set or disable the link mode for the Service Group and Resource views.

Note: There are alternative ways to set up dependency links without using the Link button.

The link mode enables you to create a dependency link by clicking on the parent icon, dragging the yellow line to the icon that will serve as the child, and then clicking the child icon. Use the Esc key to delete the yellow dependency line connecting the parent and child during the process of linking the two icons.

If the Link mode is *not* activated, click and drag an icon along a horizontal plane to move the icon. Click **Auto Arrange** to reset the appearance of the graph. The view resets the arrangement of icons after the addition or deletion of a resource, service group, or dependency link. Changes in the Resource and Service Group views will be maintained after the user logs off and logs on to the Java Console at a later time.

Zooming in on Service Group and Resource views

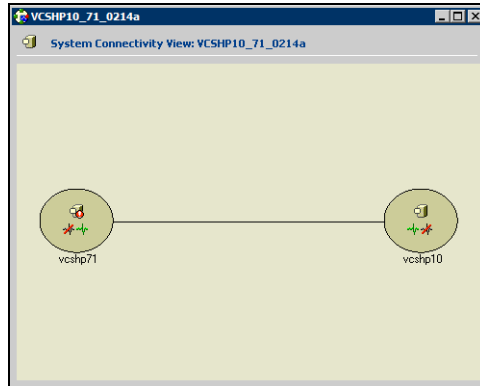
The Resource view and Service Group view include a navigator tool to zoom in or out of their graphs. Click the magnifying glass icon in the top right corner to open the zoom panel.



- To move the view to the left or right, click a distance (in pixels) from the drop-down list box between the hand icons. Click the <- or -> hand icon to move the view in the desired direction.
- To shrink or enlarge the view, click a size factor from the drop-down list box between the magnifying glass icons. Click the - or + magnifying glass icon to modify the size of the view.
- To view a segment of the graph, point to the box to the right of the + magnifying glass icon. Use the red outline in this box to encompass the appropriate segment of the graph. Click the newly outlined area to view the segment.
- To return to the original view, click the magnifying glass icon labeled 1.

System Connectivity view

The System Connectivity view displays the status of system connections in a cluster. Use this view to monitor the system links and disk group heartbeats.



VCS monitors systems and their services over a private network. The systems communicate via heartbeats over an additional private network, which enables them to recognize which systems are active members of the cluster, which are joining or leaving the cluster, and which have failed.

VCS protects against network failure by requiring that all systems be connected by two or more communication channels. When a system is down to a single heartbeat connection, VCS can no longer discriminate between the loss of a system and the loss of a network connection. This situation is referred to as jeopardy.

Point to a system icon to display a ScreenTip on the links and disk group heartbeats. If a system in the cluster is experiencing a problem connecting to other systems, the system icon changes its appearance to indicate the link is down. In this situation, a jeopardy warning may appear in the ScreenTip for this system.

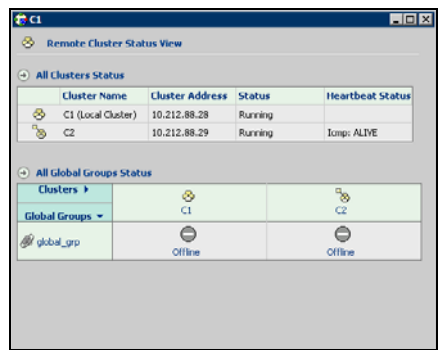
To access the System Connectivity view

- 1 From Cluster Explorer, click a cluster in the configuration tree.
- 2 In the view panel, click the **System Connectivity** tab.

Remote Cluster Status view

Note: This view requires the VCS Global Cluster Option.

The Remote Cluster Status View provides an overview of the clusters and global groups in a global cluster environment. Use this view to view the name, address, and status of a cluster, and the type (Icmp or IcmpS) and state of a heartbeat.



This view enables you to declare a remote cluster fault as a disaster, disconnect, or outage. Point to a table cell to view information about the VCS object.

To access the Remote Cluster Status view

- 1 From Cluster Explorer, click a cluster in the configuration tree.
- 2 In the view panel, click the **Remote Cluster Status** tab.

Accessing additional features of the Java Console

Use Cluster Manager to access the Template View, System Manager, User Manager, Command Center, Configuration Wizard, Notifier Resource Configuration Wizard, Remote Group Resource Configuration Wizard, Query Module, and Logs.

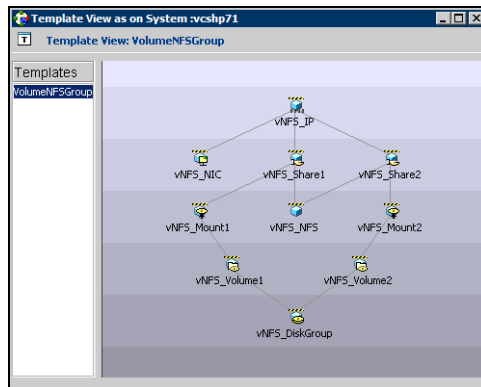
You can also use the Cluster Manager to run virtual fire drills (or HA fire drills) to check for any configurational discrepancies that might prevent a service group from coming online on a specific node.

Template view

The Template View displays the service group templates available in VCS. Templates are predefined service groups that define the resources, resource attributes, and dependencies within the service group. Use this view to add service groups to the cluster configuration, and copy the resources within a service group template to existing service groups.

In this window, the left pane displays the templates available on the system to which Cluster Manager is connected. The right pane displays the selected template's resource dependency graph.

Template files conform to the VCS configuration language and contain the extension .tf. These files reside in the VCS configuration directory.

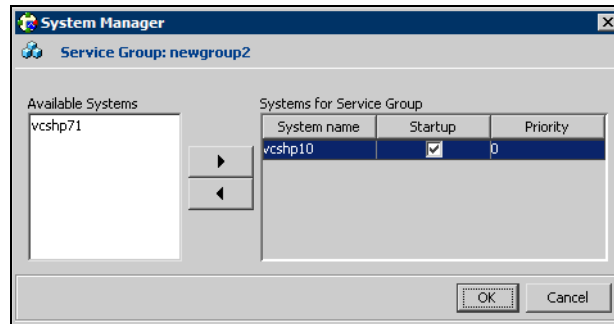


To access the template view

From Cluster Explorer, click **Templates** on the **Tools** menu.

System Manager

Use System Manager to add and remove systems in a service group's system list. A priority number (starting with 0) is assigned to indicate the order of systems on which the service group will start in case of a failover. If necessary, double-click the entry in the **Priority** column to enter a new value. Select the **Startup** check box to add the systems to the service groups AutoStartList attribute. This enables the service group to automatically come online on a system every time HAD is started.



To access system Manager

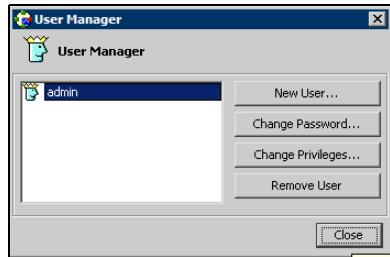
From Cluster Explorer, click the service group in the configuration tree, and click **System Manager** on the **Tools** menu.

or

In the **Service Groups** tab of the Cluster Explorer configuration tree, click a service group, and click **Manage systems for a Service Group** on the toolbar.

User Manager

User Manager enables you to add and delete user profiles and to change user privileges. If VCS is not running in secure mode, User Manager enables you to change user passwords. You must be logged in as Cluster Administrator to access User Manager.



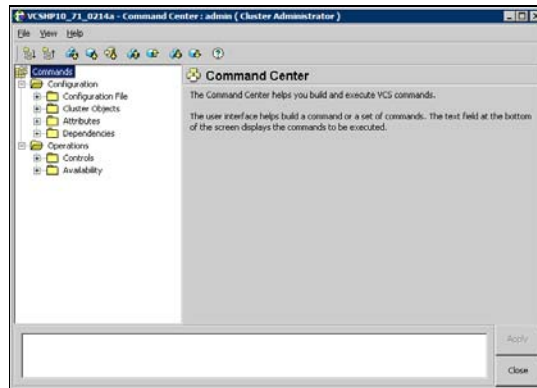
To access user Manager

From Cluster Explorer, click **User Manager** on the **File** menu.

Command Center

Command Center enables you to build and execute VCS commands; most commands that are executed from the command line can also be executed through this window. The left pane of the window displays a **Commands** tree of all VCS operations. The right pane displays a view panel that describes the selected command. The bottom pane displays the commands being executed.

The commands tree is organized into **Configuration** and **Operations** folders. Click the icon to the left of the **Configuration** or **Operations** folder to view its subfolders and command information in the right pane. Point to an entry in the commands tree to display information about the selected command.



To access Command Center

From Cluster Explorer, click **Command Center** on the **Tools** menu.

or

On the Cluster Explorer toolbar, click **Show Command Center**.

Configuration wizard

Use Configuration Wizard to create and assign service groups to systems in a cluster.

See “[Creating service groups with the configuration wizard](#)” on page 140.

To access Configuration Wizard

From Cluster Explorer, click **Configuration Wizard** on the **Tools** menu.

or

On the Cluster Explorer toolbar, click **Launch Configuration Wizard**.

Notifier Resource Configuration wizard

VCS provides a method for notifying an administrator of important events such as a resource or system fault. VCS includes a “notifier” component, which consists of the notifier daemon and the `hanotify` utility. This wizard enables you to configure the notifier component as a resource of type `NotifierMngr` as part of the `ClusterService` group.

See “[Setting up VCS event notification using the Notifier wizard](#)” on page 170.

To access Notifier Resource Configuration Wizard

From Cluster Explorer, click **Notifier Wizard** on the **Tools** menu.

or

On the Cluster Explorer toolbar, click **Launch Notifier Resource Configuration Wizard**.

Remote Group Resource Configuration Wizard

A `RemoteGroup` resource enables you to manage or monitor remote service groups.

See “[Adding a RemoteGroup resource from the Java Console](#)” on page 146.

To access Remote Group Resource Configuration Wizard

From Cluster Explorer, click **Remote Group Resource Wizard...** on the **Tools** menu.

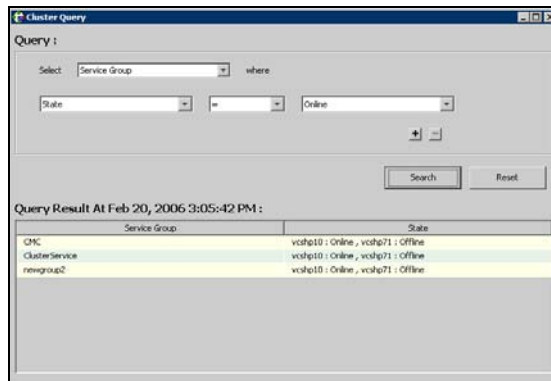
or

On the Cluster Explorer toolbar, click **Configure Remote Group Resource Wizard**.

Cluster query

Use Cluster Query to run SQL-like queries from Cluster Explorer. VCS objects that can be queried include service groups, systems, resources, and resource types. Some queries can be customized, including searching for the system's online group count and specific resource attributes.

See [“Querying the cluster configuration”](#) on page 169.



To access the Query dialog box

From Cluster Explorer, click **Query** on the **Tools** menu.

or

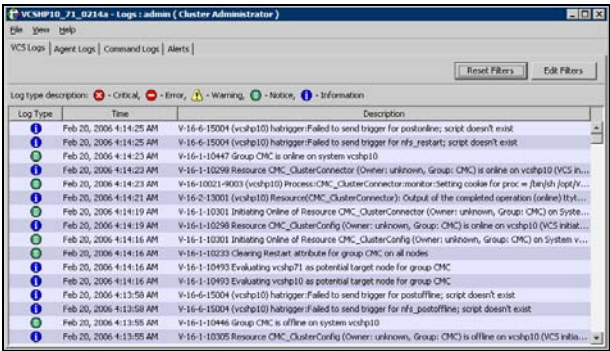
In the Cluster Explorer toolbar, click **Query**.

Logs

The Logs dialog box displays the log messages generated by the VCS engine, VCS agents, and commands issued from Cluster Manager to the cluster. Use this dialog box to monitor and take actions on alerts on faulted global clusters and failed service group failover attempts.

Note: To ensure the time stamps for engine log messages are accurate, make sure to set the time zone of the system running the Java Console to the same time zone as the system running the VCS engine.

- Click the **VCS Logs** tab to view the log type, time, and details of an event. Each message presents an icon in the first column of the table to indicate the message type. Use this window to customize the display of messages by setting filter criteria.



- Click the **Agent Logs** tab to display logs according to system, resource type, and resource filter criteria. Use this tab to view the log type, time, and details of an agent event.
- Click the **Command Logs** tab to view the status (success or failure), time, command ID, and details of a command. The Command Log only displays commands issued in the current session.
- Click the **Alerts** tab to view situations that may require administrative action. Alerts are generated when a local group cannot fail over to any system in the local cluster, a global group cannot fail over, or a cluster fault takes place. A current alert will also appear as a pop-up window when you log on to a cluster through the console.

To access the Logs dialog box

From Cluster Explorer, click **Logs** on the **View** menu.

or

On the Cluster Explorer toolbar, click **Show the Logs**.

Server and user credentials

If VCS is running in secure mode, you can view server and user credentials used to connect to the cluster from Cluster Explorer.

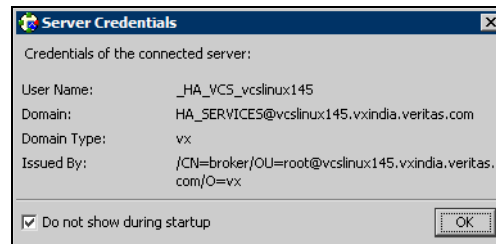
To view user credentials

From Cluster Explorer, click **User Credentials** on the **View** menu.



To view server credentials

From Cluster Explorer, click **Server Credentials** on the **View** menu.



Administering Cluster Monitor

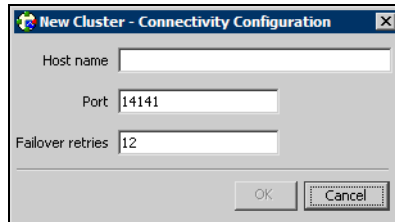
Use the Java Console to administer a cluster or simulated cluster by adding or reconfiguring a cluster panel in Cluster Monitor. To activate the connection of the procedures, log on to the cluster after completing the final step.

Configuring a new cluster panel

You must add a cluster panel for each cluster that you wish to connect to using the Java GUI.

To configure a new cluster panel

- 1 From Cluster Monitor, click **New Cluster** on the **File** menu. For simulated clusters, click **New Simulator** on the **File** menu.
or
Click **New Cluster** on the Cluster Monitor toolbar.
- 2 Enter the details to connect to the cluster:

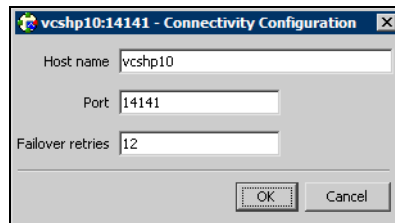


- Enter the host name or IP address of a system in the cluster.
- If necessary, change the default port number of 14141; VCS Simulator uses a default port number of 14153. Note that you must use a different port to connect to each Simulator instance, even if these instances are running on the same system.
- Enter the number of failover retries. VCS sets the default failover retries number to 12.
- For simulated clusters, click the platform for the configuration.
- Click **OK**. An inactive panel appears in Cluster Monitor.

Modifying a cluster panel configuration

Modify a cluster panel to point to another cluster, to change the port number, or the number of failover retries.

- 1 If Cluster Monitor is in the default expanded state, proceed to step 2. If Cluster Monitor is in the collapsed state:
On the **View** menu, click **Expand**.
or
On the **View** menu, click **Stop** when an active panel appears as the view panel.
- 2 Right-click the cluster panel. If the panel is inactive, proceed to step 4.
- 3 On the menu, click **Logout**. The cluster panel becomes inactive.
- 4 Right-click the inactive panel, and click **Configure...**
- 5 Edit the details to connect to the cluster:



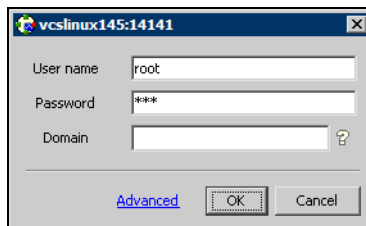
- Enter the host name or IP address of any system in the cluster.
- Enter the port number and the number of failover retries. VCS sets the default port number to 14141 and failover retries number to 12; VCS Simulator uses a default port number of 14153.
- For simulated panels, click the platform for the configuration.
- Click **OK**.

Logging on to and off of a cluster

After you add or configure a cluster panel in Cluster Monitor, log on to a cluster to access Cluster Explorer. Use Cluster Monitor to log off a cluster when you have completed administering the cluster.

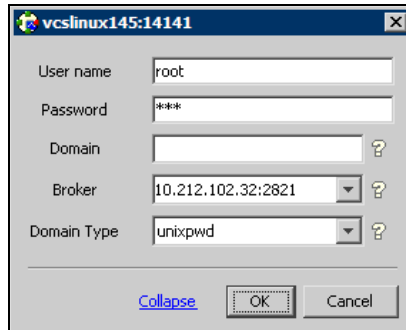
Logging on to a cluster

- 1 If Cluster Monitor is in the default expanded state, proceed to step 2. If Cluster Monitor is in the collapsed state:
On the **View** menu, click **Expand**.
or
On the **View** menu, click **Stop** when an active panel appears as the view panel.
- 2 Click the panel that represents the cluster you want to log on to.
or
If the appropriate panel is highlighted, click **Login** on the **File** menu.
- 3 Enter the information for the user:
If the cluster is not running in secure mode:
 - Enter the VCS user name and password.
 - Click **OK**.**If the cluster is running in secure mode:**



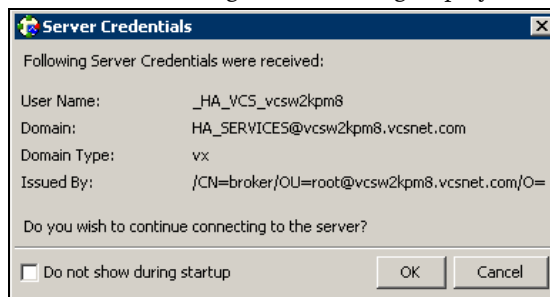
- Enter the credentials of a native user.
You can use nis or nis+ accounts or accounts set up on the local system. If you do not enter the name of the domain, VCS assumes the domain is the local system.
If the user does not have root privileges on the system, VCS assigns guest privileges to the user. To override these privileges, add the domain user to the VCS administrators' list.
See "[Administering user profiles](#)" on page 117.

- The Java Console connects to the cluster using the authentication broker and the domain type provided by the engine. To change the authentication broker or the domain type, click **Advanced**.
See “[About security services](#)” on page 33.



Select a new broker and domain type, as required.

- Click **OK**.
- The Server Credentials dialog box displays the credentials of the cluster service to which the console is connected.
To disable this dialog box from being displayed every time you connect



to the cluster, select the **Do not show during startup** check box

- Click **OK** to connect to the cluster.
The animated display shows various objects, such as service groups and resources, being transferred from the server to the console.
Cluster Explorer is launched automatically upon initial logon, and the icons in the cluster panel change color to indicate an active panel.

Logging off of a cluster

- 1 If Cluster Monitor is in the default expanded state, proceed to step 2. If Cluster Monitor is in the collapsed state:

On the **View** menu, click **Expand**.

or

On the **View** menu, click **Stop** when an active panel appears as the view panel.

- 2 Right-click the active panel, and click **Logout**.

or

If the appropriate panel is highlighted, click **Logout** on the **File** menu.

Cluster Explorer closes and the Cluster Monitor panel becomes inactive.

You may be prompted to save the configuration if any commands were executed on the cluster.

To log off from Cluster Explorer

Click **Log Out** on the **File** menu.

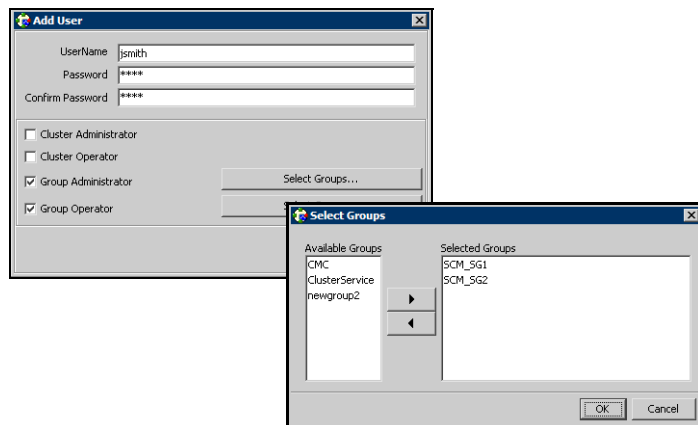
Administering user profiles

The Java Console enables a user with Cluster Administrator privileges to add, modify, and delete user profiles. The icon next to each user name in the User Manager dialog box indicates privileges for each user. Administrator and Operator privileges are separated into the cluster and group levels.

See “[About VCS user privileges and roles](#)” on page 72.

Adding a user

- 1 From Cluster Explorer, click **User Manager** on the **File** menu.
- 2 In the User Manager dialog box, click **New User**.
- 3 In the Add User dialog box:



- Enter the name of the user.
 - If the cluster is not running in secure mode, enter a password for the user and confirm it.
 - Select the appropriate check boxes to grant privileges to the user. To grant Group Administrator or Group Operator privileges, proceed to step the next step. Otherwise, proceed to the last step.
 - Click **Select Groups...**
 - Click the groups for which you want to grant privileges to the user and click the right arrow to move the groups to the **Selected Groups** box.
 - Click **OK** to exit the Select Group dialog box, then click **OK** again to exit the Add User dialog box.
- 4 Click **Close**.

Deleting a user

- 1 From Cluster Explorer, click **User Manager** on the **File** menu.
- 2 In the User Manager dialog box, click the user name.
- 3 Click **Remove User**.
- 4 Click **Yes**.
- 5 Click **Close**.

Changing a user password

A user with Administrator, Operator, or Guest privileges can change his or her own password. You must be logged on as Cluster Administrator to access User Manager. Before changing the password, make sure the configuration is in the read-write mode. Cluster administrators can change the configuration to the read-write mode.

Note: This module is not available if the cluster is running in secure mode.

To change a password as an administrator

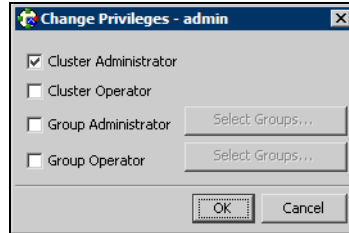
- 1 From Cluster Explorer, click **User Manager** on the **File** menu.
- 2 Click the user name.
- 3 Click **Change Password**.
- 4 In the Change Password dialog box:
 - Enter the new password.
 - Re-enter the password in the **Confirm Password** field.
 - Click **OK**.
- 5 Click **Close**.

To change a password as an operator or guest

- 1 From Cluster Explorer, click **Change Password** on the **File** menu.
- 2 In the Change Password dialog box:
 - Enter the new password.
 - Reenter the password in the **Confirm Password** field.
 - Click **OK**.
- 3 Click **Close**.

Changing a user privilege

- 1 From Cluster Explorer, click **User Manager** on the **File** menu.
- 2 Click the user name.
- 3 Click **Change Privileges** and enter the details for user privileges:



- Select the appropriate check boxes to grant privileges to the user. To grant Group Administrator or Group Operator privileges, proceed to the next step. Otherwise, proceed to the last step.
- Click **Select Groups**.
- Click the groups for which you want to grant privileges to the user, then click the right arrow to move the groups to the **Selected Groups** box.
- Click **OK** in the Change Privileges dialog box, then click **Close** in the User Manager dialog box.

Assigning privileges for OS user groups for clusters running in secure mode

For clusters running in secure mode, you can assign privileges to native users at an operating system (OS) user group level. Assigning VCS privileges to an OS user group involves adding the user group in one (or more) of the following attributes:

- **AdministratorGroups**—for a cluster or for a service group.
- **OperatorGroups**—for a cluster or for a service group.

See [“User privileges for OS user groups for clusters running in secure mode”](#) on page 76.

To assign privileges to an OS user group

- 1 From Cluster Explorer configuration tree, select the cluster to assign privileges for the cluster or a service group to assign privileges for specific service groups.
- 2 From the view panel, click the **Properties** tab.
- 3 From the list of key attributes, click the edit icon against **AdministratorGroups** or **OperatorGroups**.
- 4 In the Edit Attribute dialog box:
 - Use the **+** button to add an element.
 - Click the newly added element and enter the name of the user group in the format *group@domain*.
 - Click **OK**.

Administering service groups

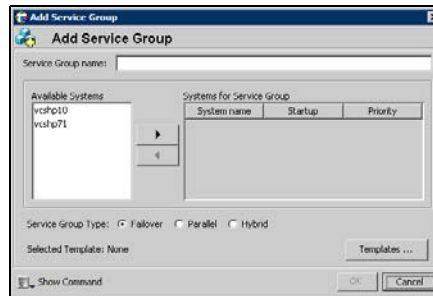
Use the Java Console to administer service groups in the cluster. Use the console to add and delete, bring online and take offline, freeze and unfreeze, link and unlink, enable and disable, autoenable, switch, and flush service groups. You can also modify the system list for a service group.

Adding a service group

The Java Console provides several ways to add a service group to the systems in a cluster. Use Cluster Explorer, Command Center, or the Template View to perform this task.

To add a service group from Cluster Explorer

- 1 On the **Edit** menu, click **Add**, and click **Service Group**.
or
Click **Add Service Group** in the Cluster Explorer toolbar.
- 2 Enter the details of the service group:

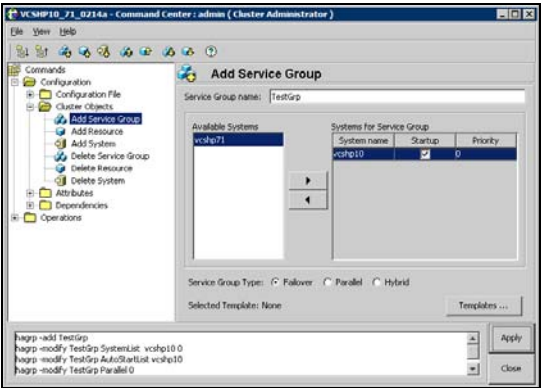


- Enter the name of the service group.
- In the **Available Systems** box, click the systems on which the service group will be added.
- Click the right arrow to move the selected systems to the **Systems for Service Group** box. The priority number (starting with 0) is automatically assigned to indicate the order of systems on which the service group will start in case of a failover. If necessary, double-click the entry in the **Priority** column to enter a new value.
Select the **Startup** check box to add the systems to the service groups AutoStartList attribute. This enables the service group to automatically come online on a system every time HAD is started.

- Click the appropriate service group type. A failover service group runs on only one system at a time; a parallel service group runs concurrently on multiple systems.
- To add a new service group based on a template, click **Templates...** Otherwise, proceed to step 2g. (Alternative method to add a new service group based on a template: From Cluster Explorer, click **Templates** on the **Tools** menu. Right-click the Template View panel, and click **Add as Service Group** from the menu.)
- Click the appropriate template name, then click **OK**.
- Click **Show Command** in the bottom left corner if you want to view the command associated with the service group. Click **Hide Command** to close the view of the command.
- Click **OK**.

To add a service group from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Configuration > Cluster Objects > Add Service Group**.
or
Click **Add service group** in the Command Center toolbar.
- 2 Enter the name of the service group.



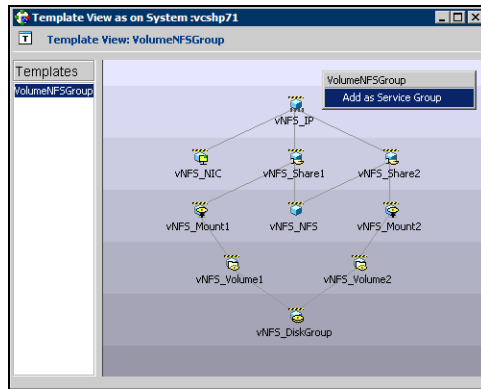
- 3 In the **Available Systems** box, click the systems on which the service group will be added.
- 4 Click the right arrow to move the selected systems to the **Systems for Service Group** box. The priority number (starting with 0) is automatically assigned to indicate the order of systems on which the service group will start in case of a failover. If necessary, double-click the entry in the **Priority** column to enter a new value.

Select the **Startup** check box to add the systems to the service groups `AutoStartList` attribute. This enables the service group to automatically come online on a system every time HAD is started.

- 5 Click the appropriate service group type. A failover service group runs on only one system at a time; a parallel service group runs concurrently on multiple systems.
- 6 To add a new service group based on a template, click **Templates...** Otherwise, proceed to step 9.
- 7 Click the appropriate template name.
- 8 Click **OK**.
- 9 Click **Apply**.

To add a service group from the template view

- 1 From Cluster Explorer, click **Templates...** on the **Tools** menu.
- 2 Right-click the Template View panel, and click **Add as Service Group** from the pop-up menu. This adds the service group template to the cluster configuration file without associating it to a particular system.



- 3 Use System Manager to add the service group to systems in the cluster. See "[System Manager](#)" on page 105.

Deleting a service group

Delete a service group from Cluster Explorer or Command Center.

Note: You cannot delete service groups with dependencies. To delete a linked service group, you must first delete the link.

To delete a service group from Cluster Explorer

- 1 In the **Service Groups** tab of the configuration tree, right-click the service group.
or
Click a cluster in the configuration tree, click the **Service Groups** tab, and right-click the service group icon in the view panel.
- 2 Click **Delete** from the menu.
- 3 Click **Yes**.

To delete a service group from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Configuration > Cluster Objects > Delete Service Group**.
- 2 Click the service group.
- 3 Click **Apply**.

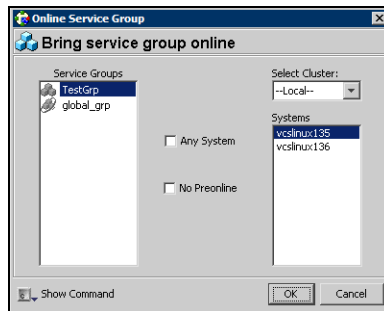
Bringing a service group online

To bring a service group online from the Cluster Explorer configuration tree

- 1 In the **Service Groups** tab of the configuration tree, right-click the service group.
or
Click a cluster in the configuration tree, click the **Service Groups** tab, and right-click the service group icon in the view panel.
- 2 Click **Online**, and click the appropriate system from the menu. Click **Any System** if you do not need to specify a system.

To bring a service group online from the Cluster Explorer toolbar

- 1 Click **Online Service Group** on the Cluster Explorer toolbar.
- 2 Specify the details for the service group:



- Click the service group.
- For global groups, select the cluster in which to bring the group online.
- Click the system on which to bring the group online, or select the **Any System** check box.
- Select the **No Preonline** check box to bring the service group online without invoking the preonline trigger.
- Click **Show Command** in the bottom left corner to view the command associated with the service group. Click **Hide Command** to close the view of the command.
- Click **OK**.

To bring a service group online from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Controls > Online Service Group**.
or
Click **Bring service group online** in the Command Center toolbar.
- 2 Click the service group.
- 3 For global groups, select the cluster in which to bring the group online.
- 4 Click the system on which to bring the group online, or select the **Any System** check box.
- 5 Click **Apply**.

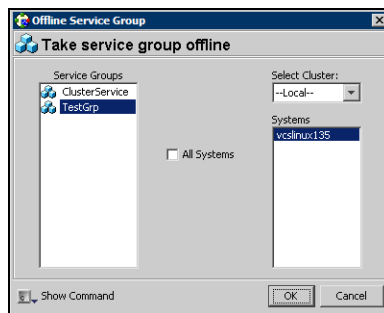
Taking a service group offline

To take a service group offline from Cluster Explorer configuration tree

- 1 In the **Service Groups** tab of the configuration tree, right-click the service group.
or
Click a cluster in the configuration tree, click the **Service Groups** tab, and right-click the service group icon in the view panel.
- 2 Click **Offline**, and click the appropriate system from the menu. Click **All Systems** to take the group offline on all systems.

To take a service group offline from the Cluster Explorer toolbar

- 1 Click **Offline Service Group** in the Cluster Explorer toolbar.
- 2 Enter the details of the service group:



- Click the service group.
- For global groups, select the cluster in which to take the group offline.
- Click the system on which to take the group offline, or click **All Systems**.
- Click **Show Command** in the bottom left corner if you want to view the command associated with the service group. Click **Hide Command** to close the view of the command.
- Click **OK**.

To take a service group offline from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Controls > Offline Service Group**.
or
Click **Take service group offline** in the Command Center toolbar.
- 2 Click the service group.
- 3 For global groups, select the cluster in which to take the group offline.
- 4 Click the system on which to take the group offline, or click the **All Systems** check box.
- 5 Click **Apply**.

Switching a service group

The process of switching a service group involves taking it offline on its current system and bringing it online on another system.

To switch a service group from Cluster Explorer

- 1 In the **Service Groups** tab of the configuration tree, right-click the service group.
or
Click the cluster in the configuration tree, click the **Service Groups** tab, and right-click the service group icon in the view panel.
- 2 Click **Switch To**, and click the appropriate system from the menu.

To switch a service group from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Controls > Switch Service Group**.
- 2 Click the service group.
- 3 For global groups, select the cluster in which to switch the service group.
- 4 Click the system on which to bring the group online, or select the **Any System** check box.
- 5 Click **Apply**.

Freezing a service group

Freeze a service group to prevent it from failing over to another system. The freezing process stops all online and offline procedures on the service group. Note that you cannot freeze a service group when the service group state is in transition.

To freeze a service group from Cluster Explorer

- 1 In the **Service Groups** tab of the configuration tree, right-click the service group.
or
Click the cluster in the configuration tree, click the **Service Groups** tab, and right-click the service group icon in the view panel.
- 2 Click **Freeze**, and click **Temporary** or **Persistent** from the menu. The persistent option maintains the frozen state after a reboot if you save this change to the configuration.

To freeze a service group from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Availability > Freeze Service Group**.
- 2 Click the service group.
- 3 Select the **persistent** check box if necessary. The persistent option maintains the frozen state after a reboot if you save this change to the configuration.
- 4 Click **Apply**.

Unfreezing a service group

Unfreeze a frozen service group to perform online or offline operations on the service group.

To unfreeze a service group from Cluster Explorer

- 1 In the **Service Groups** tab of the configuration tree, right-click the service group.
or
Click the cluster in the configuration tree, click the **Service Groups** tab, and right-click the service group icon in the view panel.
- 2 Click **Unfreeze**.

To unfreeze a service group from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Availability > Unfreeze Service Group**.
- 2 Click the service group.
- 3 Click **Apply**.

Enabling a service group

Enable a service group before bringing it online. A service group that was manually disabled during a maintenance procedure on a system may need to be brought online after the procedure is completed.

To enable a service group from Cluster Explorer

- 1 In the **Service Groups** tab of the configuration tree, right-click the service group.
or
Click the cluster in the configuration tree, click the **Service Groups** tab, and right-click the service group icon in the view panel.
- 2 Click **Enable**, and click the appropriate system from the menu. Click **All Systems** to enable the group on all systems.

To enable a service group from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Availability > Enable Service Group**.
- 2 Click the service group.
- 3 Select the **Per System** check box to enable the group on a specific system instead of all systems.
- 4 Click **Apply**.

Disabling a service group

Disable a service group to prevent it from coming online. This process temporarily stops VCS from monitoring a service group on a system undergoing maintenance operations.

To disable a service group from Cluster Explorer

- 1 In the **Service Groups** tab of the configuration tree, right-click the service group.
or
Click the cluster in the configuration tree, click the **Service Groups** tab, and right-click the service group icon in the view panel.
- 2 Click **Disable**, and click the appropriate system in the menu. Click **All Systems** to disable the group on all systems.

To disable a service group from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Availability > Disable Service Group**.
- 2 Click the service group.
- 3 Select the **Per System** check box to disable the group on a specific system instead of all systems.
- 4 Click **Apply**.

Autoenabling a service group

A service group is autodisabled until VCS probes all resources and checks that they are ready to come online. Autoenable a service group in situations where the VCS engine is not running on one of the systems in the cluster, and you must override the disabled state of the service group to enable the group on another system in the cluster.

To autoenable a service group from Cluster Explorer

- 1 In the **Service Groups** tab of the configuration tree, right-click the service group.
or
Click the cluster in the configuration tree, click the **Service Groups** tab, and right-click the service group icon in the view panel.
- 2 Click **Autoenable**, and click the appropriate system from the menu.

To autoenable a service group from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Availability > Autoenable Service Group**.
- 2 Click the service group.
- 3 Click the system on which to autoenable the group.
- 4 Click **Apply**.

Flushing a service group

As a service group is brought online or taken offline, the resources within the group are brought online and taken offline. If the online or offline operation hangs on a particular resource, flush the service group to halt the operation on the resources waiting to go online or offline. Flushing a service group typically leaves the cluster in a partial state. After completing this process, resolve the issue with the particular resource (if necessary) and proceed with starting or stopping the service group.

To flush a service group from Cluster Explorer

- 1 In the **Service Groups** tab of the configuration tree, right-click the service group.
or
Click the cluster in the configuration tree, click the **Service Groups** tab, and right-click the service group icon in the view panel.
- 2 Click **Flush**, and click the appropriate system from the menu.

To flush a service group from Command Center

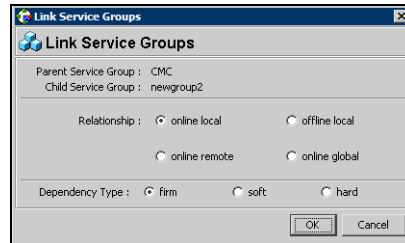
- 1 In the Command Center configuration tree, expand **Commands > Operations > Availability > Flush Service Group**.
- 2 Click the service group.
- 3 Click the system on which to flush the service group.
- 4 Click **Apply**.

Linking service groups

To link a service group from Cluster Explorer

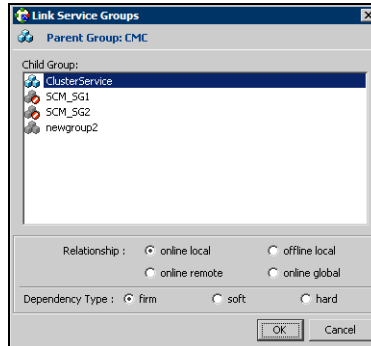
- 1 Click a cluster in the configuration tree.
- 2 In the view panel, click the **Service Groups** tab. This opens the service group dependency graph. To link a parent group with a child group:
 - Click **Link**.
 - Click the parent group.
 - Move the mouse toward the child group. The yellow line “snaps” to the child group. If necessary, press Esc on the keyboard to delete the line between the parent and the pointer before it snaps to the child.
 - Click the child group.
 - In the Link Service Groups dialog box, click the group relationship and dependency type.

See “[About service group dependencies](#)” on page 410.



- Click **OK**.

You can also link the service groups by performing steps 1 and 2, right-clicking the parent group, and clicking **Link** from the menu. In the dialog box, click the child group, relationship, dependency type, and click **OK**.



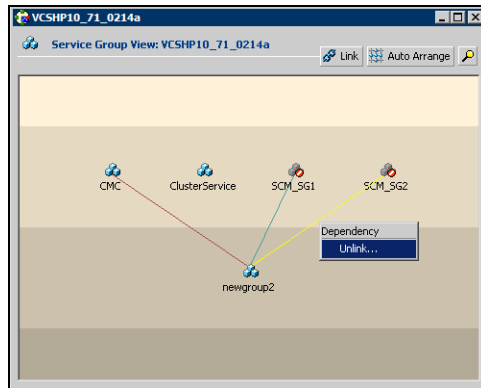
To link a service group from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Configuration > Dependencies > Link Service Groups**.
- 2 Click the parent resource group in the **Service Groups** box. After selecting the parent group, the potential groups that can serve as child groups are displayed in the **Child Service Groups** box.
- 3 Click a child service group.
- 4 Click the group relationship and dependency type.
See "[About service group dependencies](#)" on page 410.
- 5 Click **Apply**.

Unlinking service groups

To delete a service group dependency from Cluster Explorer

- 1 Click a cluster in the configuration tree.
- 2 In the view panel, click the **Service Groups** tab.
- 3 In the Service Group view, right-click the link between the service groups.
- 4 Click **Unlink** from the menu.



- 5 Click **Yes**.

To delete a service group dependency from Command Center

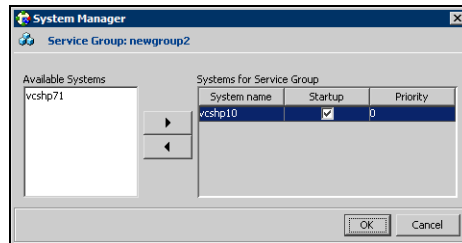
- 1 In the Command Center configuration tree, expand **Commands > Configuration > Dependencies > Unlink Service Groups**.
- 2 Click the parent resource group in the **Service Groups** box. After selecting the parent group, the corresponding child groups are displayed in the **Child Service Groups** box.
- 3 Click the child service group.
- 4 Click **Apply**.

Managing systems for a service group

From Cluster Explorer, use System Manager to add and remove systems in a service group's system list.

To add a system to the service group's system list

- 1 In the System Manager dialog box, click the system in the **Available Systems** box.



- 2 Click the right arrow to move the available system to the **Systems for Service Group** table.
- 3 Select the **Startup** check box to add the systems to the service groups AutoStartList attribute. This enables the service group to automatically come online on a system every time HAD is started.
- 4 The priority number (starting with 0) is assigned to indicate the order of systems on which the service group will start in case of a failover. If necessary, double-click the entry in the **Priority** column to enter a new value.
- 5 Click **OK**.

To remove a system from the service group's system list

- 1 In the System Manager dialog box, click the system in the **Systems for Service Group** table.
- 2 Click the left arrow to move the system to the **Available Systems** box.
- 3 Click **OK**.

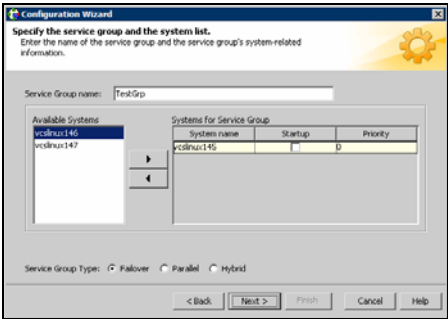
Creating service groups with the configuration wizard

This section describes how to create service groups using the configuration wizard.

Note: VCS also provides wizards to create service groups for applications and NFS shares. See the chapter “Configuring applications and resources in VCS” for more information about these wizards.

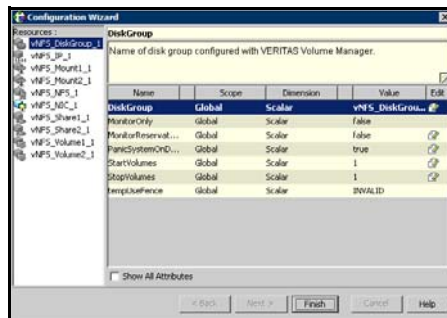
To create a service group using the configuration wizard

- 1 Open the Configuration Wizard. From Cluster Explorer, click **Configuration Wizard** on the **Tools** menu.
- 2 Read the information on the Welcome dialog box and click **Next**.
- 3 Specify the name and target systems for the service group:



- Enter the name of the group.
- Click the target systems in the **Available Systems** box.
- Click the right arrow to move the systems to the **Systems for Service Group** table. To remove a system from the table, click the system and click the left arrow.
- Select the **Startup** check box to add the systems to the service groups AutoStartList attribute. This enables the service group to automatically come online on a system every time HAD is started.
- The priority number (starting with 0) is automatically assigned to indicate the order of systems on which the service group will start in case of a failover. If necessary, double-click the entry in the **Priority** column to enter a new value.
- Click the service group type.

- Click **Next**.
- 4 Click **Next** again to configure the service group with a template and proceed to step 7. Click **Finish** to add an empty service group to the selected cluster systems and configure it at a later time.
- 5 Click the template on which to base the new service group. The Templates box lists the templates available on the system to which Cluster Manager is connected. The resource dependency graph of the templates, the number of resources, and the resource types are also displayed. Click **Next**.
- 6 If a window notifies you that the name of the service group or resource within the service group is already in use, proceed to step 9. Otherwise, proceed to step 10.
- 7 Click **Next** to apply all of the new names listed in the table to resolve the name clash.
or
 Modify the clashing names by entering text in the field next to the **Apply** button, clicking the location of the text for each name from the **Correction** drop-down list box, clicking **Apply**, and clicking **Next**.
- 8 Click **Next** to create the service group. A progress indicator displays the status.
- 9 After the service group is successfully created, click **Next** to edit attributes using the wizard. Click **Finish** to edit attributes at a later time using Cluster Explorer.
- 10 Review the attributes associated with the resources of the service group. If necessary, proceed to step 11 to modify the default values of the attributes. Otherwise, proceed to step 12 to accept the default values and complete the configuration.
- 11 Modify the values of the attributes (if necessary).



- Click the resource.

Administering service groups

- Click the attribute to be modified.
- Click the **Edit** icon at the end of the table row.
- In the Edit Attribute dialog box, enter the attribute values.
- Click **OK**.
- Repeat the procedure for each resource and attribute.

12 Click **Finish**.

Administering resources

Use the Java Console to administer resources in the cluster. Use the console to add and delete, bring online and take offline, probe, enable and disable, clear, and link and unlink resources. You can also import resource types to the configuration.

Adding a resource

The Java Console provides several ways to add a resource to a service group. Use Cluster Explorer or Command Center to perform this task.

To add a resource from Cluster Explorer

- 1 In the **Service Groups** tab of the Cluster Explorer configuration tree, click a service group to which the resource will be added.
- 2 On the **Edit** menu, click **Add**, and click **Resource**.
or
Click **Add Resource** in the Cluster Explorer toolbar.
- 3 Enter the details of the resource:
 - Enter the name of the resource.

Attribute name	Type	Dimension	Value	Edit
PathName	String	Scalar		

- Click the resource type.
- Edit resource attributes according to your configuration. The Java Console also enables you to edit attributes after adding the resource.
- Select the **Critical** and **Enabled** check boxes, if applicable. The **Critical** option is selected by default.

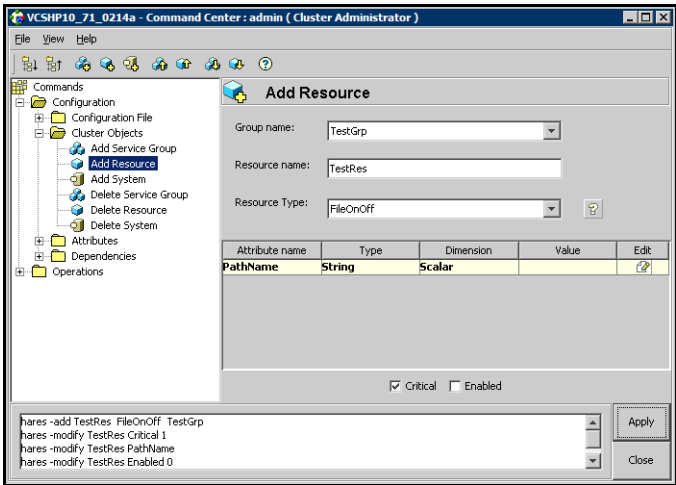
A critical resource indicates the service group is faulted when the resource, or any resource it depends on, faults. An enabled resource indicates agents monitor the resource; you must specify the values of mandatory attributes before enabling a resource. If a resource is

created dynamically while VCS is running, you must enable the resource before VCS monitors it. VCS will not bring a disabled resource nor its children online, even if the children are enabled.

- Click **Show Command** in the bottom left corner to view the command associated with the resource. Click **Hide Command** to close the view of the command.
- Click **OK**.

To add a resource from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Configuration > Cluster Objects > Add Resource**.
or
Click **Add resource** in the Command Center toolbar.



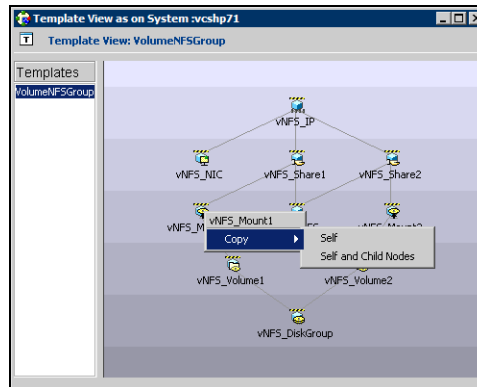
- 2 Select the service group to contain the resource.
- 3 Enter the name of the resource.
- 4 Click the resource type.
- 5 Edit resource attributes according to your configuration. The Java Console also enables you to edit attributes after adding the resource.
- 6 Select the **Critical** and **Enabled** check boxes, if applicable. The **Critical** option is selected by default.
A critical resource indicates the service group is faulted when the resource, or any resource it depends on, faults. An enabled resource indicates agents monitor the resource; you must specify the values of mandatory attributes

before enabling a resource. If a resource is created dynamically while VCS is running, you must enable the resource before VCS monitors it. VCS will not bring a disabled resource nor its children online, even if the children are enabled.

- 7 Click **Apply**.

To add a resource from the Template view

- 1 From Cluster Explorer, click **Templates...** on the **Tools** menu.
- 2 In the left pane of the Template View, click the template from which to add resources to your configuration.
- 3 In the resource graph, right-click the resource to be added to your configuration.



- 4 Click **Copy**, and click **Self** from the menu to copy the resource. Click **Copy**, and click **Self and Child Nodes** from the menu to copy the resource with its dependent resources.
- 5 In the **Service Groups** tab of the Cluster Explorer configuration tree, click the service group to which to add the resources.
- 6 In the Cluster Explorer view panel, click the **Resources** tab.
- 7 Right-click the Resource view panel and click **Paste** from the menu. After the resources are added to the service group, edit the attributes to configure the resources.

Adding a RemoteGroup resource from the Java Console

A RemoteGroup resource is typically useful in scenarios where resources configured in a local service group are dependant on the state of a remote failover service group. For example, a web-server application running in a local cluster could be dependant on a database application running in a remote cluster.

Note that the RemoteGroup agent represents that state of a failover service group; the agent is not supported with parallel service groups.

A RemoteGroup resource monitors the state of a remote service group in a local cluster. Once you have added the RemoteGroup resource to a local service group, you can link the resource to the existing resources of the service group.

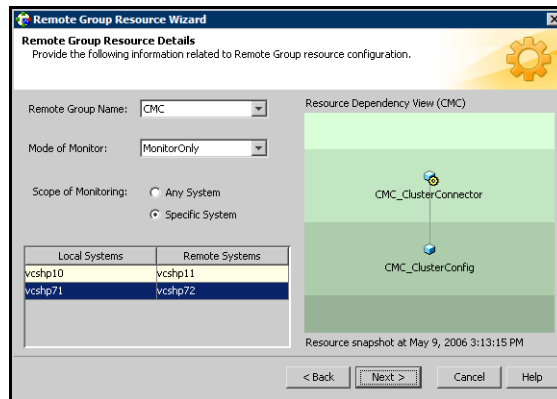
You must have administrative privileges to configure RemoteGroup resources.

See “[Configuring the RemoteGroup agent](#)” on page 251.

To add a RemoteGroup resource

- 1 On the **Tools** menu, click **Add Remote Group Resource...**
or
Click **Configure Remote Group Resource Wizard** in the Cluster Explorer toolbar.
- 2 Read the information on the Welcome dialog box and click **Next**.
- 3 In the Remote Group Resource Name dialog box, specify the name of the resource and the service group to which the resource will be added. Click **Next**.
- 4 In the Remote Cluster Information dialog box:
 - Specify the name or IP address of a node in the remote cluster.
 - Specify the port on the remote node on which the resource will communicate.
 - Specify a username for the remote cluster.
 - Specify a password for the user.
 - Select the check box if you wish to specify advance options to connect to a cluster running in secure mode. Otherwise, proceed to the last step.
 - Specify the domain of which the node is a part.
 - Select a domain type.
 - Specify the authentication broker and port.
 - Click **Next**.

5 In the Remote Group Resource Details dialog box:



- Select a group you wish to monitor.
- Select the mode of monitoring.
 - Choose the **MonitorOnly** option to monitor the remote service group. You will not be able to perform online or offline operations on the remote group.
 - Choose the **OnlineOnly** option to monitor the remote service group and bring the remote group online from the local cluster.
 - Choose the **OnOff** option to monitor the remote service group, bring the remote group online, and take it offline from the local cluster.
- Specify whether the RemoteGroup resource should monitor the state of the remote group on a specific system or any system in the remote cluster.
 - Choose the **Any System** option to enable the RemoteGroup resource to monitor the state of the remote service group irrespective of the system on which it is online.
 - Choose the **Specific System** option to enable the RemoteGroup resource to monitor the state of the remote group on a specific system in the remote cluster. Both service groups must be configured on the same number of systems.

This option provides one-to-one mapping between the local and remote systems. The **Local Systems** list displays the systems on which the RemoteGroup resource is configured. Click the fields under the **Remote Systems** list and select the systems from drop-down list. If the remote group fails over to another system in

the remote cluster, the RemoteGroup resource will also fail over to the corresponding system in the local cluster.

- Click **Next**.

- 6 Review the text in the dialog box and click **Finish** to add the RemoteGroup resource to the specified service group in the local cluster. You must now create dependencies between the RemoteGroup resource and the existing resources of the service group.

See “[Linking resources](#)” on page 156.

Deleting a resource

To delete a resource from Cluster Explorer

- 1 In the **Service Groups** tab of the configuration tree, right-click the resource.
or
Click a service group in the configuration tree, click the **Resources** tab, and right-click the resource icon in the view panel.
- 2 Click **Delete** from the menu.
- 3 Click **Yes**.

To delete a resource from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Configuration > Cluster Objects > Delete Resource**.
- 2 Click the resource.
- 3 Click **Apply**.

Bringing a resource online

To bring a resource online from Cluster Explorer

- 1 In the **Service Groups** tab of the configuration tree, right-click the resource.
or
Click a service group in the configuration tree, click the **Resources** tab, and right-click the resource icon in the view panel.
- 2 Click **Online**, and click the appropriate system from the menu.

To bring a resource online from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Controls > Online Resource**.
- 2 Click a resource.

- 3 Click a system on which to bring the resource online.
- 4 Click **Apply**.

Taking a resource offline

To take a resource offline from Cluster Explorer

- 1 In the **Service Groups** tab of the configuration tree, right-click the resource.
or
Click a service group in the configuration tree, click the **Resources** tab, and right-click the resource icon in the view panel.
- 2 Click **Offline**, and click the appropriate system from the menu.

To take a resource offline from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Controls > Offline Resource**.
- 2 Click a resource.
- 3 Click a system on which to take the resource offline.
- 4 If necessary, select the **ignoreparent** check box to take a selected child resource offline, regardless of the state of the parent resource. This option is only available through Command Center.
- 5 Click **Apply**.

Taking a resource offline and propagating the command

Use the Offline Propagate (OffProp) feature to propagate the offline state of a parent resource. This command signals that resources dependent on the parent resource should also be taken offline.

Use the Offline Propagate (OffProp) “ignoreparent” feature to take a selected resource offline, regardless of the state of the parent resource. This command propagates the offline state of the selected resource to the child resources. The “ignoreparent” option is only available in Command Center.

To take a resource and its child resources offline from Cluster Explorer

- 1 In the Resources tab of the configuration tree, right-click the resource.
- 2 Click **Offline Prop**, and click the appropriate system from the menu.

To take a resource and its child resources offline from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Controls > OffProp Resource**.
- 2 Click the resource.
- 3 Click the system on which to take the resource, and the child resources, offline.
- 4 Click **Apply**.

To take child resources offline from Command Center while ignoring the state of the parent resource

- 1 In the Command Center configuration tree, expand **Commands > Operations > Controls > OffProp Resource**.
- 2 Click the resource.
- 3 Click the system on which to take the resource, and the child resources, offline.
- 4 Select the **ignoreparent** check box.
- 5 Click **Apply**.

Probing a resource

Probe a resource to check that it is configured and ready to bring online.

To probe a resource from Cluster Explorer

- 1 In the **Service Groups** tab of the configuration tree, right-click the resource.
- 2 Click **Probe**, and click the appropriate system from the menu.

To probe a resource from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Controls > Probe Resource**.
- 2 Click the resource.
- 3 Click the system on which to probe the resource.
- 4 Click **Apply**.

Overriding resource type static attributes

You can override some resource type static attributes and assign them resource-specific values. When a static attribute is overridden and the configuration is saved, the main.cf file includes a line in the resource definition for the static attribute and its overridden value.

To override resource type static attribute

- 1 Right-click the resource in the **Service Groups** tab of the configuration tree or in the **Resources** tab of the view panel.
- 2 Click **Override Attributes**.
- 3 Select the attributes to override.
- 4 Click **OK**.
The selected attributes appear in the Overridden Attributes table in the Properties view for the resource.
- 5 To modify the default value of an overridden attribute, click the icon in the **Edit** column of the attribute.

To restore default settings to a type's static attribute

- 1 Right-click the resource in the **Service Groups** tab of the configuration tree or in the **Resources** tab of the view panel.
- 2 Click **Remove Attribute Overrides**.
- 3 Select the overridden attributes to be restored to their default settings.
- 4 Click **OK**.

Enabling resources in a service group

Enable resources in a service group to bring the disabled resources online. A resource may have been manually disabled to temporarily stop VCS from monitoring the resource. You must specify the values of mandatory attributes before enabling a resource.

To enable an individual resource in a service group

- 1 From Cluster Explorer, click the **Service Groups** tab of the configuration tree.
- 2 Right-click a disabled resource in the configuration tree, and click **Enabled** from the menu.

To enable all resources in a service group from Cluster Explorer

- 1 From Cluster Explorer, click the **Service Groups** tab in the configuration tree.
- 2 Right-click the service group.
- 3 Click **Enable Resources**.

To enable all resources in a service group from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Availability > Enable Resources for Service Group**.
- 2 Click the service group.
- 3 Click **Apply**.

Disabling resources in a service group

Disable resources in a service group to prevent them from coming online. This disabling process is useful when you want VCS to temporarily “ignore” resources (rather than delete them) while the service group is still online.

To disable an individual resource in a service group

- 1 From Cluster Explorer, click the **Service Groups** tab in the Cluster Explorer configuration tree.
- 2 Right-click a resource in the configuration tree. An enabled resource will display a check mark next to the **Enabled** option that appears in the menu.
- 3 Click **Enabled** from the menu to clear this option.

To disable all resources in a service group from Cluster Explorer

- 1 From Cluster Explorer, click the **Service Groups** tab in the configuration tree.
- 2 Right-click the service group and click **Disable Resources**.

To disable all resources in a service group from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Availability > Disable Resources for Service Group**.
- 2 Click the service group.
- 3 Click **Apply**.

Clearing a resource

Clear a resource to remove a fault and make the resource available to go online. A resource fault can occur in a variety of situations, such as a power failure or a faulty configuration.

To clear a resource from Cluster Explorer

- 1 In the **Service Groups** tab of the configuration tree, right-click the resource.
- 2 Click **Clear Fault**, and click the system from the menu. Click **Auto** instead of a specific system to clear the fault on all systems where the fault occurred.

To clear a resource from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Availability > Clear Resource**.
- 2 Click the resource. To clear the fault on all systems listed in the **Systems** box, proceed to step 5. To clear the fault on a specific system, proceed to step 3.
- 3 Select the **Per System** check box.
- 4 Click the system on which to clear the resource.
- 5 Click **Apply**.

Linking resources

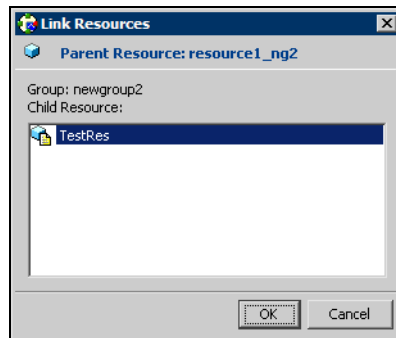
Use Cluster Explorer or Command Center to link resources in a service group.

To link resources from Cluster Explorer

- 1 In the configuration tree, click the **Service Groups** tab.
- 2 Click the service group to which the resources belong.
- 3 In the view panel, click the **Resources** tab. This opens the resource dependency graph. To link a parent resource with a child resource:
 - Click **Link...**
 - Click the parent resource.
 - Move the mouse towards the child resource. The yellow line “snaps” to the child resource. If necessary, press Esc to delete the line between the parent and the pointer before it snaps to the child.
 - Click the child resource.
 - In the Confirmation dialog box, click **Yes**.

or

Right-click the parent resource, and click **Link** from the menu. In the Link Resources dialog box, click the resource that will serve as the child. Click **OK**.

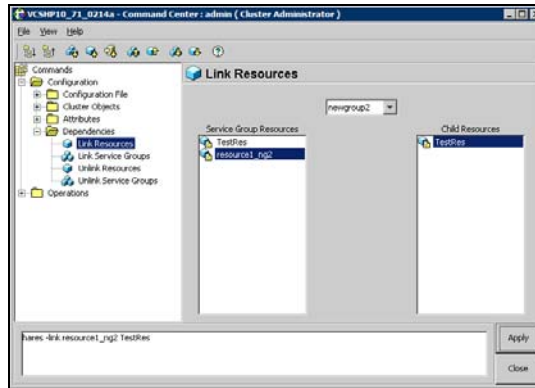


- Click **OK**.

To link resources from Command Center

- 1 In the Command Center configuration tree, expand **Commands** > **Configuration** > **Dependencies** > **Link Resources**.
- 2 Click the service group to contain the linked resources.

- 3 Click the parent resource in the **Service Group Resources** box. After selecting the parent resource, the potential resources that can serve as child resources are displayed in the **Child Resources** box.



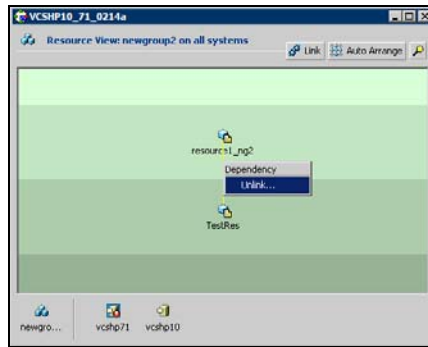
- 4 Click a child resource.
- 5 Click **Apply**.

Unlinking resources

Use Cluster Explorer or Command Center to unlink resources in a service group.

To unlink resources from Cluster Explorer

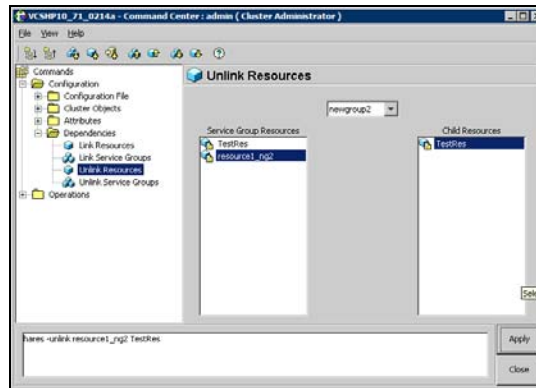
- 1 From the configuration tree, click the **Service Groups** tab.
- 2 Click the service group to which the resources belong.
- 3 In the view panel, click the **Resources** tab.
- 4 In the Resources View, right-click the link between the resources.
- 5 Click **Unlink...** from the menu.



- 6 In the Question dialog box, click **Yes** to delete the link.

To unlink resources from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Configuration > Dependencies > Unlink Resources**.
- 2 Click the service group that contains the linked resources.
- 3 Click the parent resource in the **Service Group Resources** box. After selecting the parent resource, the corresponding child resources are displayed in the **Child Resources** box.



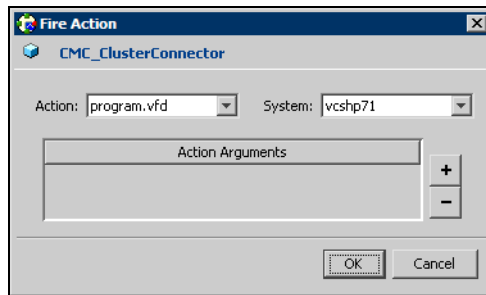
- 4 Click the child resource.
- 5 Click **Apply**.

Invoking a resource action

Cluster Explorer enables you to initiate a predefined action script. Some examples of predefined resource actions are splitting and joining disk groups.

To invoke a resource action

- 1 In the **Service Groups** tab of the configuration tree, right-click the resource.
- 2 Click **Actions...**
- 3 Specify the details of the action:



- Click the predefined action to execute.
- Click the system on which to execute the action.
- To add an argument, click the **Add** icon (+) and enter the argument. Click the **Delete** icon (-) to remove an argument.
- Click **OK**.

Refreshing the ResourceInfo attribute

Refresh the ResourceInfo attribute to view the latest values for that attribute.

To refresh the ResourceInfo attribute

- 1 In the **Service Groups** tab of the configuration tree, right-click the resource.
- 2 Click **Refresh ResourceInfo**, and click the system on which to refresh the attribute value.

Clearing the ResourceInfo attribute

Clear the ResourceInfo attribute to reset all the parameters in this attribute.

To clear the parameters of the ResourceInfo attribute

- 1 In the **Service Groups** tab of the configuration tree, right-click the resource.
- 2 Click **Clear ResourceInfo**, and click the system on which to reset the attribute value.

Importing resource types

The Java Console enables you to import resource types into your configuration (main.cf). For example, use this procedure to import the types.cf for enterprise agents to your configuration. You cannot import resource types that already exist in your configuration.

To import a resource type from Cluster Explorer

- 1 On the **File** menu, click **Import Types**.
- 2 In the Import Types dialog box:
 - Click the file from which to import the resource type. The dialog box displays the files on the system that Cluster Manager is connected to.
 - Click **Import**.

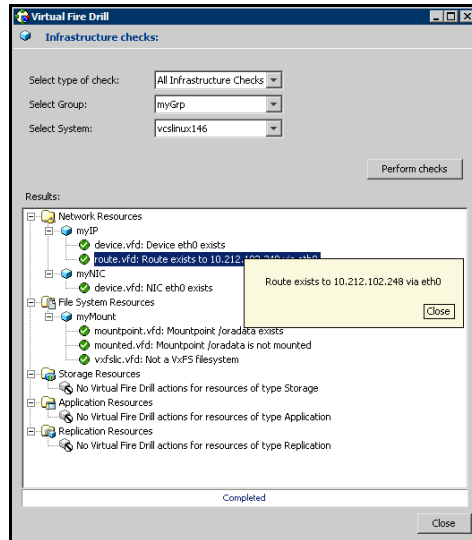
Running HA fire drill from the Java Console

Use the Cluster Manager to run HA fire drills for specific resources in a local cluster. You can run HA fire drill for agents that support the functionality.

To run HA fire drill

- 1 On the Cluster Explorer toolbar, click **Virtual Fire Drill**.
or
From Cluster Explorer, click **Virtual Fire Drill...** on the **Tools** menu.

2 Specify details to run a virtual fire drill.



- Select the type of check to run.
- Select a service group for which to run the infrastructure checks. Make sure you select a service group that is online.
- Select a system to run the checks on.
- Click **Perform checks**.
- View the result of the check. If the virtual fire drill reports any errors, right-click the resource and select **Fix it...**

3 Click **Close**.

Administering systems

Use the Java Console to administer systems in the cluster. Use the console to add, delete, freeze, and unfreeze systems.

Adding a system

Cluster Explorer and Command Center enable you to add a system to the cluster. A system must have an entry in the LLTTab configuration file before it can be added to the cluster.

To add a system from Cluster Explorer

- 1 On the **Edit** menu, click **Add**, and click **System**.
or
Click **Add System** on the Cluster Explorer toolbar.
- 2 Enter the name of the system.
- 3 Click **Show Command** in the bottom left corner to view the command associated with the system. Click **Hide Command** to close the view of the command.
- 4 Click **OK**.

To add a system from Command Center

- 1 Click **Add System** in the Command Center toolbar.
or
In the Command Center configuration tree, expand **Commands > Configuration > Cluster Objects > Add System**.
- 2 Enter the name of the system.
- 3 Click **Apply**.

Deleting a system

To delete a system from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Configuration > Cluster Objects > Delete System**.
- 2 Click the system.
- 3 Click **Apply**.

Freezing a system

Freeze a system to prevent service groups from coming online on the system.

To freeze a system from Cluster Explorer

- 1 Click the **Systems** tab of the configuration tree.
- 2 In the configuration tree, right-click the system, click **Freeze**, and click **Temporary** or **Persistent** from the menu. The persistent option maintains the frozen state after a reboot if the user saves this change to the configuration.

To freeze a system from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Availability > Freeze System**.
- 2 Click the system.
- 3 If necessary, select the **persistent** and **evacuate** check boxes. The evacuate option moves all service groups to a different system before the freeze operation takes place. The persistent option maintains the frozen state after a reboot if the user saves this change to the configuration.
- 4 Click **Apply**.

Unfreezing a system

Unfreeze a frozen system to enable service groups to come online on the system.

To unfreeze a system from Cluster Explorer

- 1 Click the **Systems** tab of the configuration tree.
- 2 In the configuration tree, right-click the system and click **Unfreeze**.

To unfreeze a system from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Operations > Availability > Unfreeze System**.
- 2 Click the system.
- 3 Click **Apply**.

Administering clusters

Use the Java Console to specify the clusters you want to view from the console, and to modify the VCS configuration. The configuration details the parameters of the entire cluster. Use Cluster Explorer or Command Center to open, save, and “save and close” a configuration. VCS Simulator enables you to administer the configuration on the local system while VCS is offline.

Opening a cluster configuration

Use Cluster Explorer or Command Center to open or make changes to the VCS configuration.

To open a configuration from Cluster Explorer

On the File menu, click **Open Configuration**.

or

Click **Open Configuration** on the Cluster Explorer toolbar.

To open a configuration from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Configuration > Configuration File > Open Configuration**.
- 2 Click **Apply**.

Saving a cluster configuration

After updating the VCS configuration, use Cluster Explorer or Command Center to save the latest configuration to disk while maintaining the configuration state in read-write mode.

To save a configuration from Cluster Explorer

On the **File** menu, click **Save Configuration**.

or

Click **Save Configuration** on the Cluster Explorer toolbar.

To save a configuration from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Configuration > Configuration File > Save Configuration**.
- 2 Click **Apply**.

Saving and closing a cluster configuration

After updating the VCS configuration, use Cluster Explorer or Command Center to save the latest configuration to disk, and close or change the configuration state to read-only mode.

To save and close a configuration from Cluster Explorer

On the **File** menu, click **Close Configuration**.

or

Click **Save and Close Configuration** on the Cluster Explorer toolbar.

To save and close a configuration from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Configuration > Configuration File > Close Configuration**.
- 2 Click **Apply**.

Executing commands

Use Command Center to execute commands on a cluster. Command Center enables you to run commands organized as “Configuration” and “Operation.”

To execute a command from Command Center

- 1 From Command Center, click the command from the command tree. If necessary, expand the tree to view the command.
- 2 In the corresponding command interface, click the VCS objects and appropriate options (if necessary).
- 3 Click **Apply**.

Editing attributes

Use the Java Console to edit attributes of VCS objects. By default, the Java Console displays key attributes and type specific attributes. To view all attributes associated with an object, click **Show all attributes**.

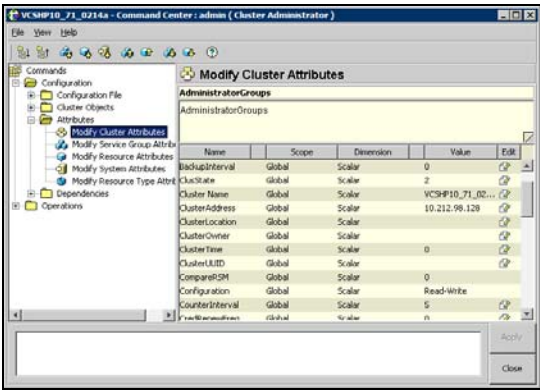
To edit an attribute from Cluster Explorer

- 1 From the Cluster Explorer configuration tree, click the object whose attributes you want to edit.
- 2 In the view panel, click the **Properties** tab. If the attribute does not appear in the Properties View, click **Show all attributes**. This opens the Attributes View.
- 3 In the Properties or Attributes View, click the icon in the **Edit** column of the **Key Attributes** or **Type Specific Attributes** table. In the Attributes View, click the icon in the **Edit** column of the attribute.
- 4 In the Edit Attribute dialog box, enter the changes to the attribute values.
To edit a scalar value:
Enter or click the value.
To edit a non-scalar value:
Use the + button to add an element. Use the - button to delete an element.
To change the attribute's scope:
Click the **Global** or **Per System** option.
To change the system for a local attribute:
Click the system from the menu.
- 5 Click **OK**.

To edit an attribute from Command Center

- 1 In the Command Center configuration tree, expand **Commands > Configuration > Attributes > Modify vcs_object Attributes**.
- 2 Click the VCS object from the menu.

- 3
- In the attribute table, click the icon in the **Edit** column of the attribute.



- 4
- In the Edit Attribute dialog box, enter the changes to the attribute values.

To edit a scalar value:
Enter or click the value.

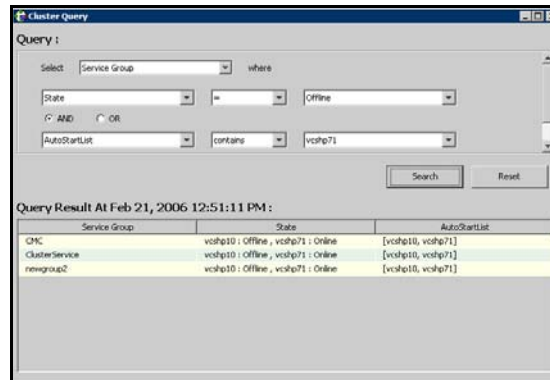
To edit a non-scalar value:
Use the + button to add an element. Use the - button to delete an element.

To change the attribute's scope:
Click the **Global** or **Per System** option.

To change the system for a local attribute:
Click the system from the menu.
- 5
- Click **OK**.

Querying the cluster configuration

- 1 From Cluster Explorer, click **Query** on the **Tools** menu.
or
 On the Cluster Explorer toolbar, click **Query**.
- 2 Enter the details of the query:



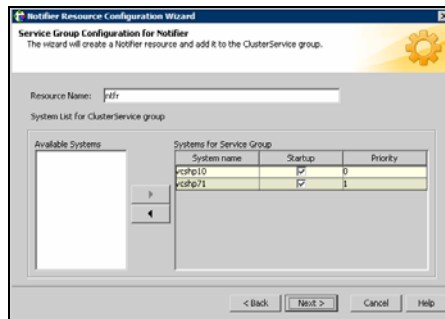
- Click the VCS object to search.
- Depending on the selected object, click the specific entity to search.
- Click the appropriate phrase or symbol between the search item and value.
- Click the appropriate value for the specified query. Certain queries allow the user to enter specific filter information:
 Click **System**, click **Online Group Count**, click **<**, and type the required value in the blank field.
or
 Click **Resource**, click **[provide attribute name]** and type in the name of an attribute, click **=** or **contains**, and type the appropriate value of the attribute in the blank field. For example, click **Resource**, click **[provide attribute name]** and type in pathname, click **contains**, and type **c:\temp** in the blank field.
- To use additional queries, click **+** as many times as necessary to select the appropriate options. Click **-** to reduce the number of queries.
- Click **AND** or **OR** for each filter selection.
- Click **Search**. The results appear in tabular format at the bottom of the dialog box. To search a new item, click **Reset** to reset the dialog box to its original blank state.

Setting up VCS event notification using the Notifier wizard

The information presented here assumes that you need to create both the ClusterService group and the Notifier resource. If the ClusterService group exists but the Notifier resource is configured under another group, you can modify the attributes of the existing Notifier resource and system list for that group. If the ClusterService group is configured but the Notifier resource is not configured, the Notifier resource will be created and added to the ClusterService group.

To set up event notification using the Notifier wizard

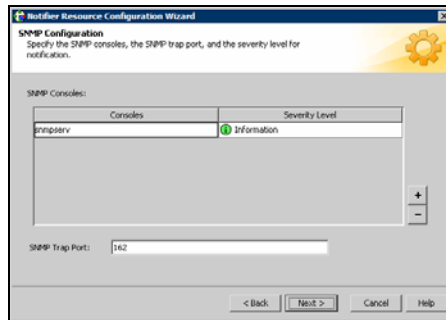
- 1 From Cluster Explorer, click **Notifier Wizard...** on the **Tools** menu.
or
On the Cluster Explorer toolbar, click **Launch Notifier Resource Configuration Wizard**.
- 2 Click **Next**.
- 3 In the dialog box:



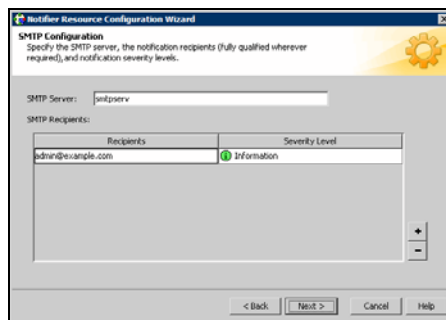
- Enter the name of the resource. For example, "ntfr".
- Click the target systems in the **Available Systems** box.
- Click the right arrow to move the systems to the **Systems for Service Group** table. To remove a system from the table, click the system and click the left arrow.
- Select the **Startup** check box to add the systems to the service groups AutoStartList attribute. This enables the service group to automatically come online on a system every time HAD is started.
- The priority number (starting with 0) is assigned to indicate the order of systems on which the service group will start in case of a failover. If

necessary, double-click the entry in the **Priority** column to enter a new value.

- Click **Next**.
- 4 Choose the mode of notification which needs to be configured. Select the check boxes to configure SNMP and/or SMTP (if applicable).
- 5 In the SNMP Configuration dialog box (if applicable):



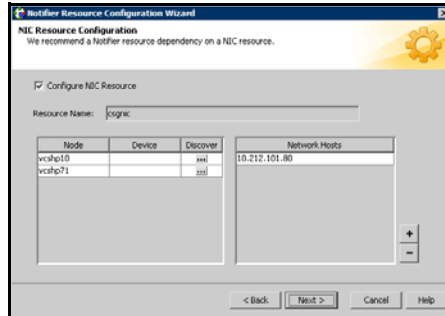
- Click + to create the appropriate number of fields for the SNMP consoles and severity levels. Click - to remove a field.
- Enter the console and click the severity level from the menu. For example, "snmpserv" and "Information".
- Enter the SNMP trap port. For example, "162" is the default value.
- Click **Next**.
- 6 In the SMTP Configuration dialog box (if applicable):



- Enter the name of the SMTP server.
- Click + to create the appropriate number of fields for recipients of the notification and severity levels. Click - to remove a field.

- Enter the recipient and click the severity level in the drop-down list box. For example, “admin@example.com” and “Information”.
- Click **Next**.

7 In the NIC Resource Configuration dialog box:

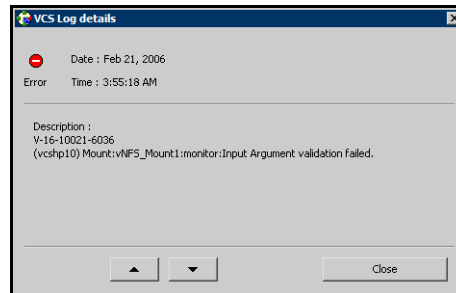


- Click **Configure NIC Resource** (recommended by Symantec) and proceed to the next step. Otherwise, click **Next**.
 - If necessary, enter the name of the resource.
 - Click the icon (...) in the **Discover** column of the table to find the MACAddress for each system.
 - Click **OK** on the Discover dialog box.
 - Click **Next**.
- 8 Click the **Bring the Notifier Resource Online** check box, if desired.
- 9 Click **Next**.
- 10 Click **Finish**.

Administering logs

The Java Console enables you to customize the log display of messages generated by the engine. In the Logs dialog box, you can set filter criteria to search and view messages, and monitor and resolve alert messages.

To browse the logs for detailed views of each log message, double-click the event's description. Use the arrows in the **VCS Log details** pop-up window to navigate backward and forward through the message list.

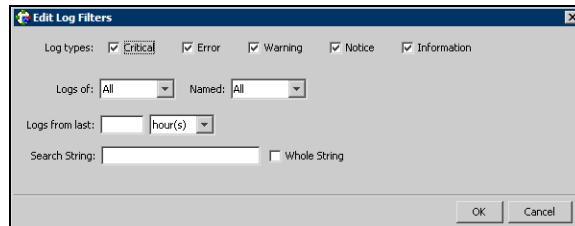


Customizing the log display

From the Logs dialog box, use the **Edit Filters** feature to customize the display of log messages.

To customize the display for VCS logs

- 1 In the **VCS Logs** tab, click **Edit Filters**.
- 2 Enter the filter criteria:
 - Click the types of logs to appear on the message display.

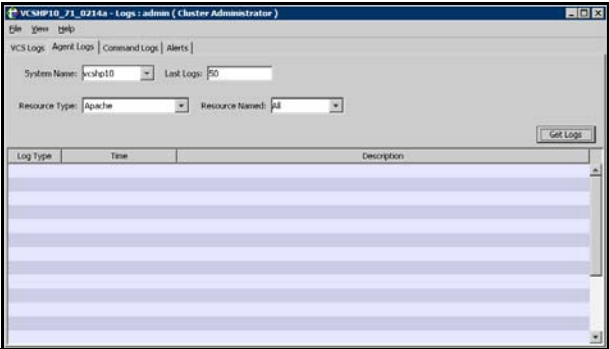


- From the **Logs of** list, select the category of log messages to display.
- From the **Named** menu, select the name of the selected object or component. To view all the messages for the selected category, click **All**.

- In the **Logs from last** field, enter the numerical value and select the time unit.
- To search log messages, enter the search string. Select the **Whole String** check box, if required.
- Click **OK**.

To customize the display for agent logs

- 1 In the **Agent Logs** tab, enter the filter criteria:



- Click the name of the system.
- Enter the number of logs to view.
- Click the resource type.
- Click the name of the resource. To view messages for all resources, click **All**.
- Click **Get Logs**.

Resetting the log display

Use the **Reset Filters** feature to set the default settings for the log view. For example, if you customized the log view to only show critical and error messages using the **Edit Filters** feature, the **Reset Filters** feature will set the view to show all log messages.

To reset the default settings for the log display

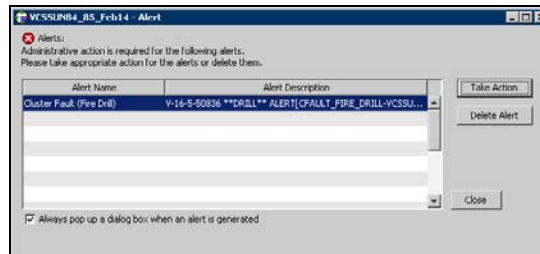
In the **VCS Logs** tab, click **Reset Filters**.

Monitoring alerts

The Java Console sends automatic alerts that require administrative action and are displayed on the **Alerts** tab of the Logs dialog box. Use this tab to take action on the alert or delete the alert.

To take action on an alert

- 1 In the **Alert** tab or dialog box, click the alert to take action on.

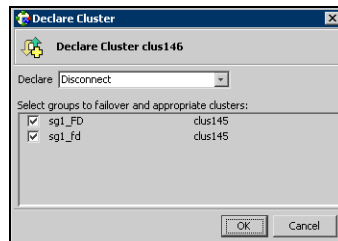


- 2 Click **Take Action**.
- 3 Enter the required information to resolve the alert.

If the alert warns that a local group cannot fail over to any system in the local cluster, you cannot take any action.

If the alert warns that a global group cannot fail over, the action involves bringing the group online on another system in the global cluster environment.

If the alert warns that a global cluster is faulted, the action involves declaring the cluster as a disaster, disconnect, or outage, and determining the service groups to fail over to another cluster.



- 4 Click **OK**.

To delete an alert

- 1** In the **Alert** tab or dialog box, click the alert to delete.
- 2** Click **Delete Alert**.
- 3** Provide the details for this operation:
 - Enter the reason for deleting the alert.
 - Click **OK**.

Administering VCS Simulator

VCS Simulator enables you to view state transitions, experiment with configuration parameters, and predict how service groups will behave during cluster or system faults. Use this tool to create and save configurations in an OFFLINE state.

Through the Java Console, VCS Simulator enables you to configure a simulated cluster panel, bring a system in an unknown state into an online state, simulate power loss for running systems, simulate resource faults, and save the configuration while VCS is offline.

For global clusters, you can simulate the process of generating and clearing cluster faults.

You can run multiple simulated clusters on a system by using different port numbers for each cluster. The Java Console provides the same views and features that are available for online configurations.

See “[Predicting VCS behavior using VCS Simulator](#)” on page 265.

Administering the cluster from the command line

- [About administering VCS from the command line](#)
- [Administering LLT](#)
- [Starting VCS](#)
- [Stopping VCS](#)
- [Logging on to VCS](#)
- [Managing VCS configuration files](#)
- [Managing VCS users from the command line](#)
- [Querying VCS](#)
- [Administering service groups](#)
- [Administering agents](#)
- [Administering resource types](#)
- [Administering resources](#)
- [Administering systems](#)
- [Administering clusters](#)
- [Enabling and disabling Security Services](#)
- [Administering simulated clusters from the command line](#)

About administering VCS from the command line

This chapter describes commonly used VCS commands. For more information about specific commands or their options, see their usage information or the man pages associated with the commands.

Most commands can be entered from any system in the cluster when VCS is running. The command to start VCS is typically invoked at system startup.

See also “[About administering I/O fencing](#)” on page 328.

See also “[VCS command line reference](#)” on page 632.

About the symbols used in the VCS command syntax

[Table 6-2](#) specifies the symbols used in the VCS commands. Do not use these symbols when you run the commands.

Table 6-2 Symbols used in the VCS commands

Symbols	Usage	Example
[]	Used for command options or arguments that are optional.	hasys -freeze [-persistent] [-evacuate] <i>system</i>
	Used to specify that only one of the command options or arguments separated with can be used at a time.	hagetcf [-s -silent]
...	Used to specify that the argument can have several values.	hagrp -modify <i>group attribute</i> value ... [-sys <i>system</i>]
{ }	Used to specify that the command options or arguments enclosed within these braces must be kept together.	haattr -display {cluster group system heartbeat <i>restype</i> } or haclus -modify <i>attribute</i> { <i>key value</i> }
<>	Used in the command help or usage output to specify that these variables must be replaced with the actual values.	haclus -help VCS INFO V-16-1-10601 Usage: haclus -add <cluster> <ip> haclus -delete <cluster>

How VCS identifies the local system

VCS checks the file `$VCS_CONF/conf/sysname`. If this file does not exist, the local system is identified by its node name. To view the system's node name, type:

```
uname -n
```

The entries in this file must correspond to those in the files `/etc/llthosts` and `/etc/llttab`.

About specifying values preceded by a dash (-)

When specifying values in a command-line syntax, you must prefix values beginning with a dash (-) with a percentage sign (%). If a value begins with a percentage sign, you must prefix it with another percentage sign. (The initial percentage sign is stripped by HAD and does not appear in the configuration file.)

About the -modify option

Most configuration changes are made using the `-modify` options of the commands `haclus`, `hagrp`, `hares`, `hasys`, and `hatype`. Specifically, the `-modify` option of these commands changes the attribute values stored in the VCS configuration file. By default, all attributes are global, meaning that the value of the attribute is the same for all systems.

Note: VCS must be in read/write mode before you can change the configuration. For instructions, see [“Setting the configuration to read/write”](#) on page 199.

Encrypting VCS passwords

Use the `vcsencrypt` utility to encrypt passwords when editing the VCS configuration file `main.cf` to add VCS users.

Note: Do not use the `vcsencrypt` utility when entering passwords from the Java and Web consoles.

To encrypt a password

- 1 Run the utility from the command line.

```
vcsencrypt -vcs
```
- 2 The utility prompts you to enter the password twice. Enter the password and press Return.

```
Enter New Password:  
Enter Again:
```

- 3 The utility encrypts the password and displays the encrypted password. Use the displayed password to edit the VCS configuration file `main.cf`.

Encrypting agent passwords

Use the `vcsencrypt` utility to encrypt passwords when editing the VCS configuration file `main.cf` when configuring agents that require user passwords. See also [“Encrypting agent passwords using security keys”](#) on page 182.

Note: Do not use the `vcsencrypt` utility when entering passwords from the Java and Web consoles.

To encrypt an agent password

- 1 Run the utility from the command line.

```
vcsencrypt -agent
```
- 2 The utility prompts you to enter the password twice. Enter the password and press Return.

```
Enter New Password:  
Enter Again:
```
- 3 The utility encrypts the password and displays the encrypted password. Use the displayed password to edit the VCS configuration file `main.cf`.

Encrypting agent passwords using security keys

Use the `vcsencrypt` utility to generate a security key to create a more secure password for agents.

See also [“Encrypting agent passwords”](#) on page 182.

Privilege requirements to generate security keys

By default, only superusers can generate security keys.

You can grant password encryption privileges to group administrators.

See [“Granting password encryption privileges to group administrators”](#) on page 183.

Creating secure agent passwords

Follow these instructions to create secure passwords for agents.

To encrypt agent passwords using security keys

- 1 Make sure you have the privileges required to encrypt passwords.
See [“Privilege requirements to generate security keys”](#) on page 182.
- 2 Generate a security key from a node where VCS is running. You need to do this once.
 - Make the VCS configuration writable.
`haconf -makerw`
 - Run the `vcsecrypt` utility:
`vcsecrypt -gensecinfo`
 - When prompted, enter a password and press Return.
Please enter a passphrase of minimum 8 characters.
Passphrase:
Generating SecInfo...please wait...
SecInfo generated successfully.
SecInfo updated successfully.
 - Save the VCS configuration file.
`haconf -dump`
- 3 Encrypt the agent password with the security key that you generated.
 - On a node where VCS is running, enter the following command:
`vcsecrypt -agent -secinfo`
 - When prompted, enter a password and press Return. The utility prompts you to enter the password twice.
Enter New Password:
Enter Again:
The utility encrypts the password and displays the encrypted password.
- 4 Verify that VCS uses the new encryption mechanism.
 - Verify that the SecInfo cluster attribute is added to the `main.cf` file with the security key as the value of the attribute.
 - Verify that the password that you encrypted resembles the following:
`SAPswd=7c:a7:4d:75:78:86:07:5a:de:9d:7a:9a:8c:6e:53:c6`

Granting password encryption privileges to group administrators

Follow these instructions to grant password encryption privileges to group administrators.

To grant password encryption privileges to group administrators

- ◆ Set the value of the cluster attribute SecInfoLevel to R+A:
`haclus -modify SecInfoLevel R+A`

To restrict password encryption privileges to superusers

- ◆ Set the value of the cluster attribute SecInfoLevel to R:
`haclus -modify SecInfoLevel R`

Changing the security key

Follow these instructions to change the security key.

If you change the security key, make sure you re-encrypt all the passwords that you created with the new security key. Otherwise, agents will fail to decrypt the encrypted password correctly and hence manage to monitor resources correctly.

To change security key

- 1 Save the VCS configuration and make it writeable.
`haconf -makerw`
- 2 Run the following command:
`vcseencrypt -gensecinfo -force`
- 3 Save the VCS configuration and make it read only.
`haconf -dump -makero`

Installing a VCS license

The Veritas product installer prompts you to select one of the following licensing methods:

- Install a license key for the product and features that you want to install. When you purchase a Symantec product, you receive a License Key certificate. The certificate specifies the product keys and the number of product licenses purchased.
- Continue to install without a license key. The installer prompts for the product modes and options that you want to install, and then sets the required product level.
Within 60 days of choosing this option, you must install a valid license key corresponding to the license level entitled or continue with keyless licensing by managing the server or cluster with a management server.

Installing and updating license keys using vxlicinst

- Use the vxlicinst command to install or update a valid product license key for the products you have purchased.
See the vxlicinst(1m) manual page.
You must have root privileges to use this utility. This utility must be run on each system in the cluster; the utility cannot install or update a license on remote nodes.

To install a new license

- ◆ Run the following command on each node in the cluster:

```
cd /opt/VRTS/bin
./vxlicinst -k XXXX-XXXX-XXXX-XXXX-XXXX-XXX
```

To update licensing information in a running cluster

- 1 Install the new license on each node in the cluster using the vxlicinst utility.
- 2 Update system-level licensing information on all nodes in the cluster:

```
hasys -updatelic -all
```

You must run the updatelic command only after you add a license key using the vxlicinst command.

You must run the updatelic command only after you add a license key using the vxlicinst command or after you set the product license level using the vxkeyless command.

You must run the updatelic command only after you add a license key using the vxlicinst command.

You must update licensing information on all nodes before proceeding to the next step.

- 3 Update cluster-level licensing information:

```
haclus -updatelic
```

Setting or changing the product level for keyless licensing

Run the `vxkeyless` command to set the product level for the products you have purchased. This option also requires that you manage the server or cluster with a management server.

See the `vxkeyless(1m)` manual page.

To set or change the product level

- 1 View the current setting for the product level.

```
# vxkeyless [-v] display
```
- 2 View the possible settings for the product level.

```
# vxkeyless displayall
```
- 3 Set the desired product level.

```
# vxkeyless [-q] set prod_levels
```

where `prod_levels` is a comma-separated list of keywords. Use the keywords returned by the `vxkeyless displayall` command.

If you want to remove keyless licensing and enter a key, you must clear the keyless licenses. Use the `NONE` keyword to clear all keys from the system. Note that clearing the keys disables the Veritas products until you install a new key or set a new product level.

To clear the product license level

- 1 View the current setting for the product license level.

```
# vxkeyless [-v] display
```
- 2 If there are keyless licenses installed, remove all keyless licenses:

```
# vxkeyless [-q] set NONE
```

Administering LLT

Displaying the cluster details and LLT version for LLT links

You can use the `ltdump` command to display the LLT version for a specific LLT link. You can also display the cluster ID and node ID details.

See the `ltdump(1M)` manual page for more details.

To display the cluster details and LLT version for LLT links

- ◆ Run the following command to display the details:

```
# /opt/VRTSllt/lltdump -D -f link
```

For example, if `en3` is connected to `galaxy`, then the command displays a list of all cluster IDs and node IDs present on the network link `en3`.

```
galaxy> # /opt/VRTSllt/lltdump -D -f en3
galaxy> galaxy> ltdump : Configuration:
device      : en3
sap         : 0xcafe
promisc sap  : 0
promisc mac  : 0
cidsnoop    : 1
```

```
=== Listening for LLT packets ===
```

cid	nid	vma	vmi
3456	1	5	0
3456	3	5	0
83	0	4	0
27	1	3	7
3456	2	5	0

Adding and removing LLT links

You can use the `lltconfig` command to add or remove LLT links when LLT is running.

See the `lltconfig(1M)` manual page for more details.

To add LLT links

- ◆ Depending on the LLT link type, run the following command to add an LLT link:

- For ether link type:

```
# lltconfig -t devtag -d device [-b ether ] [-s SAP] [-m mtu]
```

- For UDP link type:

```
# lltconfig -t devtag -d device -b udp [-s port] [-m mtu]
-I IPaddr -B bcast
```

```
■ For UDP6 link type:
# lltnconfig -t devtag -d device -b udp6 [-s port] [-m mtu]
-I IPaddr -B mcast
```

Where:

devtag	Tag to identify the link
device	Network device path of the interface For link type ether, the path is followed by a colon (:) and an integer which specifies the unit or PPA used by LLT to attach. For link types udp and udp6, the device is the udp and udp6 device path respectively.
bcast	Broadcast address for the link type udp.
mcast	Multicast address for the link type udp6
IPaddr	IP address for link types udp and udp6
SAP	SAP to bind on the network links for link type ether
port	Port for link types udp and udp6
mtu	Maximum transmission unit to send packets on network links

```
For example:
# lltnconfig -t en3 -d /dev/dlpi/en:3 -s 0xcafe -m 1500
```

To remove an LLT link

```
◆ Run the following command to remove a network link that is configured under LLT:
# lltnconfig -u devtag
```

Configuring aggregated interfaces under LLT

If you want to configure LLT to use aggregated interfaces after installing and configuring VCS, you can use one of the following approaches:

- Edit the /etc/llttab file
This approach requires you to stop LLT. The aggregated interface configuration is persistent across reboots.
- Run the lltnconfig command
This approach lets you configure aggregated interfaces on the fly. However, the changes are not persistent across reboots.

To configure aggregated interfaces under LLT by editing the /etc/llttab file

- 1
- If LLT is running, run the following command to stop LLT:
- 2
- Add the following entry to the /etc/llttab file to configure an aggregated interface.

```
link tag device_name systemid_range link_type sap mtu_size
```

Where:

tag	Tag to identify the link
device_name	Network device path of the aggregated interface The path is followed by a colon (:) and an integer which specifies the unit or PPA used by LLT to attach.
systemid_range	Range of systems for which the command is valid. If the link command is valid for all systems, specify a dash (-).
link_type	The link type must be ether.
sap	SAP to bind on the network links. Default is 0xcafe.
mtu_size	Maximum transmission unit to send packets on network links

- 3
- Restart LLT for the changes to take effect.

To configure aggregated interfaces under LLT using the llttconfig command

- ◆
- When LLT is running, use the following command to configure an aggregated interface:

```
llttconfig -t devtag -d device [-b linktype ] [-s SAP] [-m mtu]
```

Where:

devtag	Tag to identify the link
device	Network device path of the aggregated interface The path is followed by a colon (:) and an integer which specifies the unit or PPA used by LLT to attach.
link_type	The link type must be ether.
sap	SAP to bind on the network links. Default is 0xcafe.
mtu_size	Maximum transmission unit to send packets on network links

See the `lltconfig(1M)` manual page for more details.

You need not reboot after you make this change. However, to make these changes persistent across reboot, you must update the `/etc/llttab` file.

See [“To configure aggregated interfaces under LLT by editing the `/etc/llttab` file”](#) on page 189.

Starting VCS

You can start VCS using one of the following approaches:

- Using the `installvcs -start` command
- Manually start VCS on each node

To start VCS

- 1 To start VCS using the `installvcs` program, perform the following steps on any node in the cluster:
 - Log in as root user.
 - Run the following command:


```
# /opt/VRTS/install/installvcs -start
```
- 2 To start VCS manually, run the following commands on each node in the cluster:
 - Log in as root user.
 - Start LLT and GAB. Start I/O fencing if you have configured it. Skip this step if you want to start VCS on a single-node cluster.

```
LLT                # /etc/init.d/llt.rc start
```

```
GAB                # /etc/init.d/gab.rc start
```

```
I/O fencing        # /etc/init.d/vxfen.rc start
```

- Start the VCS engine.


```
# hastart
```

To start VCS in a single-node cluster, run the following command:

```
# hastart -onenode
```

See [“Starting the VCS engine \(HAD\) and related processes”](#) on page 191.
- 3 Verify that the VCS cluster is up and running.


```
# gabconfig -a
```

Make sure that port a and port h memberships exist in the output for all nodes in the cluster. If you configured I/O fencing, port b membership must also exist.

Starting the VCS engine (HAD) and related processes

The command to start VCS is invoked from the following file:

`/etc/rc.d/rc2.d/S99.vcs`

When VCS is started, it checks the state of its local configuration file and registers with GAB for cluster membership. If the local configuration is valid, and if no other system is running VCS, it builds its state from the local configuration file and enters the `RUNNING` state.

If the configuration on all nodes is invalid, the VCS engine waits for manual intervention, or for VCS to be started on a system that has a valid configuration.

See “[Cluster and system states](#)” on page 635.

To start the VCS engine

- ◆ Run the following command:

`hastart`

To start the VCS engine when all systems are in the `ADMIN_WAIT` state

- ◆ Run the following command from any system in the cluster to force VCS to use the configuration file from the system specified by the variable *system*:

`hasys -force system`

To start VCS on a single-node cluster

- ◆ Type the following command to start an instance of VCS that does not require the GAB and LLT packages. Do not use this command on a multisystem cluster.

`hastart -onenode`

Stopping VCS

You can stop VCS using one of the following approaches:

- Using the `installvcs -stop` command
- Manually stop VCS on each node

To stop VCS

- 1 To stop VCS using the `installvcs` program, perform the following steps on any node in the cluster:
 - Log in as root user.
 - Run the following command:


```
# /opt/VRTS/install/installvcs -stop
```
- 2 To stop VCS manually, run the following commands on each node in the cluster:
 - Log in as root user.
 - Take the VCS service groups offline and verify that the groups are offline.


```
# hagrps -offline service_group -sys system
# hagrps -state service_group
```
 - Stop the VCS engine.


```
# hastop -local
```

 See [“Stopping the VCS engine and related processes”](#) on page 192.
 - Verify that the VCS engine port `h` is closed.


```
# gabconfig -a
```
 - Stop I/O fencing if you have configured it. Stop GAB and then LLT.

```
I/O fencing      # /etc/init.d/vxfen.rc stop
```

```
GAB              # /etc/init.d/gab.rc stop
                 stop
```

```
LLT              # /etc/init.d/llt.rc stop
                 stop
```

Stopping the VCS engine and related processes

The `hastop` command stops HAD and related processes. You can customize the behavior of the `hastop` command by configuring the `EngineShutdown` attribute for the cluster.

See “[Controlling the hstop behavior using the EngineShutdown attribute](#)” on page 194.

The hstop command includes the following options:

```
hstop -all [-force]
hstop [-help]
hstop -local [-force | -evacuate | -noautodisable]
hstop -sys system ... [-force | -evacuate | -noautodisable]
```

Option	Description
-all	Stops HAD on all systems in the cluster and takes all service groups offline.
-help	Displays command usage.
-local	Stops HAD on the system on which you typed the command
-force	Allows HAD to be stopped without taking service groups offline on the system. The value of the EngineShutdown attribute does not influence the behavior of the -force option.
-evacuate	When combined with -local or -sys, migrates the system’s active service groups to another system in the cluster, before the system is stopped.
-noautodisable	Ensures that service groups that can run on the node where the hstop command was issued are not autodisabled. This option can be used with -evacuate but not with -force.
-sys	Stops HAD on the specified system.

Stopping VCS without -force option

When VCS is stopped on a system without using the -force option, it enters the LEAVING state, and waits for all groups to go offline on the system. Use the output of the command `hasys -display system` to verify that the values of the SysState and the OnGrpCnt attributes are non-zero. VCS continues to wait for the service groups to go offline before it shuts down.

See “[Troubleshooting resources](#)” on page 597.

Stopping VCS with options other than -force

When VCS is stopped by options other than -force on a system with online service groups, the groups running on the system are taken offline and remain offline. This is indicated by VCS setting the attribute IntentOnline to 0. Using

the option `-force` enables service groups to continue running while the VCS engine (HAD) is brought down and restarted. The value of the `IntentOnline` attribute remains unchanged after the VCS engine restarts.

Controlling the hastop behavior using the EngineShutdown attribute

Use the `EngineShutdown` attribute to define VCS behavior when a user runs the `hastop` command.

Note: VCS does not consider this attribute when the `hastop` is issued with the following options: `-force` or `-local -evacuate -noautodisable`.

Configure one of the following values for the attribute depending on the desired functionality for the `hastop` command:

EngineShutdown Value	Description
Enable	Process all <code>hastop</code> commands. This is the default behavior.
Disable	Reject all <code>hastop</code> commands.
DisableClusStop	Do not process the <code>hastop -all</code> command; process all other <code>hastop</code> commands.
PromptClusStop	Prompt for user confirmation before running the <code>hastop -all</code> command; process all other <code>hastop</code> commands.
PromptLocal	Prompt for user confirmation before running the <code>hastop -local</code> command; process all other <code>hastop</code> commands.
PromptAlways	Prompt for user confirmation before running any <code>hastop</code> command.

Additional considerations for stopping VCS

- If using the command `reboot`, behavior is controlled by the `ShutdownTimeOut` parameter. After HAD exits, if GAB exits within the time designated in the `ShutdownTimeout` attribute, the remaining systems recognize this as a reboot and fail over service groups from the departed system. For systems running several applications, consider increasing the value in the `ShutdownTimeout` attribute.
- Stopping VCS on a system autodisables each service group that includes the system in their `SystemList` attribute. (This does not apply to systems that are powered off.)

- If you use the `-evacuate` option, evacuation occurs before VCS is brought down.

Logging on to VCS

VCS prompts user name and password information when non-root users run *haxxx* commands. You can use the *halogin* command to save the authentication information so that you do not have to enter your credentials every time you run a VCS command. Note that you might need specific privileges to run VCS commands.

When you run the *halogin* command, VCS stores encrypted authentication information in the user's home directory. For clusters running in secure mode, the command also sets up a trust relationship and retrieves a certificate from an authentication broker.

If you run the command for different hosts, VCS stores authentication information for each host. After you run the command, VCS stores the information until you end the session.

For clusters that run in secure mode, you can also generate credentials for VCS to store the information for eight years and thus configure VCS to not prompt for passwords when you run VCS commands as non-root users.

See [“Running HA commands as non-root users on clusters in secure mode”](#) on page 197.

Root users need not run *halogin* when running VCS commands from the local host.

To log on to a cluster running in secure mode

- 1 Set the following environment variables:
 - **VCS_DOMAIN**—Name of the Security domain to which the user belongs.
 - **VCS_DOMAINTYPE**—Type of VxSS domain: *unixpwd*, *nt*, *nis*, *nisplus*, or *vx*.
- 2 Define the node on which the VCS commands will be run. Set the **VCS_HOST** environment variable to the name of the node. To run commands in a remote cluster, you set the variable to the virtual IP address configured in the ClusterService group.
- 3 Log on to VCS:
halogin vcsusername password

To log on to a cluster not running in secure mode

- 1 Define the node on which the VCS commands will be run. Set the **VCS_HOST** environment variable to the name of the on which to run commands. You can set the variable to the virtual IP address configured in the ClusterService group.

- 2 Log on to VCS:

```
halogin vcsusername password
```

To end a session for a host

- ◆ Run the following command:

```
halogin -endsession hostname
```

To end all sessions

- ◆ Run the following command:

```
halogin -endallsessions
```

VCS prompts you for credentials every time you run a VCS command.

Running HA commands as non-root users on clusters in secure mode

Perform the following procedure to configure VCS to not prompt for passwords when a non-root user runs HA commands on a cluster which runs in secure mode.

To run HA commands as non-root users on clusters in secure mode

- 1 Set the VCS_DOMAIN and VCS_DOMAINTYPE environment variables as follows:

- VCS_DOMAIN=nodomain

- VCS_DOMAINTYPE=localhost

You can set the VCS_DOMAIN value to any dummy value.

- 2 Run any HA command. This step generates credentials for VCS to store the authentication information in the user's home directory. For example, run the following command:

```
# hasys -state
```

Managing VCS configuration files

This section describes how to verify, back up, and restore VCS configuration files.

See [“About the main.cf file”](#) on page 53.

See [“The types.cf file”](#) on page 55.

About the hacf utility

The hacf utility translates the VCS configuration language into a syntax that can be read by the VCS engine. Specifically, hacf translates the contents of the main configuration file, main.cf, into commands for the VCS server.

The hacf utility verifies the configuration before loading it into VCS. The configuration is not loaded under the following conditions:

- If main.cf or include files are missing.
- If syntax errors appear in the .cf files.
- If the configuration file is invalid.

See [“Setting the configuration to read/write”](#) on page 199.

About multiple versions of .cf files

When hacf creates a .cf file, it does *not* overwrite existing .cf files. A copy of the file remains in the directory, and its name includes a suffix of the date and time it was created, such as main.cf.03Dec2001.175904. In addition, the previous version of any .cf file is saved with the suffix .previous; for example, main.cf.previous.

Verifying a configuration

Use hacf to verify (check syntax of) the main.cf and the type definition file, types.cf. VCS does not execute if hacf detects errors in the configuration.

To verify a configuration

- ◆ Run the following command:

```
hacf -verify config_directory
```

The variable *config_directory* refers to directories containing a main.cf file and any .cf files included in main.cf.

No error message and a return value of zero indicates that the syntax is legal.

Scheduling automatic backups for VCS configuration files

Configure the BackupInterval attribute to instruct VCS to create a back up of the configuration periodically. VCS backs up the main.cf and types.cf files as main.cf.autobackup and types.cf.autobackup respectively.

To start periodic backups of VCS configuration files

- ◆ Set the cluster-level attribute BackupInterval to a non-zero value. For example, to back up the configuration every 5 minutes, set BackupInterval to 5.

Saving a configuration

When you save a configuration, VCS renames the file main.cf.autobackup to main.cf. VCS also save your running configuration to the file main.cf.autobackup.

If have not configured the BackupInterval attribute, VCS saves the running configuration.

See “[Scheduling automatic backups for VCS configuration files](#)” on page 199.

To save a configuration

- ◆ Run the following command
haconf -dump -makero
The option -makero sets the configuration to read-only.

Setting the configuration to read/write

To set the mode to read/write

- ◆ Type the following command:
haconf -makerw

Formatting configuration files

When you manually edit VCS configuration files (for example, the main.cf or types.cf file) you can potentially create formatting issues that prevent the files from being parsed correctly.

To display the configuration files in the correct format

- ◆ Run the following commands to display the configuration files in the correct format:

```
# haconf -cftocmd config
# haconf -cmdtoci config
```

Taking snapshots of VCS configuration files

Use the `hasnap` command to take snapshots of VCS configuration files on each node in the cluster. You can also restore the configuration from a snapshot.

The `hasnap` command requires JRE 1.5. Make sure that you meet at least one of the following requirements for the `hasnap` command to run successfully:

- `JAVA_HOME` is specified and it points to Java v1.5+ installation
- `/usr/bin/java` is at least v1.5
- `$PATH` has java and is at least v1.5

The command includes the following options; each option is described in detail in the following sections:

<code>hasnap -backup</code>	Backs up files in a snapshot format.
<code>hasnap -restore</code>	Restores a previously created snapshot.
<code>hasnap -display</code>	Displays details of previously created snapshots.
<code>hasnap -sdiff</code>	Displays files that were changed on the local system after a specific snapshot was created.
<code>hasnap -fdiff</code>	Displays the differences between a file in the cluster and its copy stored in a snapshot.
<code>hasnap -export</code>	Exports a snapshot from the local, predefined directory to the specified file.
<code>hasnap -include</code>	Configures the list of files or directories to be included in new snapshots, in addition to those included automatically by the <code>-backup</code> command.
<code>hasnap -exclude</code>	Configures the list of files or directories to be excluded from new snapshots when backing up the configuration using the <code>-backup</code> command.
<code>hasnap -delete</code>	Deletes snapshots from the predefined local directory on each node.

Note: With the exception of the `-include`, `-exclude`, and the `-delete` options, all options can be combined with the `-f` option. This option indicates that all files be backed up to or restored from the specified single file instead of a local, predefined directory on each node. This option is useful when you want to store the configuration data to an alternate location that is periodically backed up using backup software like Veritas Net Backup.

Backing up configuration files

The `hasnap -backup` command backs up files in a snapshot format. A snapshot is a collection of VCS configuration files backed up at a particular point in time, typically before making changes to the existing configuration. A snapshot also contains information such as the snapshot name, description, creation time, and file permissions.

The command backs up a predefined list of VCS configuration files as well as a user-defined list. The predefined list includes all the *.cf files, custom agents, LLT and GAB configuration files, triggers, custom heartbeats, and action scripts. See the `-include` and `-exclude` commands to construct a user-defined list.

To back up VCS configuration files

- ◆ Run the following command

```
hasnap -backup [-f filename] [-n] [-m description]
```

Use the `-n` option to run the command in the non-interactive mode.

Use the `-m` option to specifies a description of the snapshot.

Examples

The following command creates a backup of the configuration in the non-interactive mode and adds Test Backup as the backup description.

```
hasnap -backup -n -m "Test Backup"
```

The following command creates a backup of the configuration files and saves it as `/tmp/backup-2-2-2003` on the node where the command was run.

```
hasnap -backup -f /tmp/backup-2-2-2003
```

Restoring VCS configuration files

The `hasnap -restore` command restores configuration files from a previously created snapshot.

To restore VCS configuration files

- ◆ Run the following command:

```
hasnap -restore [-f filename] [-n] [-s snapid]
```

`-n` Run the command in the non-interactive mode.

`-s` option to specifies the ID of the snapshot to be restored.

If you do not specify a snapshot ID, the command lists the snapshots that are available for restoration.

Examples

The following command restores the snapshot vcs-20030101-22232 in the non-interactive mode.

```
hasnap -restore -n -s vcs-20030101-22232
```

The following command restores the snapshot stored in the file /tmp/backup-2-2-2003.

```
hasnap -restore -f /tmp/backup-2-2-2003
```

Viewing snapshots of configuration files

Use the `hasnap -display` command to view details of previously created snapshots.

To view snapshots of configuration files

```
hasnap -display [-f filename] [-list|-s snapid] [-m] [-l] [-t]
```

<code>-list</code>	Displays the list of snapshots in the repository.
<code>-s</code>	Specifies the snapshot ID.
<code>-m</code>	Displays snapshot description.
<code>-l</code>	Lists files in the snapshot
<code>-t</code>	Displays the snapshot timestamp

If no options are specified, the command displays all information about the latest snapshot.

Examples

The following command lists all snapshots.

```
hasnap -display -list
```

The following command displays the description and the time of creation of the specified snapshot.

```
hasnap -display -s vcs-20030101-2232 -m -t
```

The following command displays the description, the timestamp, and the list of all files in the snapshot file /tmp/backup-2-2-2003

```
hasnap -display -f /tmp/backup-2-2-2003
```

Viewing files changed after a snapshot

Use the `hasnap -sdiff` command to display files that were changed on the local system after a specific snapshot was created.

To view files that changed after a snapshot

- ◆ Run the following command:

```
hasnap -sdiff [-f filename] [-s snapid] [-sys hostname]
```

-s	Identifies the snapshot ID of the comparison snapshot.
-sys	Indicates the host on which the snapshot is to be compared.

If you do not specify any options, the command uses the latest snapshot to compare the files on each node in the cluster.

Examples

The following command displays the differences between the current configuration and the snapshot vcs-20030101-22232.

```
hasnap -sdiff -s vcs-20030101-22232
```

The following command displays the difference between the configuration on system host1 and the snapshot stored in the file /tmp/backup-2-2-2003.

```
hasnap -sdiff -f /tmp/backup-2-2-2003 -sys host1
```

Comparing a file with its snapshot copy

Use the `hasnap -fdiff` to displays differences between a file on the cluster and its copy stored in a previously created snapshot.

To compare a file with its snapshot copy

- ◆ Run the following command:

```
hasnap -fdiff [-f filename] [-s snapid] [-sys hostname] file
```

-s	Specifies the ID of the snapshot.
-sys	Specifies the host on which the snapshot is to be compared.
-file	The file to compare.

If you do not specify any options, the command uses the latest snapshot to compare the file on each node in the cluster.

Examples

The following command displays the differences between the files /etc/VRTSvcs/conf/config/main.cf on host1 and its version in the last snapshot.

```
hasnap -fdiff -sys host1 /etc/VRTSvcs/conf/config/main.cf
```

The following command displays the differences between the files /etc/llttab on each node in the cluster and the version stored in the snapshot contained in the file /var/backup-2-2-2003.

```
hasnap -fdiff -f /tmp/backup-2-2-2003 /etc/llttab
```

Exporting snapshots

Use the `hasnap -export` command to export a snapshot from the local, predefined directory on each node in the cluster to a specified file. This option is useful when you want to store a previously created snapshot to an alternate location that is periodically backed up using backup software like Veritas NetBackup.

To export a snapshot

- ◆ Run the following command:

```
hasnap -export -f filename [-s snapid]
```

 -s Specifies the ID of the snapshot.

 -f Specifies the file.

If you do not specify a snapshot ID, the command exports the latest snapshot to the specified file.

Example

The following command exports data from snapshot vcs-20030101-22232 from each node in the cluster to the file /tmp/backup-2-2-2003 on the current node.

```
hasnap -export -f /tmp/backup-2-2-2003 -s vcs-20030101-22232
```

Adding and removing files for snapshots

Use the `hasnap -include` command to configure the list of files or directories to be included in new snapshots. This list is in addition to the files included by the `-backup` command.

See “[Backing up configuration files](#)” on page 201.

To add or remove files for a snapshots

- ◆ Run the following command:

```
hasnap -include -add|-del|-list [-sys hostname]  
files|directories
```

<code>-add</code>	Adds the specified files or directories to the include file list.
<code>-del</code>	Removes the specified files or directories from the include file list.
<code>files/ directories</code>	Files or directories to be added or removed.

Examples

The following command displays the list of files or directories to be included in new snapshots on each node of the cluster.

```
hasnap -include -list
```

Excluding files from snapshots

Use the `hasnap -exclude` command to configure the list of files or directories that should not be included in new snapshots.

To exclude files from snapshots

- ◆ Run the following command:

```
hasnap -exclude -add|-del|-list [-sys hostname]  
files|directories
```

<code>-add</code>	Adds the specified files or directories to the exclude file list.
<code>-del</code>	Removes the specified files or directories from the exclude file list.
<code>files/ directories</code>	Files or directories to be added or removed.

Examples

The following command displays the exclude file list on each node in the cluster.

```
hasnap -exclude -list
```

The following command adds the file `/etc/VRTSvcs/conf/config/temp.cf` to the exclude file list on `host1`, which results in this file being excluded from the snapshot the next time the `hasnap -backup` command is run.

```
hasnap -exclude -add -sys host1  
/etc/VRTSvcs/conf/config/temp.cf
```

The following command removes the file `/etc/VRTSvcs/conf/config/temp.cf` from the exclude list on `host1`.

```
hasnap -exclude -del -sys host1  
/etc/VRTSvcs/conf/config/temp.cf
```

Deleting snapshots

Use the `hasnap -delete` command to delete snapshots from the predefined local directory on each node.

To delete a snapshot

- ◆ Run the following command:

```
hasnap -delete [-s snapshotid]
```

`-s` Specifies the ID of the snapshot to be deleted.

If you do not specify the snapshot ID, the command lists the snapshots that can be deleted.

Example

The following command deletes snapshot `vcs-20030101-22232` from the cluster.

```
hasnap -delete -s vcs-20030101-22232
```

Managing VCS users from the command line

You can add, modify, and delete users on any system in the cluster, provided you have the privileges to do so.

If VCS is running in secure mode, specify fully-qualified user names, in the format `username@domain`. You cannot assign or change passwords for users when VCS is running in secure mode.

The commands to add, modify, and delete a user must be executed only as root or administrator and only if the VCS configuration is in read/write mode.

See “[Setting the configuration to read/write](#)” on page 199.

Note: You must add users to the VCS configuration to monitor and administer VCS from the graphical user interface Cluster Manager.

Adding a user

Users in the category Cluster Guest cannot add users.

To add a user

- 1 Set the configuration to read/write mode:
`haconf -makerw`
- 2 Add the user:
`hauser -add user [-priv <Administrator|Operator> [-group service_groups]]`
- 3 Enter a password when prompted.
- 4 Reset the configuration to read-only:
`haconf -dump -makero`

To add a user with cluster administrator access

- ◆ Type the following command:
`hauser -add user -priv Administrator`

To add a user with cluster operator access

- ◆ Type the following command:
`hauser -add user -priv Operator`

To add a user with group administrator access

- ◆ Type the following command:
`hauser -add user -priv Administrator -group service_groups`

To add a user with group operator access

- ◆ Type the following command:
hauser -add user -priv Operator -group service_groups

Assigning and removing user privileges

To assign privileges to an administrator or operator

- ◆ Type the following command:
**hauser -addpriv user Administrator|Operator
[-group service_groups]**

To remove privileges from an administrator or operator

- ◆ Type the following command:
**hauser -delpriv user Administrator|Operator
[-group service_groups]**

To assign privileges to an OS user group

- ◆ Type the following command:
**hauser -addpriv usergroup AdministratorGroup|OperatorGroup
[-group service_groups]**

To remove privileges from an OS user group

- ◆ Type the following command:
**hauser -delpriv usergroup AdministratorGroup|OperatorGroup
[-group service_groups]**

Modifying a user

Users in the category Cluster Guest cannot modify users. You cannot modify a VCS user in clusters that run in secure mode.

To modify a user

- 1 Set the configuration to read/write mode:
haconf -makerw
- 2 Modify the user:
hauser -update user
- 3 Enter a new password when prompted.
- 4 Reset the configuration to read-only:
haconf -dump -makero

Deleting a user

You can delete a user from the VCS configuration.

To delete a user

- 1 Set the configuration to read/write mode:
haconf -makerw
- 2 For users with Administrator and Operator access, remove their privileges:
**hauser -delpriv user Administrator|Operator [-group
service_groups]**
- 3 Delete the user from the list of registered users:
hauser -delete user
- 4 Reset the configuration to read-only:
haconf -dump -makero

Displaying a user

Display a list of users and their privileges.

To display a list of users

- ◆ Type the following command:
hauser -list

To display the privileges of all users

- ◆ Type the following command:
hauser -display

To display the privileges of a specific user

- ◆ Type the following command:
hauser -display user

Querying VCS

VCS enables you to query various cluster objects, including resources, service groups, systems, resource types, agents, and clusters. You may enter query commands from any system in the cluster. Commands to display information on the VCS configuration or system states can be executed by all users: you do not need root privileges.

Querying service groups

To display the state of a service group on a system

- ◆ Type the following command:
`hagrp -state [service_group] [-sys system]`

To display the resources for a service group

- ◆ Type the following command:
`hagrp -resources service_group`

To display a list of a service group's dependencies

- ◆ Type the following command:
`hagrp -dep [service_group]`

To display a service group on a system

- ◆ Type the following command:
`hagrp -display [service_group] [-sys system]`
 If *service_group* is not specified, information regarding all service groups is displayed.

To display attributes of a system

- ◆ Type the following command:
`hagrp -display [service_group] [-attribute attribute]
 [-sys system]`

Note that system names are case-sensitive.

Querying resources

To display a resource's dependencies

- ◆ Type the following command:

```
hares -dep [resource]
```

To display information about a resource

- ◆ Type the following command:

```
hares -display [resource]
```

If *resource* is not specified, information regarding all resources is displayed.

To confirm an attribute's values are the same on all systems

- ◆ Type the following command:

```
hares -global resource attribute value ... | key... |  
{key value}...
```

To display resources of a service group

- ◆ Type the following command:

```
hares -display -group service_group
```

To display resources of a resource type

- ◆ Type the following command:

```
hares -display -type resource_type
```

To display attributes of a system

- ◆ Type the following command:

```
hares -display -sys system
```

Querying resource types

To display all resource types

- ◆ Type the following command:
`hatype -list`

To display resources of a particular resource type

- ◆ Type the following command:
`hatype -resources resource_type`

To display information about a resource type

- ◆ Type the following command:
`hatype -display resource_type`
If *resource_type* is not specified, information regarding all types is displayed.

Querying agents

To display the run-time status of an agent'

- ◆ Type the following command:
`haagent -display [agent]`
If *agent* is not specified, information regarding all agents is displayed.

Run-Time Status	Definition
Faults	Indicates the number of agent faults and the time the faults began.
Messages	Displays various messages regarding agent status.
Running	Indicates the agent is operating.
Started	Indicates the file is executed by the VCS engine (HAD).

Querying systems

To display a list of systems in the cluster

- ◆ Type the following command:
hasys -list

To display information about each system

- ◆ Type the following command:
hasys -display [system]

Querying clusters

To display the value of a specific cluster attribute

- ◆ Type the following command:
haclus -value attribute

To display information about the cluster

- ◆ Type the following command:
haclus -display

Querying status

To display the status of all service groups in the cluster, including resources

- ◆ Type the following command:
hastatus

To display the status of a particular service group, including its resources

- ◆ Type the following command:
hastatus [-sound] [-time] -group service_group [-group service_group]...

If you do not specify a service group, the status of all service groups is displayed. The **-sound** option enables a bell to ring each time a resource faults.

The **-time** option prints the system time at which the status was received.

To display the status of service groups and resources on specific systems

- ◆ Type the following command:
hastatus [-sound] [-time] -sys system_name [-sys system_name]...

To display the status of specific resources

- ◆ Type the following command:

```
hastatus [-sound] [-time] -resource resource_name [-resource  

resource_name]...
```

To display the status of cluster faults, including faulted service groups, resources, systems, links, and agents

- ◆ Type the following command:

```
hastatus -summary
```

Note: Unless executed with the `-summary` options, the `hastatus` command continues to produce output of online state transitions until you interrupt it with the command `CTRL+C`.

Querying log data files (LDFs)

Log data files (LDFs) contain data regarding messages written to a corresponding English language file. Typically, for each English file there is a corresponding LDF.

To display the hamsg usage list

- ◆ Type the following command:

```
hamsg -help
```

To display the list of LDFs available on the current system

- ◆ Type the following command:

```
hamsg -list
```

To display general LDF data

- ◆ Type the following command:

```
hamsg -info [-path path_name] LDF
```

The option `-path` specifies where `hamsg` looks for the specified LDF. If not specified, `hamsg` looks for files in the default directory:

```
/var/VRTSvcs/ldf
```

To display specific LDF data

- ◆ Type the following command:

```
hamsg [-any] [-sev C|E|W|N|I] [-otype VCS|RES|GRP|SYS|AGT]  

[-oname object_name] [-msgid message_ID] [-path  

path_name] [-lang language] LDF
```

-any	Specifies hamsg return messages matching any of the specified query options.
-sev	Specifies hamsg return messages matching the specified message severity Critical, Error, Warning, Notice, or Information.
-otype	Specifies hamsg return messages matching the specified object type <ul style="list-style-type: none">■ VCS = general VCS messages■ RES = resource■ GRP = service group■ SYS = system■ AGT = agent
-oname	Specifies hamsg return messages matching the specified object name.
-msgid	Specifies hamsg return messages matching the specified message ID.
-path	Specifies where hamsg looks for the specified LDF. If not specified, hamsg looks for files in the default directory /var/VRTSvc/ldf.
-lang	Specifies the language in which to display messages. For example, the value en specifies English and "ja" specifies Japanese.

Using conditional statements to query VCS objects

Some query commands include an option for conditional statements.

Conditional statements take three forms:

Attribute=Value (the attribute equals the value)

Attribute!=Value (the attribute does not equal the value)

Attribute=~Value (the value is the prefix of the attribute, for example a query for the state of a resource = ~FAULTED returns all resources whose state begins with FAULTED.)

Multiple conditional statements can be used and imply AND logic.

You can only query attribute-value pairs displayed in the output of the command `hagrp -display`.

See “[Querying service groups](#)” on page 210.

To display the list of service groups whose values match a conditional statement

- ◆ Type the following command:

```
hagrp -list [conditional_statement]
```

If no conditional statement is specified, all service groups in the cluster are listed.

To display a list of resources whose values match a conditional statement

- ◆ Type the following command:

```
hares -list [conditional_statement]
```

If no conditional statement is specified, all resources in the cluster are listed.

To display a list of agents whose values match a conditional statement

- ◆ Type the following command:

```
haagent -list [conditional_statement]
```

If no conditional statement is specified, all agents in the cluster are listed.

Administering service groups

This section describes how to add, delete, and modify service groups. It also describes how to perform service group operations from the command line.

Adding and deleting service groups

To add a service group to your cluster

```
hagrp -add service_group
```

The variable *service_group* must be unique among all service groups defined in the cluster.

This command initializes a service group that is ready to contain various resources. To employ the group properly, you must populate its SystemList attribute to define the systems on which the group may be brought online and taken offline. (A system list is an association of names and integers that represent priority values.)

To delete a service group

- ◆ Type the following command:

```
hagrp -delete service_group
```

Note that you cannot delete a service group until all of its resources are deleted.

Modifying service group attributes

To modify a service group attribute

- ◆ Type the following command:

```
hagrp -modify service_group attribute value [-sys system]
```

The variable *value* represents:

```
system_name1 priority system_name2 priority2
```

If the attribute being modified has local scope, you must specify the system on which to modify the attribute, except when modifying the attribute on the system from which you run the command.

For example, to populate the system list of service group groupx with Systems A and B, type:

```
hagrp -modify groupx SystemList -add SystemA 1 SystemB 2
```

Similarly, to populate the AutoStartList attribute of a service group, type:

```
hagrp -modify groupx AutoStartList SystemA SystemB
```

You may also define a service group as parallel. To set the Parallel attribute to 1, type the following command. (Note that the default for this attribute is 0, which designates the service group as a failover group.):

```
hagrp -modify groupx Parallel 1
```

This attribute cannot be modified if resources have already been added to the service group.

You can modify the attributes SystemList, AutoStartList, and Parallel only by using the command `hagrp -modify`. You cannot modify attributes created by the system, such as the state of the service group.

About modifying the SystemList attribute

When using the `hagrp -modify` command to change a service group's existing system list, you can use the options `-modify`, `-add`, `-update`, `-delete`, or `-delete -keys`.

For example, suppose you originally defined the SystemList of service group `groupx` as `SystemA` and `SystemB`. Then after the cluster was brought up you added a new system to the list:

```
hagrp -modify groupx SystemList -add SystemC 3
```

You must take the service group offline on the system being modified.

When you add a system to a service group's system list, the system must have been previously added to the cluster. When using the command line, you can use the `hasys -add` command.

When you delete a system from a service group's system list, the service group must not be online on the system to be deleted.

If you attempt to change a service group's existing system list using `hagrp -modify` without other options (such as `-add` or `-update`) the command fails.

Bringing service groups online

To bring a service group online

- ◆ Type the following command:

```
hagrp -online service_group -sys system
```

To start a service group on a system and bring online only the resources already online on another system

- ◆ Type the following command:

```
hagrp -online service_group -sys system -checkpartial  
other_system
```

If the service group does not have resources online on the other system, the service group is brought online on the original system and the `checkpartial` option is ignored.

Note that the `checkpartial` option is used by the Preonline trigger during failover. When a service group configured with `Preonline = 1` fails over to another system (system 2), the only resources brought online on system 2 are those that were previously online on system 1 prior to failover.

Taking service groups offline

To take a service group offline

- ◆ Type the following command:

```
hagrp -offline service_group -sys system
```

To take a service group offline only if all resources are probed on the system

- ◆ Type the following command:

```
hagrp -offline [-ifprobed] service_group -sys system
```

Switching service groups

The process of switching a service group involves taking it offline on its current system and bringing it online on another system

To switch a service group from one system to another

- ◆ Type the following command:

```
hagrp -switch service_group -to system
```

A service group can be switched only if it is fully or partially online. The `-switch` option is not supported for switching hybrid service groups across system zones.

Switch parallel global groups across cluster using the following command:

```
hagrp -switch service_group -any -clus remote_cluster
```

VCS brings the parallel service group online on all possible nodes in the remote cluster.

Freezing and unfreezing service groups

Freeze a service group to prevent it from failing over to another system. This freezing process stops all online and offline procedures on the service group.

Unfreeze a frozen service group to perform online or offline operations on the service group.

To freeze a service group (disable online, offline, and failover operations)

- ◆ Type the following command:

```
hagrp -freeze service_group [-persistent]
```

The option `-persistent` enables the freeze to be remembered when the cluster is rebooted.

To unfreeze a service group (reenable online, offline, and failover operations)

- ◆ Type the following command:

```
hagrp -unfreeze service_group [-persistent]
```

Enabling and disabling service groups

Enable a service group before bringing it online. A service group that was manually disabled during a maintenance procedure on a system may need to be brought online after the procedure is completed.

Disable a service group to prevent it from coming online. This process temporarily stops VCS from monitoring a service group on a system undergoing maintenance operations

To enable a service group

- ◆ Type the following command:

```
hagrp -enable service_group [-sys system]
```

A group can be brought online only if it is enabled.

To disable a service group

- ◆ Type the following command:

```
hagrp -disable service_group [-sys system]
```

A group cannot be brought online or switched if it is disabled.

To enable all resources in a service group

- ◆ Type the following command:
hagrp -enableresources *service_group*

To disable all resources in a service group

- ◆ Type the following command:
hagrp -disableresources *service_group*
Agents do not monitor group resources if resources are disabled.

Clearing faulted resources in a service group

Clear a resource to remove a fault and make the resource available to go online.

To clear faulted, non-persistent resources in a service group

- ◆ Type the following command:
hagrp -clear *service_group* [-sys *system*]
Clearing a resource initiates the online process previously blocked while waiting for the resource to become clear.
 - If *system* is specified, all faulted, non-persistent resources are cleared from that system only.
 - If *system* is not specified, the service group is cleared on all systems in the group's SystemList in which at least one non-persistent resource has faulted.

To clear resources in ADMIN_WAIT state in a service group

- ◆ Type the following command:
hagrp -clearadminwait [-fault] *service_group* -sys *system*
See [“Changing agent file paths and binaries”](#) on page 386.

Flushing service groups

As a service group is brought online or taken offline, the resources within the group are brought online and taken offline. If the online or offline operation hangs on a particular resource, flush the service group to halt the operation on the resources waiting to go online or offline. Flushing a service group typically leaves the cluster in a partial state. After completing this process, resolve the issue with the particular resource (if necessary) and proceed with starting or stopping the service group.

To flush a service group on a system

- ◆ Type the following command:

```
hagrp -flush group -sys system [-clus cluster | -localclus]
```

To flush all service groups on a system

- 1 Save the following script as `haflush` at the location `/opt/VRTSvcs/bin/`

```
#!/bin/ksh
PATH=/opt/VRTSvcs/bin:$PATH; export PATH
if [ $# -ne 1 ]; then
    echo "usage: $0 <system name>"
    exit 1
fi

hagrp -list |
while read grp sys junk
do
    locsys="${sys##*:}"
    case "$locsys" in
        "$1")
            hagrp -flush "$grp" -sys "$locsys"
            ;;
        *)
            ;;
    esac
done
```

- 2 Run the script.

```
haflush systemname
```

Linking and unlinking service groups

Link service groups to create a dependency between them.

See “[About service group dependencies](#)” on page 410.

To link service groups

- ◆ Type the following command

```
hagrp -link parent_group child_group gd_category  
                                gd_location gd_type
```

parent_group Name of the parent group

child_group Name of the child group

gd_category Category of group dependency (online/offline).

gd_location The scope of dependency (local/global/remote).

gd_type Type of group dependency (soft/firm/hard). Default is firm.

To unlink service groups

- ◆ Type the following command:

```
hagrp -unlink parent_group child_group
```

Administering agents

Under normal conditions, VCS agents are started and stopped automatically.

To start an agent

- ◆ Run the following command:

```
haagent -start agent -sys system
```

To stop an agent

- ◆ Run the following command:

```
haagent -stop agent [-force] -sys system
```

The `-force` option stops the agent even if the resources for the agent are online. Use the `-force` option when you want to upgrade an agent without taking its resources offline.

Administering resources

About adding resources

When you add a resource, all non-static attributes of the resource's type, plus their default values, are copied to the new resource.

Three attributes are also created by the system and added to the resource:

- **Critical** (default = 1). If the resource or any of its children faults while online, the entire service group is marked faulted and failover occurs.
- **AutoStart** (default = 1). If the resource is set to AutoStart, it is brought online in response to a service group command. All resources designated as AutoStart=1 must be online for the service group to be considered online. (This attribute is unrelated to AutoStart attributes for service groups.)
- **Enabled**. If the resource is set to Enabled, the agent for the resource's type manages the resource. The default is 1 for resources defined in the configuration file `main.cf`, 0 for resources added on the command line.

Note: Adding resources on the command line requires several steps, and the agent must be prevented from managing the resource until the steps are completed. For resources defined in the configuration file, the steps are completed before the agent is started.

Adding resources

Add resource to a service group or remove resources from a service group.

To add a resource

- ◆ Type the following command:

```
hares -add resource resource_type service_group
```

The resource name must be unique throughout the cluster. The resource type must be defined in the configuration language. The resource belongs to the group *service_group*.

Deleting resources

Delete resources from a service group.

To delete a resource

- ◆ Type the following command:

```
# hares -delete resource
```

VCS does not delete online resources. However, you can enable deletion of online resources by changing the value of the DeleteOnlineResource attribute.

See “[Cluster attributes](#)” on page 688.

To delete a resource forcibly, use the `-force` option, which takes the resource offline irrespective of the value of the DeleteOnlineResource attribute.

```
# hares -delete -force resource
```

Adding, deleting, and modifying resource attributes

Resource names must be unique throughout the cluster and you cannot modify resource attributes defined by the system, such as the resource state.

To modify a new resource

- ◆ Type the following command:

```
# hares -modify resource attribute value
# hares -modify <resource> <attr> <value>
          [-sys <system>] [-wait [-time <waittime>]]
```

The variable *value* depends on the type of attribute being created.

To set a new resource’s enabled attribute to 1

- ◆ Type the following command:

```
# hares -modify resourceA Enabled 1
```

The agent managing the resource is started on a system when its Enabled attribute is set to 1 on that system. Specifically, the VCS engine begins to monitor the resource for faults. Agent monitoring is disabled if the Enabled attribute is reset to 0.

To add a resource attribute

```
# haattr -add resource_type attribute [value]
          [dimension] [default ...]
```

The variable *value* is a -string (default), -integer, or -boolean.

The variable *dimension* is -scalar (default), -keylist, -assoc, or -vector.

The variable *default* is the default value of the attribute and must be compatible with the *value* and *dimension*. Note that this may include more than one item, as indicated by ellipses (. . .).

To delete a resource attribute

```
# haattr -delete resource_type attribute
```

To add a static resource attribute

```
# haattr -add -static resource_type static_attribute [value]  
[dimension] [default ...]
```

To delete a static resource attribute

```
# haattr -delete -static resource_type static_attribute
```

To add a temporary resource attribute

```
# haattr -add -temp resource_type attribute [value]  
[dimension] [default ...]
```

To delete a temporary resource attribute

```
# haattr -delete -temp resource_type attribute
```

To modify the default value of a resource attribute

```
# haattr -default resource_type attribute new_value ...
```

The variable *new_value* refers to the attribute's new default value.

Defining attributes as local

Localizing an attribute means that the attribute has a per-system value for each system listed in the group’s SystemList. These attributes are localized on a per-resource basis. For example, to localize the attribute *attribute_name* for *resource* only, type:

```
# hares -local resource attribute_name
```

Note that global attributes cannot be modified with the `hares -local` command. The following table lists the commands to be used to localize attributes depending on their dimension.

Table 6-3 Making VCS attributes local

Dimension	Task and Command
scalar	Replace a value: <code>-modify [object] attribute_name value [-sys system]</code>
vector	<ul style="list-style-type: none">■ Replace list of values: <code>-modify [object] attribute_name value [-sys system]</code>■ Add list of values to existing list: <code>-modify [object] attribute_name -add value [-sys system]</code>■ Update list with user-supplied values: <code>-modify [object] attribute_name -update entry_value ... [-sys system]</code>■ Delete user-supplied values in list (if the list has multiple occurrences of the same value, then all the occurrences of the value is deleted): <code>-modify [object] attribute_name -delete -key ... [-sys system]</code>■ Delete all values in list: <code>-modify [object] attribute_name -delete -keys [-sys system]</code>

Table 6-3 Making VCS attributes local

Dimension	Task and Command
keylist	<div><div><div>■</div><div>Replace list of keys (duplicate keys not allowed): <code>-modify [object] attribute_name value ... [-sys system]</code></div></div><div><div>■</div><div>Add keys to list (duplicate keys not allowed): <code>-modify [object] attribute_name -add value ... [-sys system]</code></div></div><div><div>■</div><div>Delete user-supplied keys from list: <code>-modify [object] attribute_name -delete key ... [-sys system]</code></div></div><div><div>■</div><div>Delete all keys from list: <code>-modify [object] attribute_name -delete -keys [-sys system]</code></div></div></div>
association	<div><div><div>■</div><div>Replace list of key-value pairs (duplicate keys not allowed): <code>-modify [object] attribute_name value ... [-sys system]</code></div></div><div><div>■</div><div>Add user-supplied list of key-value pairs to existing list (duplicate keys not allowed): <code>-modify [object] attribute_name -add value ... [-sys system]</code></div></div><div><div>■</div><div>Replace value of each key with user-supplied value: <code>-modify [object] attribute_name -update key value ... [-sys system]</code></div></div><div><div>■</div><div>Delete a key-value pair identified by user-supplied key: <code>-modify [object] attribute_name -delete key ... [-sys system]</code></div></div><div><div>■</div><div>Delete all key-value pairs from association: <code>-modify [object] attribute_name -delete -keys [-sys system]</code></div></div><div><div>Note: If multiple values are specified and if one is invalid, VCS returns an error for the invalid value, but continues to process the others. In the following example, if sysb is part of the attribute SystemList, but sysa is not, sysb is deleted and an error message is sent to the log regarding sysa.</div><div><code>hagrp -modify group1 SystemList -delete sysa sysb [-sys system]</code></div></div></div>

Linking and unlinking resources

Link resources to specify a dependency between them. A resource can have an unlimited number of parents and children. When linking resources, the parent cannot be a resource whose Operations attribute is equal to None or OnOnly. Specifically, these are resources that cannot be brought online or taken offline by an agent (None), or can only be brought online by an agent (OnOnly).

Loop cycles are automatically prohibited by the VCS engine. You cannot specify a resource link between resources of different service groups.

To link resources

- ◆ Type the following command:

```
# hares -link parent_resource child_resource
```

The variable *parent_resource* depends on *child_resource* being online before going online itself. Conversely, *parent_resource* go offline before *child_resource* goes offline.

For example, a NIC resource must be available before an IP resource can go online, so for resources IP1 of type IP and NIC1 of type NIC, specify the dependency as:

```
# hares -link IP1 NIC1
```

To unlink resources

- ◆ Type the following command:

```
# hares -unlink parent_resource child_resource
```

Bringing resources online

To bring a resource online

- ◆ Type the following command:

```
# hares -online resource -sys system
```

Taking resources offline

To take a resource offline

- ◆ Type the following command:

```
# hares -offline [-ignoreparent| -parentprop] resource
-sys system
```

The option `-ignoreparent` enables a resource to be taken offline even if its parent resources in the service group are online. This option does not work if taking the resources offline violates the group dependency.

To take a resource and its parent resources offline

- ◆ Type the following command:

```
# hares -offline -parentprop resource -sys system
```

The command stops all parent resources in order before taking the specific resource offline.

To take a resource offline and propagate the command to its children

- ◆ Type the following command:

```
# hares -offprop [-ignoreparent] resource -sys system
```

As in the above command, the option `-ignoreparent` enables a resource to be taken offline even if its parent resources in the service group are online. This option does not work if taking the resources offline violates the group dependency.

Probing a resource

To prompt an agent to monitor a resource on a system

- ◆ Type the following command:

```
# hares -probe resource -sys system
```

Though the command may return immediately, the monitoring process may not be completed by the time the command returns.

Clearing a resource

To clear a resource

- ◆ Type the following command:
Initiate a state change from `RESOURCE_FAULTED` to `RESOURCE_OFFLINE`:

```
# hares -clear resource [-sys system]
```

Clearing a resource initiates the online process previously blocked while waiting for the resource to become clear. If *system* is not specified, the fault is cleared on each system in the service group's `SystemList` attribute. See [“To clear faulted, non-persistent resources in a service group”](#) on page 221.
This command also clears the resource's parents. Persistent resources whose static attribute `Operations` is defined as `None` cannot be cleared with this command and must be physically attended to, such as replacing a raw disk. The agent then updates the status automatically.

Administering systems

To modify a system's attributes

- ◆ Type the following command:

```
# hasys -modify modify_options
```

Some attributes are internal to VCS and cannot be modified. For details on system attributes, see “[About the -modify option](#)” on page 181.

To display the value of a system's node ID as defined in the file /etc/llttab

- ◆ Type the following command:

```
# hasys -nodeid node_ID
```

To freeze a system (prevent groups from being brought online or switched on the system)

- ◆ Type the following command:

```
# hasys -freeze [-persistent] [-evacuate] system
```

- | | |
|-------------|--|
| -persistent | Enables the freeze to be “remembered” when the cluster is rebooted. Note that the cluster configuration must be in read/write mode and must be saved to disk (dumped) to enable the freeze to be remembered. |
| -evacuate | Fails over the system's active service groups to another system in the cluster before the freeze is enabled. |

To unfreeze a frozen system (reenable online and switch of service groups)

- ◆ Type the following command:

```
# hasys -unfreeze [-persistent] system
```

To run a command on any system in a cluster

- ◆ Type the following command:

```
# hacli -cmd command [-sys | -server system(s)]
```

Issues a command to be executed on the specified system(s). VCS must be running on the systems.

The use of the hacli command requires setting HacliUserLevel to at least COMMANDROOT. By default, the HacliUserLevel setting is NONE.

If the users do not want the root user on system A to enjoy root privileges on another system B, HacliUserLevel should remain set to NONE (the default) on system B.

You can specify multiple systems separated by a single space as arguments to the option -sys. If no system is specified, command runs on all systems in

cluster with VCS in a RUNNING state. The command argument must be entered within double quotes if command includes any delimiters or options.

Administering clusters

Configuring and unconfiguring the cluster UUID value

When you install VCS using the installer, the installer generates the cluster UUID (Universally Unique ID) value. This value is the same across all the nodes in the cluster.

You can use the `uuidconfig` utility to display, copy, configure, and unconfigure the cluster UUID (universally unique id) on the cluster nodes.

Make sure you have ssh or rsh communication set up between the systems. The utility uses ssh by default.

To display the cluster UUID value on the VCS nodes

- ◆ Run the following command to display the cluster UUID value:
 - For specific nodes:

```
# /opt/VRTSvcs/bin/uuidconfig.pl [-rsh] -clus -display sys1 [sys2 sys3...]
```
 - For all nodes that are specified in the `/etc/llthosts` file:

```
# /opt/VRTSvcs/bin/uuidconfig.pl -display -clus -use_llthost
```

To configure cluster UUID on the VCS nodes

- ◆ Run the following command to configure the cluster UUID value:
 - For specific nodes:

```
# /opt/VRTSvcs/bin/uuidconfig.pl [-rsh] -clus -configure sys1 [sys2 sys3...]
```
 - For all nodes that are specified in the `/etc/llthosts` file:

```
# /opt/VRTSvcs/bin/uuidconfig.pl -clus -configure -use_llthost
```

The utility configures the cluster UUID on the cluster nodes based on whether a cluster UUID exists on any of the VCS nodes:

- If no cluster UUID exists or if the cluster UUID is different on the cluster nodes, then the utility does the following:
 - Generates a new cluster UUID using the `/opt/VRTSvcs/bin/osuuid`.
 - Creates the `/etc/vx/.uuids/clusuuid` file where the utility stores the cluster UUID.
 - Configures the cluster UUID on all nodes in the cluster.

- If a cluster UUID exists and if the UUID is same on all the nodes, then the utility retains the UUID.
Use the `-force` option to discard the existing cluster UUID and create new cluster UUID.
- If some nodes in the cluster have cluster UUID and if the UUID is the same, then the utility configures the existing UUID on the remaining nodes.

To unconfigure cluster UUID on the VCS nodes

- ◆ Run the following command to unconfigure the cluster UUID value:

- For specific nodes:

```
# /opt/VRTSvcs/bin/uuidconfig.pl [-rsh] -clus
-unconfigure sys1 [sys2 sys3...]
```
- For all nodes that are specified in the `/etc/llthosts` file:

```
# /opt/VRTSvcs/bin/uuidconfig.pl [-rsh] -clus
-unconfigure -use_llthost
```

The utility removes the `/etc/vx/.uuids/clusuuid` file from the nodes.

To copy the cluster UUID from one node to other nodes

- ◆ Run the following command to copy the cluster UUID value:

```
# /opt/VRTSvcs/bin/uuidconfig.pl [-rsh] -clus -copy
-from_sys sys -to_sys sys1 sys2 [sys3...]
```

The utility copies the cluster UUID from a system that is specified using the `-from_sys` option to all the systems that are specified using the `-to_sys` option.

Retrieving version information

Retrieve information about the version of VCS running on the system.

To retrieve information about the VCS version on the system

- ◆ Run one of the following commands:

```
# had -version
```

The command retrieves information about the engine version, the join version, the build date, and the PSTAMP.

```
# had -v
```

The command retrieves information about the engine version.

Adding and removing systems

This section provides an overview of tasks involved in adding and removing systems from a cluster. For detailed instructions, see the *Veritas Cluster Server Installation Guide*.

To add a system to a cluster

- 1 Make sure the system meets the hardware and software requirements for VCS.
See the *Veritas Cluster Server Installation Guide* for details.
- 2 Set up the private communication links from the new system.
- 3 Install VCS and required patches on the new system.
- 4 Add the VCS license key.
See [“Installing a VCS license”](#) on page 185.
- 5 Configure LLT and GAB to include the new system in the cluster membership.
- 6 Add the new system using the `hasys -add` command.

To remove a node from a cluster

- 1 Make a backup copy of the current configuration file, `main.cf`.
- 2 Switch or remove any VCS service groups from the node. The node cannot be removed as long as it runs service groups on which other service groups depend.
- 3 Stop VCS on the node.
`# hastop -sys systemname`
- 4 Delete the system from the SystemList of all service groups.
`# hagrps -modify groupname SystemList -delete systemname`
- 5 Delete the node from the cluster.
`# hasys -delete systemname`
- 6 Remove the entries for the node from the following file on each remaining node: `/etc/llthosts`.
- 7 Change the node count entry from the following file on each remaining node: `/etc/gabtab`.
- 8 Unconfigure GAB and LLT on the node leaving the cluster.
- 9 Remove VCS and other packages from the node.
- 10 Remove GAB and LLT configuration files from the node.

Changing ports for VCS

You can change some of the default ports for VCS and its components.

The following is a list of changes that you can make and information concerning ports in a VCS environment:

- Changing VCS's default port.
Add an entry for a VCS service name in `/etc/services` file, for example:

```
vcs          3333/tcp          # Veritas Cluster Server
```

Where 3333 in the example is the port number where you want to run VCS. When the engine starts, it listens on the port that you configured above (3333) for the service. You need to modify the port to the `/etc/services` file on all the nodes of the cluster.
See [“To change the default port”](#) on page 235.
- You do not need to make changes for agents or HA commands. Agents and HA commands use locally present UDS sockets to connect to the engine, not TCP/IP connections.
- You do not need to make changes for HA commands that you execute to talk to a remotely running VCS engine (HAD), using the facilities that the `VCS_HOST` environment variable provides. You do not need to change these settings because the HA command queries the `/etc/services` file and connects to the appropriate port.
- For GUIs (the Java Console and Management Console), you can specify the port number that you want the GUI to connect to while logging into the GUI. You have to specify the port number that is configured in the `/etc/services` file (for example 3333 above).

To change the default port

- 1 Stop VCS.
- 2 Add an entry service name `vcs` in `/etc/services`.

```
vcs          3333/tcp          # Veritas Cluster Server
```
- 3 You need to modify the port to the `/etc/services` file on all the nodes of the cluster.
- 4 Restart VCS.
- 5 Check the port.

```
# netstat -an|grep 3333
*.3333      *.*          0    0 49152      0 LISTEN
*.3333      *.*          0    0 49152      0 LISTEN
```
- 6 Use the Java Console to connect to VCS through port 3333.

Setting cluster attributes from the command line

To modify a cluster attribute

```
# haclus [-help [-modify]]
```

To update the EngineClass

```
# haclus -modify EngineClass value
```

For example, to set the EngineClass attribute to RealTime:

```
# haclus -modify EngineClass "RT"
```

To update the EnginePriority

```
# haclus -modify EnginePriority value
```

For example, to set the EnginePriority to 20:

```
# haclus -modify EnginePriority "20"
```

To update the ProcessClass

```
# haclus -modify ProcessClass value
```

For example, to set the ProcessClass to TimeSharing:

```
# haclus -modify ProcessClass "TS"
```

To update the ProcessPriority

```
# haclus -modify ProcessPriority value
```

For example, to set the ProcessPriority to 40:

```
# haclus -modify ProcessPriority "40"
```

Note: For the attributes EngineClass and EnginePriority, changes are effective immediately. For ProcessClass and ProcessPriority changes become effective only for processes fired *after* the execution of the `haclus` command.

Initializing cluster attributes in the configuration file

You may assign values for cluster attributes while configuring the cluster. (See “[Cluster Attributes](#)” on page 645 for a description of each attribute cited below.)

Review the following sample configuration:

```
cluster vcs-india (
    EngineClass = "RT"
    EnginePriority = "20"
    ProcessClass = "TS"
    ProcessPriority = "40"
```

Enabling and disabling Security Services

This section describes how to enable and disable Security Services. *Do not edit the VCS configuration file `main.cf` to enable or disable VxSS.*

To enable Symantec Product Authentication Service (AT), VCS requires a system in your enterprise that serves as a root broker. You can do one of the following:

- Use an existing root broker system in your enterprise.
- Set up a root broker system.
See the *Veritas Cluster Server Installation Guide* for instructions on setting up a root broker.
- Configure one of the nodes in the cluster to serve as root broker.
When you enable AT on a VCS cluster using the `installvcs` program, the installer provides you an option to configure one of the nodes as root broker.

To enable Symantec Product Authentication Services on a Veritas Cluster Server cluster

- 1 Make sure that all nodes in the cluster are running.
hasys -state
The output must show the SysState value as RUNNING.
- 2 If you want to use an external root broker system, verify you have a root broker configured.
See the *Veritas Cluster Server Installation Guide* for instructions.
- 3 Start the `installvcs` program with the `-security` option.
/opt/VRTS/install/installvcs -security
The installer displays the directory where the logs are created.
- 4 Enter **1** to enable the Authentication Service on the cluster.
1) Enable Symantec Security Services on a VCS Cluster

2) Disable Symantec Security Services on a VCS Cluster

Select the Security option you would like to perform [1-2,q] **1**

- 5 If Veritas Cluster Server is not configured in the system from where you started the installvcs program, enter the name of a node in the cluster that you want to enable the Authentication Service.

Enter the name of one system in the VCS Cluster that you would like to enable Veritas Security Services: **north**

The installer proceeds to verify communication with the node in the cluster.

- 6 Review the output as the installer verifies whether Veritas Cluster Server configuration files exist.

The installer also verifies that Veritas Cluster Server is running on all systems in the cluster.

- 7 Press Enter to confirm that you want to enable the Authentication Service.

Would you like to enable Symantec Security Services on this cluster? [y,n,q] (y) **y**

- 8 Proceed with the configuration tasks.

See the *Veritas Cluster Server Installation Configuration Guide* for details on the configuration modes.

Based on the root broker system you plan to use, do one of the following:

External root broker system	Enter the root broker name at the installer prompt. For example: If you already have an external RB(Root Broker) installed and configured, enter the RB name, or press Enter to skip: [b] venus
-----------------------------	---

One of the nodes as root broker	<p>Press Enter at the installer prompt:</p> <p>If you already have an external RB(Root Broker) installed and configured, enter the RB name, or press Enter to skip: [b]</p> <p>If AT is not already configured on any of the nodes, the installer asks you to choose a node to use as root broker:</p> <p>Do you want to configure galaxy as RAB, and other nodes as AB? [y,n,q,b] (y)</p> <p>Based on the node you choose, the installer configures the node to run in RAB (Root+Authentication Broker) mode. The installer configures the other nodes as authentication brokers.</p>
---------------------------------	--

- 9 Review the output as the installer modifies the Veritas Cluster Server configuration files to enable the Authentication Service, and starts Veritas Cluster Server in a secure mode.

The installer creates the Security service group, creates Authentication Server credentials on each node in the cluster and Web credentials for Veritas Cluster Server users, and sets up trust with the root broker.

To disable Symantec Product Authentication Services on a Veritas Cluster Server cluster

- 1 Verify you have a root broker configured.
See the *Veritas Cluster Server Installation Guide* for instructions.
- 2 Start the `installvcs` program with the `-security` option.
`/opt/VRTS/install/installvcs -security`
The installer displays the directory where the logs are created.
- 3 Enter **2** to disable the Authentication Service on the cluster.
1) Enable Symantec Security Services on a VCS Cluster
2) Disable Symantec Security Services on a VCS Cluster

Select the Security option you would like to perform [1-2,q] **2**
- 4 If Veritas Cluster Server is not configured in the system from where you started the `installvcs` program, enter the name of a node in the cluster that you want to disable the Authentication Service.
Enter the name of one system in the VCS Cluster that you would like to disable Symantec Security Services: **north**
- 5 Review the output as the installer proceeds with a basic verification.
- 6 Press Enter at the prompt to confirm that you want to disable the Authentication Service.
Would you like to disable Symantec Security Services on this cluster? [y,n,q] (y) **y**
- 7 Review the output as the installer modifies the Veritas Cluster Server configuration files to disable the Authentication Service and starts Veritas Cluster Server.

Administering resource types

Adding, deleting, and modifying resource types

After creating a resource type, use the command `haattr` to add its attributes. By default, resource type information is stored in the `types.cf` configuration file.

To add a resource type

```
# haatype -add resource_type
```

To delete a resource type

```
# hatype -delete resource_type
```

You must delete all resources of the type before deleting the resource type.

To add or modify resource types in main.cf without shutting down VCS

```
# hatype -modify resource_type SourceFile "./resource_type.cf"
```

The information regarding *resource_type* is stored in the file *config/resource_type.cf*, and an include line for *resource_type.cf* is added to the *main.cf* file. Make sure that the path to the *SourceFile* exists on all nodes before you run this command.

To set the value of static resource type attributes

```
# hatype -modify ...
```

Overriding resource type static attributes

You can override some resource type static attributes and assign them resource-specific values. When a static attribute is overridden and the configuration is saved, the *main.cf* file includes a line in the resource definition for the static attribute and its overridden value.

To override a type's static attribute

```
# hares -override resource static_attribute
```

To restore default settings to a type's static attribute

```
# hares -undo_override resource static_attribute
```

Initializing resource type scheduling and priority attributes

The following configuration shows how to initialize resource type scheduling and priority attributes through configuration files. The example shows attributes of a *FileOnOff* resource.

```
type FileOnOff (  
    static str AgentClass = RT  
    static str AgentPriority = 10  
    static str ScriptClass = RT  
    static str ScriptPriority = 40  
    static str ArgList[] = { PathName }  
    str PathName  
)
```


Setting Scheduling and Priority attributes

To update the AgentClass

```
# hatype -modify resource_type AgentClass value
```

For example, to set the AgentClass attribute of the FileOnOff resource to RealTime, type:

```
# hatype -modify FileOnOff AgentClass "RT"
```

To update the AgentPriority

```
# hatype -modify resource_type AgentPriority value
```

For example, to set the AgentPriority attribute of the FileOnOff resource to 10, type:

```
# hatype -modify FileOnOff AgentPriority "10"
```

To update the ScriptClass

```
# hatype -modify resource_type ScriptClass value
```

For example, to set the ScriptClass of the FileOnOff resource to RealTime, type:

```
# hatype -modify FileOnOff ScriptClass "RT"
```

To update the ScriptPriority

```
# hatype -modify resource_type ScriptPriority value
```

For example, to set the ScriptClass of the FileOnOff resource to RealTime, type:

```
# hatype -modify FileOnOff ScriptPriority "40"
```

Note: For attributes AgentClass and AgentPriority, changes are effective immediately. For ScriptClass and ScriptPriority, changes become effective for scripts fired after the execution of the `hatype` command.

Using the `-wait` option in scripts

The `-wait` option is for use in scripts using VCS commands to change attribute values. The option blocks the VCS command until the value of the specified attribute is changed or until the timeout, if specified, expires. Specify the timeout in seconds.

The option can be used only with changes to scalar attributes.

The `-wait` option is supported with the following commands:

- **haclus**

```
haclus -wait attribute value [-clus cluster] [-time timeout]
```

Use the **-clus** option in a global cluster environment.

■ **hagrp**

```
hagrp -wait group attribute value [-clus cluster] [-sys system]  
[-time timeout]
```

Use the **-sys** option when the scope of the attribute is local.

Use the **-clus** option in a global cluster environment.

■ **hares**

```
hares -wait resource attribute value [-clus cluster] [-sys  
system] [-time timeout]
```

Use the **-sys** option when the scope of the attribute is local.

Use the **-clus** option in a global cluster environment.

■ **hasys**

```
hasys -wait system attribute value [-clus cluster] [-time  
timeout]
```

Use the **-clus** option in a global cluster environment.

See the man pages associated with these commands for more information.

Running HA fire drills

The service group must be online when you run the HA fire drill.

See “[Testing resource failover using HA fire drills](#)” on page 263

To run HA fire drill for a specific resource

- ◆ Type the following command.

```
# hares -action <resname> <vfdaction>.vfd -sys <sysname>
```

The command runs the infrastructure check verifies whether the system <sysname> has the required infrastructure to host the resource <resname>, should a failover require the resource to come online on the system. For the variable <sysname>, specify the name of a system on which the resource is offline. The variable <vfdaction> specifies the Action defined for the agent. The HA fire drill checks for a resource type can are defined in the SupportedActions attribute and can be identified with the .vfd suffix.

To run HA fire drill for a service group

- ◆ Type the following command.

```
# havfd <grpname> -sys <sysname>
```

The command runs the infrastructure check and verifies whether the system <sysname> has the required infrastructure to host resources in the service group <grpname> should a failover require the service group to come online on the system. For the variable <sysname>, specify the name of a system on which the resource is offline

To fix detected errors

- ◆ Type the following command.

```
# hares -action <resname> <vfdaction>.vfd fix -sys <sysname>
```

The variable <vfdaction> represents the check that reported errors for the system specified by the variable <sysname>. The HA fire drill checks for a resource type can are defined in the SupportedActions attribute and can be identified with the .vfd suffix.

Administering simulated clusters from the command line

VCS Simulator is a tool to assist you in building and simulating cluster configurations. With VCS Simulator you can predict service group behavior during cluster or system faults, view state transitions, and designate and fine-tune various configuration parameters. This tool is especially useful when evaluating complex, multi-node configurations. It is convenient in that you can design a specific configuration without test clusters or changes to existing configurations.

You can also fine-tune values for attributes governing the rules of failover, such as Load and Capacity in a simulated environment. VCS Simulator enables you to simulate various configurations and provides the information you need to make the right choices. It also enables simulating global clusters.

See [“Predicting VCS behavior using VCS Simulator”](#) on page 265.

Configuring applications and resources in VCS

- [About configuring resources and applications](#)
- [About VCS bundled agents](#)
- [Configuring the RemoteGroup agent](#)
- [Configuring the RemoteGroup agent](#)
- [Configuring NFS service groups](#)
- [Testing resource failover using HA fire drills](#)

About configuring resources and applications

Configuring resources and applications in VCS involves the following tasks:

- Create a service group comprising all resources required for the application. VCS provides configuration wizards to configure commonly-used resources. You can also use Cluster Manager (Java Console), the web-based Cluster Management Console, or the command line to configure resources.
- Add required resources to the service group and configure them. For example, to configure a database in VCS, you must configure resources for the database and for the underlying shared storage and network resources.

Use appropriate agents to configure resources.
See “[About VCS bundled agents](#)” on page 246.
Configuring a resource involves defining values for its attributes. See the *Veritas Cluster Server Bundled Agents Reference Guide* for a description of the agents provided by VCS.

The resources must be logically grouped in a service group. When a resource faults, the entire service group fails over to another node.

- Assign dependencies between resources. For example, an IP resource depends on a NIC resource.
- Bring the service group online to make the resources available.

About VCS bundled agents

Bundled agents are categorized according to the type of resources they make available.

Storage agents

Storage agents make your shared disks, diskgroups, volumes, and mounts highly available.

Agent	Description
■ DiskGroup	Brings online, takes offline, and monitors a Veritas Volume Manager (VxVM) disk group.
■ DiskGroupSnap	Brings online, takes offline, and monitors disk groups used for fire drill testing.
■ Volume agent	Brings online, takes offline, and monitors a Veritas Volume Manager (VxVM) volume.

Agent	Description
■ LVMVG (AIX)	Activates, deactivates, and monitors a Logical Volume Manager (LVM) volume group.
■ Mount	Brings online, takes offline, and monitors a file system or NFS client mount point.

Network agents

Network agents make your IP addresses and computer names highly available.

Agent	Description
■ NIC	Monitors a NIC (Network Interface Card)
■ IP	Monitors an IP address.
■ MultiNICA	Monitors multiple network interfaces.
■ IPMultiNIC	Manages the virtual IP address configured as an alias on an interface of a MultiNICA resource.
■ MultiNICB (AIX, HP-UX, Solaris)	Monitors multiple network interfaces.
■ IPMultiNICB (AIX, HP-UX, Solaris)	Configures and manages virtual IP addresses (IP aliases) on an active network device specified by the MultiNICB resource.
■ DNS	Updates and monitors the canonical name (CNAME) mapping in the domain name server when failing over applications across subnets (performing a wide-area failover.)

File share agents

File Service agents make shared directories and subdirectories highly available.

Agent	Description
■ NFS	Monitors a shared directory. Use in conjunction with the NFSRestart agent.
■ NFSRestart	Recovers NFS record locks after sudden reboots or crashes on clients and servers. This avoids file corruption and provides the high availability of NFS record locks. Use with the NFS agent. Configure as the top-most resource in the service group dependency.
■ Share	Shares, unshares, and monitors a directory.
■ Samba	Suite of three agents that work together to provide high availability to Samba shares. Include SambaServer, SambaShare, and NetBIOS agents.
■ NetBIOS	Starts, stops, and monitors the nmbd daemon.

Services and Applications agents

Services and application agents make web sites, applications, and processes highly available.

Agent	Description
■ Apache	Makes an Apache Web server highly available.
■ Application	Brings applications online, takes them offline, and monitors their status.
■ Process	Starts, stops, and monitors a user-specified process.
■ ProcessOnOnly	Starts and monitors a user-specified process.
■ WPAR (AIX)	Brings workload partitions online, takes them offline, and monitors their status.
■ MemCPUAllocator (AIX)	Allocates CPU and memory for IBM AIX dedicated partitions.
■ Zone	Brings online, takes offline, and monitors Solaris 10 zones.
■ LDom	Brings online, takes offline, and monitors Solaris logical domains (LDoms.)

VCS infrastructure and support agents

The VCS infrastructure and support agents provide high availability for VCS-related operations.

Agent	Description
■ NotifierMngr	Monitors the VCS notifier process.
■ Proxy Agent	Monitors the state of a resource on a local or remote system.
■ Phantom Agent	Determines the state of service groups having resources of type <i>None</i> only.
■ RemoteGroup agent	Monitors the status of a remote failover service group.
■ Coordination Point agent	Monitors the I/O fencing coordination points.

Testing agents

Use the following agents to test VCS functionality:

Agent	Description
■ ElifNone Agent	Monitors a file. Checks for the file's absence
■ FileNone Agent	Monitors a file.
■ FileOnOff Agent	Creates a file, monitors it, and deletes it.
■ FileOnOnly Agent	Creates and monitors a file.

Configuring the RemoteGroup agent

The RemoteGroup agent monitors and manages failover service groups in a remote cluster. Use the RemoteGroup agent to establish dependencies between applications that are configured on different VCS clusters.

For example, you configure an Apache resource in a local cluster, and an Oracle resource in a remote cluster. In this example, the Apache resource depends on the Oracle resource. You can use the RemoteGroup agent to establish this dependency between these two resources.

See the *Veritas Cluster Server Bundled Agents Reference Guide* for more information about the agent and its attributes.

See “[Adding a RemoteGroup resource from the Java Console](#)” on page 146.

See “[Adding a RemoteGroup resource from Cluster Management Console](#)” on page 112.

About the ControlMode attribute

In the ControlMode attribute, you can use these values, depending on your needs: OnOff, MonitorOnly, and OnlineOnly.

OnOff

Select the OnOff value of this attribute when you want the RemoteGroup resource to manage the remote service group completely.

In case of one-to-one mapping, set the value of the AutoFailOver attribute of the remote service group to 0. This avoids unnecessary onlining or offlining of the remote service group.

MonitorOnly

Select the MonitorOnly value of this attribute when you want to monitor the state of the remote service group. When you choose the MonitorOnly attribute, the RemoteGroup agent does not have control over the remote service group and cannot bring it online nor take it offline.

The remote service group should be in an ONLINE state before you bring the RemoteGroup resource online.

Symantec recommends that the AutoFailOver attribute of the remote service group be set to 1.

OnlineOnly

Select the OnlineOnly value of this attribute when the remote service group takes a long time to come online or to go offline. When you use OnlineOnly for

the ControlMode attribute, a switch or fail over of the local service group with VCSSysName set to ANY does not cause the remote service group to be taken offline and brought online.

Taking the RemoteGroup resource offline does not take the remote service group offline.

If you are choosing one-to-one mapping between the local nodes and remote nodes, then the value of the AutoFailOver attribute of the remote service group must be 0.

Note: When you set the value of ControlMode to OnlineOnly or to MonitorOnly, the recommend value of the VCSSysName attribute of the RemoteGroup resource is ANY. If you want one-to-one mapping between the local nodes and the remote nodes, then a switch or fail over of local service group is impossible. It is important to note that in both these configurations the RemoteGroup agent does not take the remote service group offline.

Example: Configuring a RemoteGroup resource

In this example:

- VCS cluster (cluster1) provides high availability for Web services.
Configure a VCS service group (ApacheGroup) with an agent to monitor the Web server (for example Apache) to monitor the Web services.
- VCS cluster (cluster2) provides high availability for the database required by the Web-services.
Configure a VCS service group (OracleGroup) with a database agent (for example Oracle) to monitor the database.

The database resource must come online before the Web server comes online. You create this dependency using the RemoteGroup agent.

To configure the RemoteGroup agent in the example

- 1 Add a RemoteGroup resource in the ApacheGroup service group (in cluster 1).
- 2 Link the resources such that the Web server resource depends on the RemoteGroup resource.
- 3 Configure the RemoteGroup resource to monitor or manage the service group containing the database resource.
 - **IpAddress**—Set to the IP address or DNS name of a node in cluster2. You can also set this to a virtual IP address.
 - **GroupName**—Set to OracleGroup.

- **ControlMode**—Set to OnOff.
- **Username**—Set to the name of a user having administrative privileges for OracleGroup.
- **Password**—Encrypted password for defined in Username. Encrypt the password using the `vcsencrypt -agent` command.
- **VCSSysName**—Set to local, per-node values.
 - **VCSSysName@local1**—Set this value to remote1.
 - **VCSSysName@local2**—Set this value to remote2.

Note: If the remote cluster runs in secure mode, you must set the value for DomainType or BrokerIp attributes.

- 4 Set the value of the AutoFailOver attribute of the OracleGroup to 0.

Service group behavior with the RemoteGroup agent

Consider the following potential actions to better understand this solution.

Bringing the Apache service group online

- The Apache resource depends on the RemoteGroup resource.
- The RemoteGroup agent communicates to the remote cluster and authenticates the specified user.
- The RemoteGroup agent brings the database service group online in cluster2.
- The Apache resource comes online after the RemoteGroup resource is online.

Thus, you have established an application-level dependency across two different VCS clusters. The Apache resource does not go online unless the RemoteGroup goes online. The RemoteGroup resource does not go online unless the database service group goes online.

Unexpected offline of the database service group

- The RemoteGroup resource detects that the database group has gone OFFLINE or has FAULTED.
- The RemoteGroup resource goes into a FAULTED state.
- All the resources in the Apache service group are taken offline on the node.
- The Apache group fails over to another node.
- As part of the fail over, the Oracle service group goes online on another node in cluster2.

Taking the Apache service group offline

- All the resources dependant on the RemoteGroup resource are taken offline.
 - The RemoteGroup agent tries to take the Oracle service group offline.
 - Once the Oracle service group goes offline, the RemoteGroup goes offline.
- Thus, the Web server is taken offline before the database goes offline.

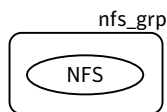
Configuring NFS service groups

When you configure an NFS service group, refer to the following instructions. On each node in your cluster, you can find sample NFS, NFSRestart, and Share configurations in the /etc/VRTSvcs/conf/sample_nfs/ directory.

Creating an NFS resource service group

Configure a separate parallel service group `nfs_grp`. Set the value of the `AutoStart` and `Parallel` attributes to 1 for the service group. The value for the `AutoStartList` must contain the list of all the cluster nodes in the service group. Configure an NFS resource (`nfs`) inside this service group. You can also put NIC resource in this service group to monitor a NIC. You may have to create a Phantom resource in this service group to display the correct state of the service group.

Figure 7-1 The NFS resource service group



Creating an NFS exports service group

Create new service groups that are based on the virtual IPs that you want to configure for share exports. If you want to export shares using four virtual IP addresses, you need to configure four service groups.

These service groups contain the Share and IP resources for exports. The value for the PathName attribute for the Share resource must be on shared storage and it must be visible to all nodes in the cluster. Isolate the storage for each service group from all the other service groups. If more than one service groups share the same storage, the storage resources must be kept in a separate service group. Make sure that the service groups depend on the storage service group.

- Create a Proxy resource (nfspxy) that points to the NFS resource (nfs), which is configured on the system.

You can also configure a group level dependency between the NFS exports service group and the NFS resource service group. You must use an online local firm dependency. You do not have to configure a Proxy resource that points to the NFS resource when you create this dependency.

- If you configure the backing store for the NFS exports with VxVM, create DiskGroup and Mount resources for the mountpoint that you want to export. Refer to Storage agents chapter for more details.
- If the backing store for the NFS exports is configured using LVM, configure the LVMVolumeGroup plus the Mount resources for the mount point that you want to export. Refer to Storage agents in the *Veritas Cluster Server Bundled Agents Reference Guide* for details.
- Create a Share resource. Set the PathName to the mountpoint that you want to export and NFSRes to the NFS resource (nfs). In case of multiple shares, create multiple Share resources with different values for their PathName attributes.

All the Share resources configured in the service group should have dependency on the following:

- The corresponding Mount resources that are inside the service group.
- The Proxy (nfspxy) resource that is configured in the service group.
- Create an IP resource. The value of the Address attribute for this IP resource is used to mount the NFS exports on the client systems. Make the IP resource depend on the Share resources that are configured in the service group.
- Create an NFSRestart resource. Set the NFSRes attribute to the NFS resource (nfs) that is configured on the system. Make the NFSRestart resource depend on the IP resource or the DNS resource (if you want to use NFS lock recovery.)

- Create a DNS resource if you want NFS lock recovery. The DNS resource depends on the IP resource.
- Refer to the sample configuration on how to configure the DNS resource.
See [“Sample configurations for NFS”](#) on page 258.

Figure 7-2 Dependency graph of an NFS service group with a single share

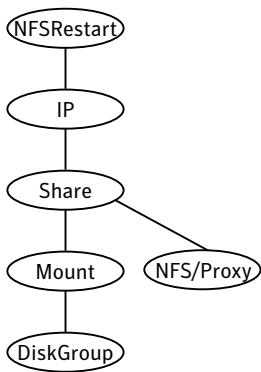


Figure 7-3 Dependency graph of an NFS service group with multiple shares

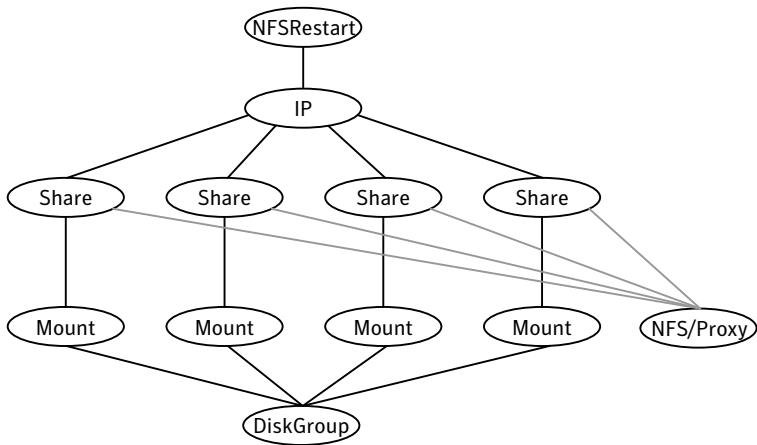
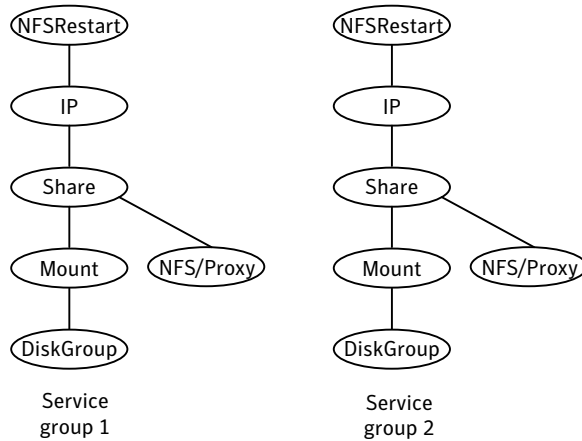


Figure 7-4 Dependency graph of multiple NFS service groups



Using triggers for NFSRestart resources

You must perform the following instructions for VCS configurations that have NFSRestart resources. Failure to perform these instructions can result in NFS/NFSRestart resources not functioning correctly.

Using the `nfs_preonline` and `nfs_postoffline` triggers

Symantec implemented this change to prevent the invocation of NFSRestart-related triggers when no NFSRestart resources in the VCS configuration.

To copy the `nfs_preonline` and `nfs_postoffline` files

- ◆ Copy the `nfs_preonline` and `nfs_postoffline` files to the `/opt/VRTSvcs/bin/triggers` directory.


```
# cp /opt/VRTSvcs/bin/sample_triggers/nfs_preonline \
/opt/VRTSvcs/bin/triggers
# cp /opt/VRTSvcs/bin/sample_triggers/nfs_postoffline \
/opt/VRTSvcs/bin/triggers
```

Enabling PreOnline for the NFS exports service group

For the service groups that contain the Share and NFSRestart resources, you must enable their PreOnline triggers.

Refer to the sample configurations for more information to enable these triggers.

Sample configurations for NFS

The following sample configurations describe different ways to set up NFS, NFSRestart, and Share resources. In these configurations, *platform_specific_nic0* represents the value of the Device attribute name for different platforms (eth0, lan0, en0, le0, etc.)

The AIX platform does not have the NFSRes attribute for the Share resource.

Note: On each node in your cluster, you can find sample NFS, NFSRestart, and Share configurations in /etc/VRTSvcs/conf/sample_nfs/.

Sample configuration for a service group without an NFS lock failover

```
include "types.cf"

cluster vcscclus (
    UserNames = { admin = IJKcJEjGKfKKiSKeJH, a = gJJd }
    Administrators = { admin, a }
)

system sysA (
)

system sysB (
)

group nfs_grp (
    SystemList = { sysA = 0, sysB = 1 }
    Parallel = 1
    AutoStart = 1
    AutoStartList = { sysA, sysB }
)

NFS test_nfs (
    Nproc = 16
)

group nfstest (
    SystemList = { sysA = 0, sysB = 1 }
    PreOnline @sysA = 1
    PreOnline @sysB = 1
)

DiskGroup test_dg (
    DiskGroup = nfsdg
)
```

```
IP test_ip (
    Device = platform_specific_nic0
    Address = "10.212.88.37"
    NetMask = "255.255.254.0"
)

Mount test_mnt (
    MountPoint = "/home/export"
    BlockDevice = "/dev/vx/dsk/nfsdg/vol0"
    FSType = vxfs
    MountOpt = rw
    FsckOpt = "-n"
)

NFSRestart test_nfsres (
    NFSRes = test_nfs
)

Share test_share (
    PathName = "/home/export"
    Options = "rw,nohide,fsid=0"
    NFSRes = test_nfs
)

Volume test_vol (
    DiskGroup = nfsdg
    Volume = vol0
)

requires group nfs_grp online local firm
test_ip requires test_share
test_mnt requires test_vol
test_nfsres requires test_ip
test_share requires test_mnt
test_vol requires test_dg

// resource dependency tree
//
//   group nfstest
//   {
//       NFSRestart test_nfsres
//       {
//           IP test_ip
//           {
//               Share test_share
//               {
//                   Mount test_mnt
//                   {
//                       Volume test_vol
//                       {
//                           DiskGroup test_dg
```

```
//
//
//
//
//
//
// }
```

Sample configuration for a service group with NFS lock failover

```
include "types.cf"

cluster vcsclus (
    UserNames = { admin = IJKcJEjGKfKKiSKeJH, a = gJd }
    Administrators = { admin, a }
)

system sysA (
)

system sysB (
)

group nfs_grp (
    SystemList = { sysA = 0, sysB = 1 }
    Parallel = 1
    AutoStart = 1
    AutoStartList = { sysA, sysB }
)

NFS test_nfs (
    Nproc = 16
)

group nfstest (
    SystemList = { sysA = 0, sysB = 1 }
    PreOnline @sysA = 1
    PreOnline @sysB = 1
)

DNS dns1 (
    Domain = "oradb.sym"
    TSIGKeyFile = "/Koradb.sym.+157+13021.private"
    StealthMasters = { "10.209.172.202" }
    ResRecord @sysA = { sysA = "10.212.88.37" }
    ResRecord @sysB = { sysB = "10.212.88.37" }
    CreatePTR = 1
    OffDelRR = 1
)
```

```
DiskGroup test_dg (  
    DiskGroup = nfsdg  
)  
  
IP test_ip (  
    Device = platform_specific_nic0  
    Address = "10.212.88.37"  
    NetMask = "255.255.254.0"  
)  
  
Mount test_mnt (  
    MountPoint = "/home/export"  
    BlockDevice = "/dev/vx/dsk/nfsdg/vol0"  
    FSType = vxfs  
    MountOpt = rw  
    FsckOpt = "-n"  
)  
  
Mount lock_mnt (  
    MountPoint = "/home/lock"  
    BlockDevice = "/dev/vx/dsk/nfsdg/vol1"  
    FSType = vxfs  
    MountOpt = rw  
    FsckOpt = "-n"  
)  
  
NFSRestart test_nfsres (  
    NFSRes = test_nfs  
    NFSLockFailover = 1  
    LocksPathName = "/home/lock"  
)  
  
Share test_share (  
    PathName = "/home/export"  
    Options = "rw,nohide,fsid=0"  
    NFSRes = test_nfs  
)  
  
Volume test_vol (  
    DiskGroup = nfsdg  
    Volume = vol0  
)  
  
Volume lock_vol (  
    DiskGroup = nfsdg  
    Volume = vol1  
)  
  
requires group nfs_grp online local firm  
dns1 requires test_ip  
test_ip requires test_share
```

```

test_mnt requires test_vol
lock_mnt requires lock_vol
test_nfsres requires dns1
test_nfsres requires lock_mnt
test_share requires test_mnt
test_vol requires test_dg
lock_vol requires test_dg

// resource dependency tree
//
//   group nfstest
//   {
//     NFSRestart test_nfsres
//     {
//       DNS dns1
//       {
//         IP test_ip
//         {
//           Share test_share
//           {
//             Mount test_mnt
//             {
//               Volume test_vol
//               {
//                 DiskGroup test_dg
//               }
//             }
//           }
//         }
//       }
//     }
//     Mount lock_mnt
//     {
//       Volume lock_vol
//       {
//         DiskGroup test_dg
//       }
//     }
//   }
// }

```

Configuring the Coordination Point agent

To monitor I/O fencing coordination points, configure the Coordination Point agent in the VCS cluster where you have configured I/O fencing.

For server-based fencing, you can either use the `-fencing` option of the installer to configure the agent or you can manually configure the agent. For disk-based fencing, you must manually configure the agent.

See the *Veritas Cluster Server Bundled Agents Reference Guide* for more information on the agent.

See the *Veritas Cluster Server Installation Guide* for instructions to configure the agent.

Testing resource failover using HA fire drills

Configuring high availability for a database or an application requires several infrastructure and configuration settings on multiple systems. However, cluster environments are subject to change after the initial setup. Administrators add disks, create new diskgroups and volumes, add new cluster nodes, or new NICs to upgrade and maintain the infrastructure. Keeping the cluster configuration updated with the changing infrastructure is critical.

HA fire drills detect discrepancies between the VCS configuration and the underlying infrastructure on a node; discrepancies that might prevent a service group from going online on a specific node.

See the *Veritas Cluster Server Bundled Agents Reference Guide* for information on which agents support HA fire drills.

About HA fire drills

The HA fire drill (earlier known as virtual fire drill) feature uses the Action function associated with the agent. The Action functions of the supported agents are updated to support the HA fire drill functionality—running infrastructure checks and fixing specific errors.

The infrastructure check verifies the resources defined in the VCS configuration file (`main.cf`) have the required infrastructure to fail over on another node. For example, an infrastructure check for the Mount resource verifies the existence of the mount directory defined in the MountPoint attribute for the resource.

You can run an infrastructure check only when the service group is online. The check verifies that the specified node is a viable failover target capable of hosting the service group.

The HA fire drill provides an option to fix specific errors detected during the infrastructure check.

Running an HA fire drill

You can run a HA fire drill from the command line or from Cluster Manager (Java Console).

See “[Running HA fire drill from the Java Console](#)” on page 161.

See “[Running HA fire drills](#)” on page 243.

Predicting VCS behavior using VCS Simulator

- [About VCS Simulator](#)
- [Administering VCS Simulator from the Java Console](#)
- [Administering VCS Simulator from the command line](#)

About VCS Simulator

VCS Simulator enables you to simulate and test cluster configurations. Use VCS Simulator to view and modify service group and resource configurations and test failover behavior. VCS Simulator can be run on a stand-alone system and does not require any additional hardware.

VCS Simulator runs an identical version of the VCS High Availability Daemon (HAD) as in a cluster, ensuring that failover decisions are identical to those in an actual cluster.

You can test configurations from different operating systems using VCS Simulator. For example, you can run VCS Simulator on a Windows system and test VCS configurations for Windows, Linux, and Solaris clusters. VCS Simulator also enables creating and testing global clusters.

You can administer VCS Simulator from the Java Console or from the command line.

To download VCS Simulator, go to <http://www.symantec.com/business/cluster-server> and click **Utilities**.

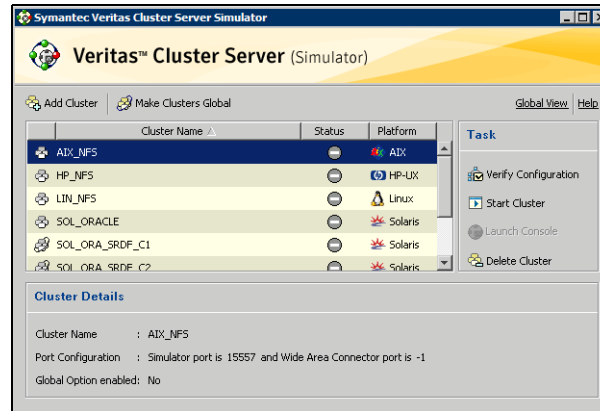
Simulator ports

VCS Simulator uses the following ports:

- Ports 15550 through 15559 and 15580 through 15585 to connect to the various cluster configurations.
- Ports 15560 through 15571 for the wide area connector (WAC) process. Set the WAC port to -1 to disable WAC simulation.

Administering VCS Simulator from the Java Console

The Simulator Console enables you to start, stop, and manage simulated clusters.



The console provides two views:

- Cluster View—Lists all simulated clusters.
- Global View—Lists global clusters.

Through the Java Console, VCS Simulator enables you to configure a simulated cluster panel, bring a system in an unknown state into an online state, simulate power loss for running systems, simulate resource faults, and save the configuration while VCS is offline. For global clusters, you can simulate the process of generating and clearing cluster faults.

You can run multiple simulated clusters on a system by using different port numbers for each cluster.

The Java Console provides the same views and features that are available for online configurations.

See “[Administering the cluster from Cluster Manager \(Java console\)](#)” on page 79.

Starting VCS Simulator from the Java Console

To start VCS Simulator from the Java Console (UNIX)

- 1 Type the following command to grant the system permission to display on the desktop:

```
xhost +
```

- 2 Configure the shell environment variable `DISPLAY` on the system where Cluster Manager will be launched. For example, if using Korn shell, type the following command to display on the system `myws`:

```
export DISPLAY=myws:0
```

- 3 Run the following command:

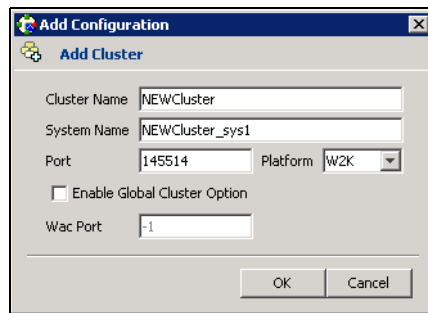
```
/opt/VRTSvc/bin/hasimgui
```

Creating a simulated cluster

You can start a sample cluster configuration or create a new simulated cluster.

To create a simulated cluster

- 1 In the Simulator console, click **Add Cluster**.
- 2 In the Add Cluster dialog box:



- Enter a name for the new cluster.
- Accept the suggested system name or enter a new name for a system in the cluster.
- Enter a unique port number for the simulated cluster.
- Select the platform for the cluster nodes.
- If the cluster is part of a global cluster configuration, select the **Enable Global Cluster Option** check box and enter a unique port number for the wide-area connector (WAC) process.
- Click **OK**.

VCS creates a simulated one-node cluster and creates a new directory for the cluster's configuration files. VCS also creates a user called *admin* with Cluster Administrator privileges. You can start the simulated cluster and administer it by launching the Java Console.

Deleting a cluster

Deleting a simulated cluster removes all files and directories that are associated with the cluster. Before deleting a cluster, make sure the cluster is not configured as a global cluster. You can delete global clusters from the Global View.

To delete a simulated cluster

- 1 From Simulator Explorer, select the cluster and click **Delete Cluster**.
- 2 In the confirmation dialog box, click **Yes**.

Starting a simulated cluster

Start the cluster to begin administering it.

To start a simulated cluster

- 1 In the Simulator console, select the cluster.
- 2 Click **Start Cluster**.
- 3 After the cluster starts, click **Launch Console** to administer the cluster.
- 4 Enter a valid user name and password to log on to the cluster.
VCS Simulator does not validate passwords; you can log on to a simulated cluster by entering a valid VCS user name. If you use the default configuration, enter **admin** for the user name and any non-blank value for password.
Cluster Explorer is launched upon initial logon, and the icons in the cluster panel change color to indicate an active panel.

Verifying a simulated cluster configuration

Verify that the configuration is valid.

To verify the simulated cluster configuration

- 1 In the Simulator console, select the cluster.
- 2 Click **Verify Configuration**.

Simulating an existing cluster configuration

Simulate an existing cluster configuration to test your cluster configuration.

To simulate an existing cluster configuration

- 1 In the Simulator console, click **Add Cluster**.

- 2 In the Add Cluster dialog box:
 - Enter a name for the new cluster.
 - Specify the system name. Make sure that you specify the name of one of the systems from the main.cf that you want to simulate.
 - Enter a unique port number for the simulated cluster.
 - Select the platform for the cluster nodes.
 - Click **OK**.VCS creates a simulated one-node cluster and creates a new directory for the cluster's configuration files.
- 3 Open the directory `<SIMULATOR_DIR>\<CLUSTERNAME>\conf\config`
- 4 Copy main.cf, types.cf, and all the required types files of the cluster that you want to simulate to this directory.
- 5 Click **Verify Configuration** on the Simulator console.
- 6 Click **Start Cluster**.

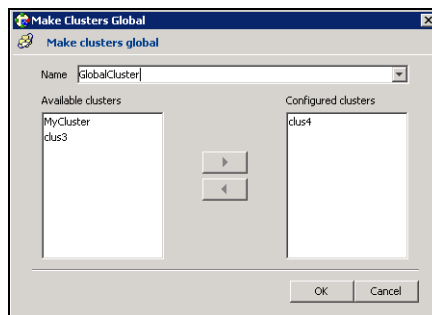
Simulating a global cluster configuration

Simulate a global cluster environment to test your global cluster configuration.

See [“How VCS global clusters work”](#) on page 466.

To simulate a global cluster configuration

- 1 Create the simulated clusters for the global configuration.
See [“Creating a simulated cluster”](#) on page 268.
Select the **Enable Global Cluster Option** check box and enter a unique port number for the wide-area connector (WAC) process.
- 2 In the Simulator console, click **Make Global**.
- 3 In the Make Global Configuration dialog box:



- Select an existing global cluster or enter the name for a new global cluster.
- From the **Available Clusters** list, select the clusters to add to the global cluster and click the right arrow. The clusters move to the **Configured Clusters** list.
- Click **OK**.

Bringing a system up

Bring a system up to simulate a running system.

To bring a system up

- 1 From Cluster Explorer, click the **Systems** tab of the configuration tree.
- 2 Right-click the system in an unknown state, and click **Up**.

Powering off a system

- 1 From Cluster Explorer, click the **Systems** tab of the configuration tree.
- 2 Right-click the online system, and click **Power Off**.

Saving the offline configuration

- 1 From Cluster Explorer, click **Save Configuration As** from the **File** menu.
- 2 Enter the path location.
- 3 Click **OK**.

Simulating a resource fault

Use VCS Simulator to imitate a resource fault.

To generate a resource fault

- 1 From Cluster Explorer, click the **Service Groups** tab of the configuration tree.
- 2 Right-click an online resource, click **Fault Resource**, and click the system name.

Simulating cluster faults in global clusters

Use VCS Simulator to imitate the process of generating and clearing cluster faults.

See “[Monitoring alerts](#)” on page 175.

To generate a cluster fault

- 1 From Cluster Explorer, click the cluster in the configuration tree.
- 2 Right-click the cluster, click **Fault Cluster**, and click the cluster name.
If any Cluster Explorer windows are open for the cluster being faulted, these become inoperative for a short period during which the Cluster Monitor tries to connect to the simulated High Availability Daemon for the cluster. Following this, an alert message appears and the Cluster Explorer windows close on their own.

Note: When a faulted cluster is brought up, its fault is automatically cleared. In case of a GCO configuration, the Remote Cluster status is also automatically updated. Hence there is no need to clear the cluster fault.

Simulating failed fire drills

Use VCS Simulator to demonstrate a failed fire drill. The following simulated clusters have fire drill service groups:

- SOL_ORA_SRDF_C2 (fire drill group is OracleGrp_fd)
- WIN_SQL_VVR_C2 (firedrill group is SQLPROD_fd)
- Win_Exch_2k3_Secondary (firedrill group is sample_fd)

See [“Setting up a fire drill”](#) on page 492.

To simulate a failed fire drill

- 1 Start Cluster Explorer and click the cluster in which you want to simulate the fire drill.
- 2 Select the fire drill service group from the Tree View, then select the Properties Tab in the right pane.
- 3 Click **Show all attributes**. Scroll down to choose the Tag attribute and double-click to edit the attribute value.
- 4 If prompted, switch the configuration to the read-write mode.
- 5 In the Edit Attribute window, set the value of the Tag attribute to the name of a critical resource in the FireDrill Service Group.
The Tag attribute values for service groups SQLPROD_fd (in cluster WIN_SQL_VVR_C2) and sample_fd (in cluster Win_Exch_2K3_secondary) should be blank before these modifications.
For the SQLPROD_fd fire-drill service group, set the attribute value to the name of the SQL Server instance - SQLServer2000-VSQL01_fd.
You do not need to change the attribute value for the Oracle group; by default, the Tag attribute of the OracleGrp_fd is set to the name of a critical resource.
- 6 Try to bring the FireDrill service group up. Right-click the service group in the Cluster Explorer and bring it online on a specified system. The FireDrill service group faults.

Note: To simulate a successful fire drill, keep the Tag attribute of the fire drill service group blank and bring the Firedrill service group online.

Administering VCS Simulator from the command line

Start VCS Simulator before creating or administering simulated clusters.

Note: VCS Simulator treats clusters that are created from the command line and the Java Console separately. Hence, clusters that are created from the command line are not visible in the graphical interface. If you delete a cluster from the command line, you may see the cluster in the Java Console.

Starting VCS Simulator from the command line

To start VCS Simulator from the command line (UNIX)

- 1 To simulate a cluster running a particular operating system, copy the `types.cf` file for the operating system from the `types` directory to `/opt/VRTScsim/default_clus/conf/config/`.
For example, if the cluster to be simulated runs on the AIX platform, copy the file `types.cf.aix`.
- 2 Add custom type definitions to the file, if required, and rename the file to `types.cf`.
- 3 If you have a `main.cf` file to run in the simulated cluster, copy it to `/opt/VRTScsim/default_clus/conf/config/`.
- 4 Start VCS Simulator:

```
VCS_SIMULATOR_HOME/bin/hasim -start system_name
```

where `VCS_SIMULATOR_HOME` is the Simulator installation directory, typically `/opt/VRTScsim`.
The variable `system_name` represents a system name, as defined in the configuration file `main.cf`. This command starts Simulator on port 14153.
For example, to start the default cluster:

```
VCS_SIMULATOR_HOME/bin/hasim -start sys1
```

Note that the default configuration includes system `sys1`.
- 5 Add systems to the configuration, if desired:

```
VCS_SIMULATOR_HOME/bin/hasim -sys -add system_name  
VCS_SIMULATOR_HOME/bin/hasim -up system_name
```
- 6 Verify the states of each node in the cluster:

```
VCS_SIMULATOR_HOME/bin/hasim -sys -state
```


Use the command line or the Java Console to manage the simulated cluster. See [“To simulate global clusters from the command line”](#) on page 275.

To start VCS Simulator from the command line (Windows)

VCS Simulator installs platform-specific types.cf files at the path %VCS_SIMULATOR_HOME%\types\. The variable %VCS_SIMULATOR_HOME% represents the Simulator installation directory, typically C:\Program Files\Veritas\VCS Simulator\.

- 1 To simulate a cluster running a particular operating system, copy the types.cf. file for the operating system from the types directory to %VCS_SIMULATOR_HOME%\default_clus\conf\config\.
For example, if the cluster to be simulated runs on the AIX platform, copy the file types.cf.aix.
- 2 Add custom type definitions to the file, if required, and rename the file to types.cf.
- 3 If you have a main.cf file to run in the simulated cluster, copy it to %VCS_SIMULATOR_HOME%\default_clus\conf\config\.
- 4 Start VCS Simulator:
%VCS_SIMULATOR_HOME%\bin> **hasim -start system_name**
The variable *system_name* represents a system name, as defined in the configuration file main.cf.
This command starts Simulator on port 14153.
- 5 Add systems to the configuration, if desired:
%VCS_SIMULATOR_HOME%\bin> **hasim -sys -add system_name**
%VCS_SIMULATOR_HOME%\bin> **hasim -up system_name**
- 6 Verify the state of each node in the cluster:
%VCS_SIMULATOR_HOME%\bin> **hasim -sys -state**
See [“To simulate global clusters from the command line”](#) on page 275.

To simulate global clusters from the command line

- 1 Install VCS Simulator in a directory (*sim_dir*) on your system.
See the section Installing VCS Simulator in the *Veritas Cluster Server Installation Guide*.
- 2 Set up the clusters on your system. Run the following command to add a cluster:
sim_dir/hasim -setupclus clustername -simport port_no -wacport port_no
Do not use *default_clus* as the cluster name when simulating a global cluster.
VCS Simulator copies the sample configurations to the path *sim_dir/clustername* and creates a system named *clustername_sys1*.
For example, to add cluster *clus_a* using ports 15555 and 15575, run the following command:

```
sim_dir/hasim -setupclus clus_a -simport 15555 -wacport 15575
```

Similarly, add the second cluster:

```
sim_dir/hasim -setupclus clus_b -simport 15556 -wacport 15576
```

To create multiple clusters without simulating a global cluster environment, specify -1 for the wacport.

- 3 Start the simulated clusters:

```
sim_dir/hasim -start clustername_sys1 -clus clustername
```

- 4 Set the following environment variables to access VCS Simulator from the command line:

- `VCS_SIM_PORT=port_number`

- `VCS_SIM_WAC_PORT=wacport`

Note that you must set these variables for each simulated cluster, otherwise Simulator always connects default_clus, the default cluster.

You can use the Java Console to link the clusters and to configure global service groups.

See “[Administering the cluster from Cluster Manager \(Java console\)](#)” on page 79.

You can also edit the configuration file main.cf manually to create the global cluster configuration.

Administering simulated clusters from the command line

The functionality of VCS Simulator commands mimic that of standard `ha` commands.

Command	Description
<code>hasim -start <i>system_name</i></code>	Starts VCS Simulator. The variable <i>system_name</i> represents the system that will transition from the <code>LOCAL_BUILD</code> state to <code>RUNNING</code> .
<code>hasim -setupclus <i>clustername</i> -simport port_no [-wacport port_no] [-sys <i>systemname</i>]</code>	Creates a simulated cluster and associates the specified ports with the cluster.
<code>hasim -deleteclus <clus></code>	Deletes specified cluster. Deleting the cluster removes all files and directories associated with the cluster. Before deleting a cluster, make sure the cluster is not configured as a global cluster.
<code>hasim -start <i>clustername_sys1</i> [-clus <i>clustername</i>] [- disablel10n]</code>	Starts VCS Simulator on the cluster specified by <i>clustername</i> . If you start VCS Simulator with the <code>-disablel10n</code> option, the simulated cluster does not accept localized values for attributes. Use this option when simulating a UNIX configuration on a Windows system to prevent potential corruption when importing the simulated configuration to a UNIX cluster.
<code>hasim -stop</code>	Stops the simulation process.
<code>hasim -poweroff <i>system_name</i></code>	Gracefully shuts down the system.
<code>hasim -up <i>system_name</i></code>	Brings the system up.
<code>hasim -fault <i>system_name</i> <i>resource_name</i></code>	Faults the specified resource on the specified system.
<code>hasim -online <i>system_name</i> <i>resource_name</i></code>	Brings specified resource online. This command is useful if you have simulated a fault of a persistent resource and want to simulate the fix.
<code>hasim -faultcluster <i>clustername</i></code>	Simulates a cluster fault.
<code>hasim -clearcluster <i>clustername</i></code>	Clears a simulated cluster fault.

Command	Description
<code>hasim -getsimconfig <i>cluster_name</i></code>	Retrieves information about VCS Simulator ports.
<code>hasim -hb [...]</code>	Equivalent to standard <code>hahb</code> command.
<code>hasim -disablel10n</code>	Disables localized inputs for attribute values. Use this option when simulating UNIX configurations on Windows systems.
<code>hasim -clus [...]</code>	Equivalent to standard <code>haclus</code> command.
<code>hasim -sys [...]</code>	Equivalent to standard <code>hasys</code> command.
<code>hasim -grp [...]</code>	Equivalent to standard <code>hagrps</code> command.
<code>hasim -res [...]</code>	Equivalent to standard <code>hares</code> command.
<code>hasim -type [...]</code>	Equivalent to standard <code>hatype</code> command.
<code>hasim -conf [...]</code>	Equivalent to standard <code>haconf</code> command.
<code>hasim -attr [...]</code>	Equivalent to standard <code>haattr</code> command.



VCS communication and operations

- [Chapter 9, “About communications, membership, and data protection in the cluster” on page 281](#)
- [Chapter 10, “Administering I/O fencing” on page 327](#)
- [Chapter 11, “Controlling VCS behavior” on page 361](#)
- [Chapter 12, “The role of service group dependencies” on page 409](#)



About communications, membership, and data protection in the cluster

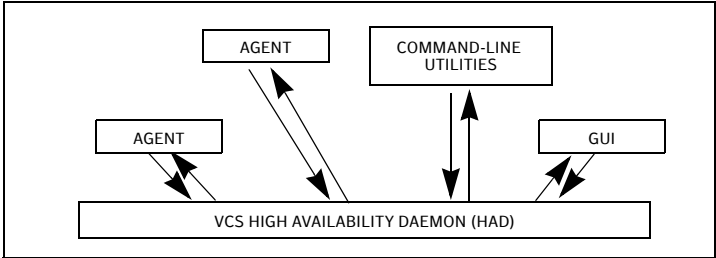
- [About cluster communications](#)
- [About cluster membership](#)
- [About membership arbitration](#)
- [About data protection](#)
- [Examples of VCS operation with I/O fencing](#)
- [About cluster membership and data protection without I/O fencing](#)
- [Examples of VCS operation without I/O fencing](#)
- [Summary of best practices for cluster communications](#)

About cluster communications

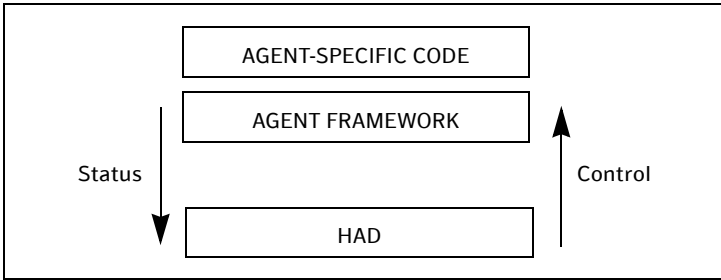
VCS uses local communications on a system and system-to-system communications.

About intra-system communications

Within a system, the VCS engine (HAD) uses a VCS-specific communication protocol known as Inter Process Messaging (IPM) to communicate with the GUI, the command line, and the agents. The following illustration shows basic communication on a single VCS system. Note that agents only communicate with HAD and never communicate with each other.



The following illustration depicts communication from a single agent to HAD.



The agent uses the agent framework, which is compiled into the agent itself. For each resource type configured in a cluster, an agent runs on each cluster system. The agent handles all resources of that type. The engine passes commands to the agent and the agent returns the status of command execution. For example, an agent is commanded to bring a resource online. The agent responds back with the success (or failure) of the operation. Once the resource is online, the agent communicates with the engine only if this status changes.

About inter-system cluster communications

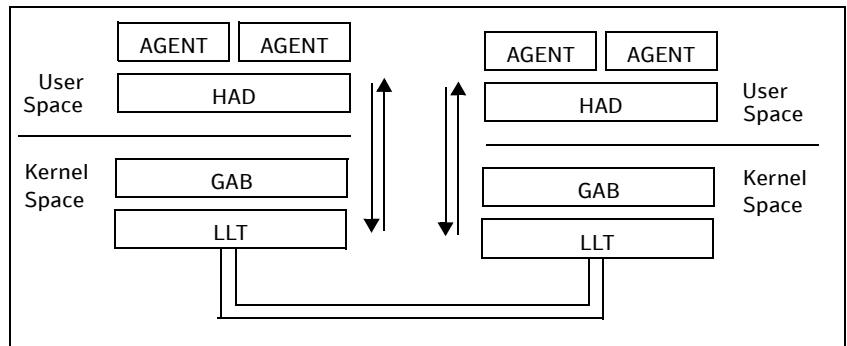
VCS uses the cluster interconnect for network communications between cluster systems. Each system runs as an independent unit and shares information at the cluster level. On each system the VCS High Availability Daemon (HAD), which is the decision logic for the cluster, maintains a view of the cluster configuration. This daemon operates as a replicated state machine, which means all systems in the cluster have a synchronized state of the cluster configuration. This is accomplished by the following:

- All systems run an identical copy of HAD.
- HAD on each system maintains the state of its own resources, and sends all cluster information about the local system to all other machines in the cluster.
- HAD on each system receives information from the other cluster systems to update its own view of the cluster.
- Each system follows the same code path for actions on the cluster.

The replicated state machine communicates over a proprietary communications package consisting of two components, *Group Membership Services/Atomic Broadcast (GAB)* and *Low Latency Transport (LLT)*.

Figure 9-5 illustrates the overall communications paths between two systems of the replicated state machine model.

Figure 9-5 Cluster communications with replicated state machine



Group Membership Services/Atomic Broadcast (GAB)

The Group Membership Services/Atomic Broadcast protocol (GAB) is responsible for cluster membership and reliable cluster communications. GAB has two major functions.

- **Cluster membership**
GAB maintains cluster membership by receiving input on the status of the heartbeat from each system via LLT. When a system no longer receives heartbeats from a cluster peer, LLT passes the heartbeat loss to GAB. GAB marks the peer as **DOWN** and excludes it from the cluster. In most configurations, membership arbitration is used to handle network partitions.
- **Cluster communications**
GAB's second function is reliable cluster communications. GAB provides guaranteed delivery of messages to all cluster systems. The Atomic Broadcast functionality is used by HAD to ensure that all systems within the cluster receive all configuration change messages, or are rolled back to the previous state, much like a database atomic commit. While the communications function in GAB is known as Atomic Broadcast, no actual network broadcast traffic is generated. An Atomic Broadcast message is a series of point to point unicast messages from the sending system to each receiving system, with a corresponding acknowledgement from each receiving system.

Low Latency Transport (LLT)

The Low Latency Transport protocol is used for all cluster communications as a high-performance, low-latency replacement for the IP stack. LLT has two major functions.

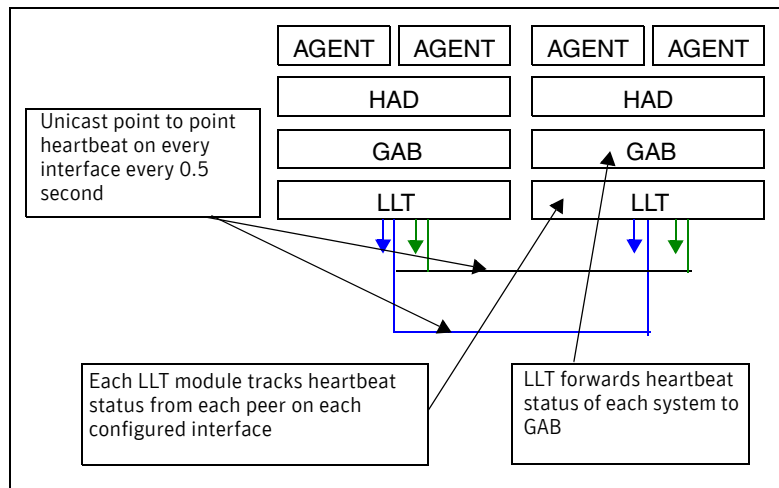
- **Traffic distribution**
LLT provides the communications backbone for GAB. LLT distributes (load balances) inter-system communication across all configured network links. This distribution ensures all cluster communications are evenly distributed across all network links for performance and fault resilience. If a link fails, traffic is redirected to the remaining links. A maximum of eight network links are supported.
- **Heartbeat**
LLT is responsible for sending and receiving heartbeat traffic over each configured network link. The heartbeat traffic is point to point unicast. LLT uses ethernet broadcast to learn the address of the nodes in the cluster. All other cluster communications, including all status and configuration traffic

is point to point unicast. The heartbeat is used by the Group Membership Services to determine cluster membership.

The heartbeat signal is defined as follows:

- LLT on each system in the cluster sends heartbeat packets out on all configured LLT interfaces every half second.
- LLT on each system tracks the heartbeat status from each peer on each configured LLT interface.
- LLT on each system forwards the heartbeat status of each system in the cluster to the local Group Membership Services function of GAB.
- GAB receives the status of heartbeat from all cluster systems from LLT and makes membership determination based on this information.

Figure 9-6 Heartbeat in the cluster



LLT can be configured to designate specific cluster interconnect links as either high priority or low priority. High priority links are used for cluster communications to GAB as well as heartbeat signals. Low priority links, during normal operation, are used for heartbeat and link state maintenance only, and the frequency of heartbeats is reduced to 50% of normal to reduce network overhead.

If there is a failure of all configured high priority links, LLT will switch all cluster communications traffic to the first available low priority link. Communication traffic will revert back to the high priority links as soon as they become available.

While not required, best practice recommends to configure at least one low priority link, and to configure two high priority links on dedicated cluster interconnects to provide redundancy in the communications path. Low priority links are typically configured on the public or administrative network.

About cluster membership

The current members of the cluster are the systems that are actively participating in the cluster. It is critical for HAD to accurately determine current cluster membership in order to take corrective action on system failure and maintain overall cluster topology.

A change in cluster membership is one of the starting points of the logic to determine if HAD needs to perform any fault handling in the cluster.

There are two aspects to cluster membership, initial joining of the cluster and how membership is determined once the cluster is up and running.

Initial joining of systems to cluster membership

When the cluster initially boots, LLT determines which systems are sending heartbeat signals, and passes that information to GAB. GAB uses this information in the process of seeding the cluster membership.

Seeding a new cluster

Seeding insures a new cluster will start with an accurate membership count of the number of systems in the cluster. This prevents the possibility of one cluster splitting into multiple subclusters upon initial startup. A new cluster can be automatically seeded as follows:

- When the cluster initially boots, all systems in the cluster are unseeded.
- GAB checks the number of systems that have been declared to be members of the cluster in the `/etc/gabtab` file.

The number of systems declared in the cluster is denoted as follows:

```
/sbin/gabconfig -c -n#
```

where the variable `#` is replaced with the number of systems in the cluster.

Note: Symantec recommends that you replace `#` with the exact number of nodes in the cluster.

- When GAB on each system detects that the correct number of systems are running, based on the number declared in `/etc/gabtab` and input from LLT, it will seed.

- HAD will start on each seeded system. HAD will only run on a system that has seeded.
HAD can provide the HA functionality only when GAB has seeded.

Manual seeding of a cluster

Seeding the cluster manually is appropriate when the number of cluster systems declared in `/etc/gabtab` is more than the number of systems that will join the cluster. This could occur if a system is down for maintenance when the cluster comes up.

Caution: It is not recommended to seed the cluster manually unless the administrator is aware of the risks and implications of the command.

Before manually seeding the cluster, check that systems that will join the cluster are able to send and receive heartbeats to each other. Confirm there is no possibility of a network partition condition in the cluster.

Before manually seeding the cluster, do the following:

- Check that systems that will join the cluster are able to send and receive heartbeats to each other.
- Confirm there is no possibility of a network partition condition in the cluster.

To manually seed the cluster, type the following command:

```
/sbin/gabconfig -c -x
```

Note there is no declaration of the number of systems in the cluster with a manual seed. This command will seed all systems in communication with the system where the command is run.

So, make sure not to run this command in more than one node in the cluster

See “[Seeding and I/O fencing](#)” on page 589.

Ongoing cluster membership

Once the cluster is up and running, a system remains an active member of the cluster as long as peer systems receive a heartbeat signal from that system over the cluster interconnect. A change in cluster membership is determined as follows:

- When LLT on a system no longer receives heartbeat messages from a system on any of the configured LLT interfaces for a predefined time, LLT informs GAB of the heartbeat loss from that specific system.
This predefined time is 16 seconds by default, but can be configured. It is set with the `set-timer peerinact` command as described in the `llttab` manual page.
- When LLT informs GAB of a heartbeat loss, the systems that are remaining in the cluster coordinate to agree which systems are still actively participating in the cluster and which are not. This happens during a time period known as GAB Stable Timeout (5 seconds).
VCS has specific error handling that takes effect in the case where the systems do not agree.
- GAB marks the system as DOWN, excludes the system from the cluster membership, and delivers the membership change to the fencing module.
- The fencing module performs membership arbitration to ensure that there is not a split brain situation and only one functional cohesive cluster continues to run.

The fencing module is turned on by default.

See “[About cluster membership and data protection without I/O fencing](#)” for actions that occur if the fencing module has been deactivated.

About membership arbitration

Membership arbitration is necessary on a perceived membership change because systems may falsely appear to be down. When LLT on a system no longer receives heartbeat messages from another system on any configured LLT interface, GAB marks the system as DOWN. However, if the cluster interconnect network failed, a system can appear to be failed when it actually is not. In most environments when this happens, it is caused by an insufficient cluster interconnect network infrastructure, usually one that routes all communication links through a single point of failure.

If all the cluster interconnect links fail, it is possible for one cluster to separate into two subclusters, each of which does not know about the other subcluster. The two subclusters could each carry out recovery actions for the departed systems. This is termed split brain.

In a split brain condition, two systems could try to import the same storage and cause data corruption, have an IP address up in two places, or mistakenly run an application in two places at once.

Membership arbitration guarantees against such split brain conditions.

Components of membership arbitration

The components of membership arbitration are the fencing module and the coordination points.

Fencing module

Each system in the cluster runs a kernel module called vxfen, or the *fencing module*. This module is responsible for ensuring valid and current cluster membership on a membership change through the process of membership arbitration. vxfen performs the following actions:

- Registers with the coordinator disks during normal operation
- Races for control of the coordinator disks during membership changes

Coordination points

Coordination points provide a lock mechanism to determine which nodes get to fence off data drives from other nodes. A node must eject a peer from the coordination points before it can fence the peer from the data drives. Racing for control of the coordination points to fence data disks is the key to understand how fencing prevents split brain.

The coordination points can either be disks or servers or both. Typically, a cluster must have three coordination points.

- **Coordinator disks**

Disks that act as coordination points are called coordinator disks. Coordinator disks are three standard disks or LUNs set aside for I/O fencing during cluster reconfiguration. Coordinator disks do not serve any other storage purpose in the VCS configuration.

You can configure I/O fencing to use either the DMP devices or the underlying raw character devices. Veritas Volume Manager Dynamic Multipathing (DMP) feature allows coordinator disks to take advantage of the path failover and the dynamic adding and removal capabilities of DMP. Based on the disk device type that you use, you must define the I/O fencing SCSI-3 disk policy as either raw or dmp. The disk policy is dmp by default. See the *Veritas Volume Manager Administrator's Guide* for details on the DMP feature.
- **Coordination point servers**

The coordination point server (CP server) is a software solution which runs on a remote system or cluster. CP server provides arbitration functionality by allowing the VCS cluster nodes to perform the following tasks:

 - Self-register to become a member of an active VCS cluster (registered with CP server with access to the data drives)
 - Check which other nodes are registered as members of this active VCS cluster
 - Self-unregister from this active VCS cluster
 - Forcefully unregister other nodes (preempt) as members of this active VCS cluster

Note: With the CP server, the fencing arbitration logic still remains on the VCS cluster.

Multiple VCS clusters running different operating systems can simultaneously access the CP server. TCP/IP based communication is used between the CP server and the VCS clusters.

How the fencing module starts up

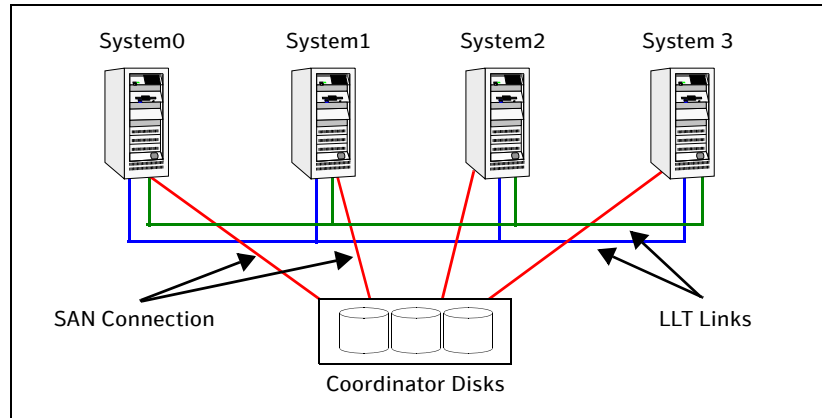
The fencing module starts up as follows:

- The coordinator disks are placed in a disk group.

This allows the fencing start up script to use Veritas Volume Manager (VxVM) commands to easily determine which disks are coordinator disks, and what paths exist to those disks. This disk group is never imported, and is not used for any other purpose.

- The fencing start up script on each system uses VxVM commands to populate the file `/etc/vxfentab` with the paths available to the coordinator disks.
 For example, if the user has configured 3 coordinator disks with 2 paths to each disk, the `/etc/vxfentab` file will contain 6 individual lines, representing the path name to each disk, such as
`/dev/rhdisk74`
- When the fencing driver is started, it reads the physical disk names from the `/etc/vxfentab` file. Using these physical disk names, it determines the serial numbers of the coordinator disks and builds an in-memory list of the drives.
- The fencing driver verifies that any other systems in the cluster that are already up and running see the same coordinator disks.
 The fencing driver examines GAB port B for membership information. If no other systems are up and running, it is the first system up and is considered the correct coordinator disk configuration. When a new member joins, it requests a coordinator disks configuration. The system with the lowest LLT ID will respond with a list of the coordinator disk serial numbers. If there is a match, the new member joins the cluster. If there is not a match, `vxfen` enters an error state and the new member is not allowed to join. This process ensures all systems communicate with the same coordinator disks.
- The fencing driver determines if a possible preexisting split brain condition exists.
 This is done by verifying that any system that has keys on the coordinator disks can also be seen in the current GAB membership. If this verification fails, the fencing driver prints a warning to the console and system log and does not start.
- If all verifications pass, the fencing driver on each system registers keys with each coordinator disk.

Figure 9-7 Topology of coordinator disks in the cluster



How membership arbitration works

Upon startup of the cluster, all systems register a unique key on the coordinator disks. The key is unique to the cluster and the node, and is based on the LLT cluster ID and the LLT system ID.

See [“About the I/O fencing registration key format”](#) on page 336.

When there is a perceived change in membership, membership arbitration works as follows:

- GAB marks the system as DOWN, excludes the system from the cluster membership, and delivers the membership change—the list of departed systems—to the fencing module.
- The system with the lowest LLT system ID in the cluster races for control of the coordinator disks
 - In the most common case, where departed systems are truly down or faulted, this race has only one contestant.
 - In a split brain scenario, where two or more subclusters have formed, the race for the coordinator disks is performed by the system with the lowest LLT system ID of that subcluster. This system races on behalf of all the other systems in its subcluster.
- The race consists of executing a preempt and abort command for each key of each system that appears to no longer be in the GAB membership. The preempt and abort command allows only a registered system with a valid key to eject the key of another system. This ensures that even when multiple systems attempt to eject other, each race will have only one winner. The first system to issue a preempt and abort command will win

and eject the key of the other system. When the second system issues a preempt and abort command, it can not perform the key eject because it is no longer a registered system with a valid key.

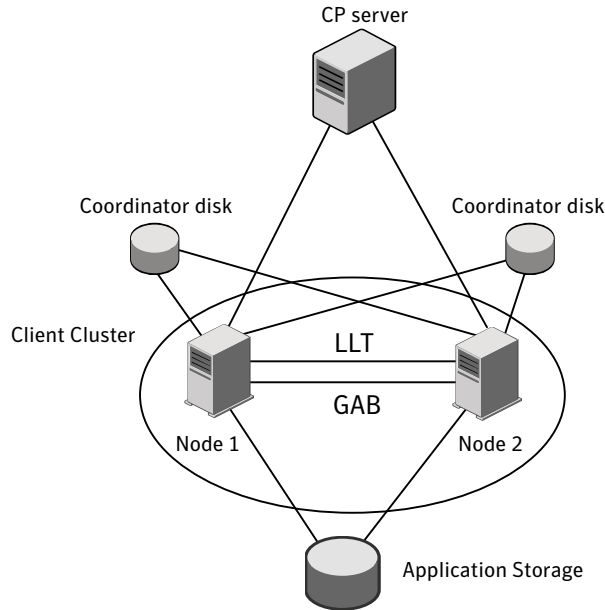
- If the preempt and abort command returns success, that system has won the race for that coordinator disk.
 Each system will repeat this race to all the coordinator disks. The race is won by, and control is attained by, the system that ejects the other system's registration keys from a majority of the coordinator disks.
- On the system that wins the race, the vxfen module informs all the systems that it was racing on behalf of that it won the race, and that subcluster is still valid.
- On the system(s) that do not win the race, the vxfen module will trigger a system panic. The other systems in this subcluster will note the panic, determine they lost control of the coordinator disks, and also panic and restart.
- Upon restart, the systems will attempt to seed into the cluster.
 - If the systems that restart can exchange heartbeat with the number of cluster systems declared in `/etc/gabtab`, they will automatically seed and continue to join the cluster. Their keys will be replaced on the coordinator disks. This case will only happen if the original reason for the membership change has cleared during the restart.
 - If the systems that restart can not exchange heartbeat with the number of cluster systems declared in `/etc/gabtab`, they will not automatically seed, and HAD will not start. This is a possible split brain condition, and requires administrative intervention.

Note: Forcing a manual seed at this point will allow the cluster to seed. However, when the fencing module checks the GAB membership against the systems that have keys on the coordinator disks, a mismatch will occur. vxfen will detect a possible split brain condition, print a warning, and will not start. In turn, HAD will not start. Administrative intervention is required.

About server-based I/O fencing

In a disk-based I/O fencing implementation, the vxfen driver handles various SCSI-3 PR based arbitration operations completely within the driver. I/O fencing also provides a framework referred to as customized fencing wherein arbitration operations are implemented in custom scripts. The vxfen driver invokes the custom scripts.

The CP server-based coordination point uses a customized fencing framework. Note that SCSI-3PR based fencing arbitration can also be enabled using customized fencing framework. This allows the user to specify a combination of SCSI-3 LUNs and CP servers as coordination points using customized fencing. Customized fencing can be enabled by specifying `vxfen_mode=customized` and `vxfen_mechanism=cps` in the `/etc/vxfenmode` file.



A user level daemon `vxfsend` interacts with the `vxfsen` driver, which in turn interacts with GAB to get the node membership update. Upon receiving membership updates, `vxfsend` invokes various scripts to race for the coordination point and fence off data disks. The `vxfsend` daemon manages various fencing agents. The customized fencing scripts are located in the `/opt/VRTSvcs/vxfen/bin/customized/cps` directory. The scripts that are involved include the following:

- `generate_snapshot.sh`: Retrieves the SCSI ID's of the coordinator disks and/or UUID ID's of the CP servers
- `join_local_node.sh`: Registers the keys with the coordinator disks or CP servers
- `race_for_coordination_point.sh`: Races to determine a winner after cluster reconfiguration

- `fence_data.sh`: Protects the data disks by removing the SCSI keys of the panicked or leaving nodes
- `unjoin_local_node.sh`: Removes the keys that are registered in `join_local_node.sh`

I/O fencing enhancements provided by CP server

CP server configurations enhance disk-based I/O fencing by providing the following new capabilities:

- CP server configurations are scalable, and a configuration with three CP servers can provide I/O fencing for multiple VCS clusters. Since a single CP server configuration can serve a large number of VCS clusters, the cost of multiple VCS cluster deployments can be significantly reduced.
- Appropriately situated CP servers can eliminate any coordinator disk location bias in the I/O fencing process. For example, this location bias may occur where due to logistical restrictions two of the three coordinator disks are located at a single site, and the cost of setting up a third coordinator disk location is prohibitive. In such a configuration, if the site with two coordinator disks is inaccessible, the other site does not survive due to a lack of a majority of coordination points. Disk-based I/O fencing would require extension of the SAN to the third site which may not be a suitable solution. An alternative is to place a CP server at a remote site as the third coordination point.

Note: The CP server provides an alternative arbitration mechanism without having to depend on SCSI-3 compliant coordinator disks. Data disk fencing in CVM will still require SCSI-3 I/O fencing.

About making CP server highly available

If you want to configure a multi-node CP server cluster, install and configure SFHA on the CP server nodes. Otherwise, install and configure VCS on the single node.

In both the configurations, VCS provides local start and stop of the CP server process, taking care of dependencies such as NIC, IP address, etc. Moreover, VCS also serves to restart the CP server process in case the process faults.

To make the CP server process highly available, you must perform the following tasks:

- Install and configure SFHA on the CP server systems.
- Configure the CP server process as a failover service group.

- Configure disk-based I/O fencing to protect the shared CP server database. Symantec recommends that you do not run any other applications on the single node or SFHA cluster that is used to host CP server. A single CP server can serve multiple VCS clusters. A common set of CP servers can serve up to 128 VCS clusters

About the CP server database

CP server requires a database for storing the registration keys of the VCS cluster nodes. CP server uses a SQLite database for its operations. By default, the database is located at `/etc/VRTScps/db`.

For a single node VCS cluster hosting a CP server, the database can be placed on a local file system. For an SFHA cluster hosting a CP server, the database must be placed on a shared file system. The file system must be shared among all nodes that are part of the cluster.

In an SFHA cluster hosting the CP server, the shared database is protected by setting up SCSI-3 PR based I/O fencing. SCSI-3 PR based I/O fencing protects against split-brain scenarios.

Warning: The CP server database must not be edited directly and should only be accessed using `cpsadm(1M)`. Manipulating the database manually may lead to undesirable results including system panics.

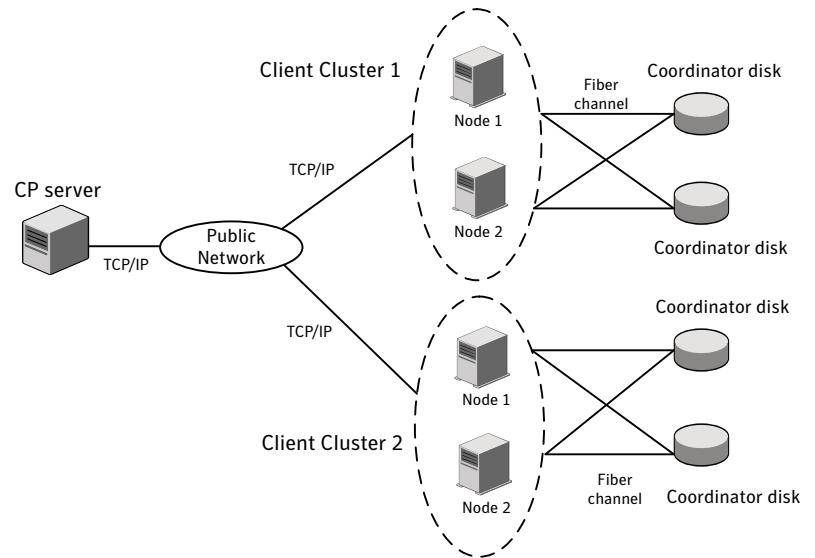
Recommended CP server configurations

This section discusses the following recommended CP server configurations:

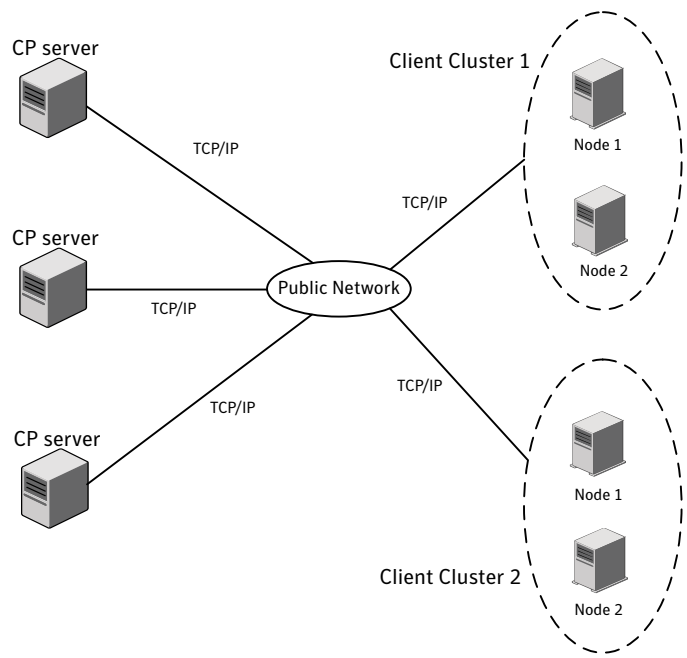
- A CP server configuration where multiple VCS clusters use 3 CP servers as their coordination points
- A CP server configuration where multiple VCS clusters use a single CP server and multiple pairs of coordinator disks (2) as their coordination points

Although the recommended CP server configurations use three coordination points, three or more odd number of coordination points may be used for I/O fencing. In a configuration where multiple VCS clusters share a common set of CP server coordination points, the VCS cluster as well as the CP server use a Universally Unique Identifier (UUID) to uniquely identify a VCS cluster.

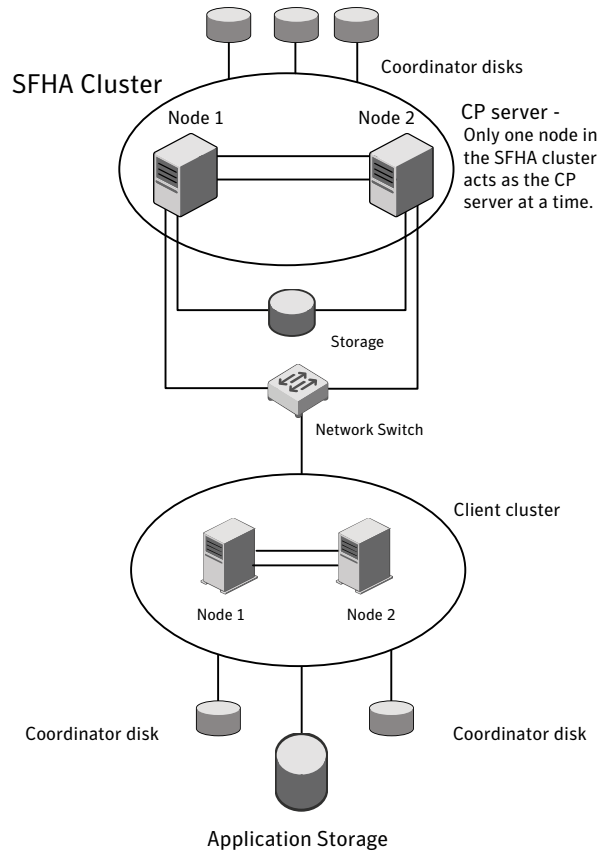
The following figure displays a configuration using a single CP server that is connected to multiple VCS clusters (referred to as client clusters in the illustration) with each VCS cluster also using two coordinator disks.



The following figure displays a configuration using 3 CP servers that are connected to multiple VCS clusters.



The following figure displays a schematic of a highly available CP server cluster where SFHA is installed and configured.



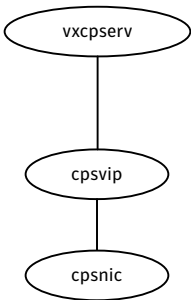
Symantec recommends that the CP server be made highly available in your configuration. The requirements for this configuration are the same as for that of a basic two node SFHA cluster.

As displayed in the figure, the CP server database must be persistently stored on shared storage in case of a CP server failure.

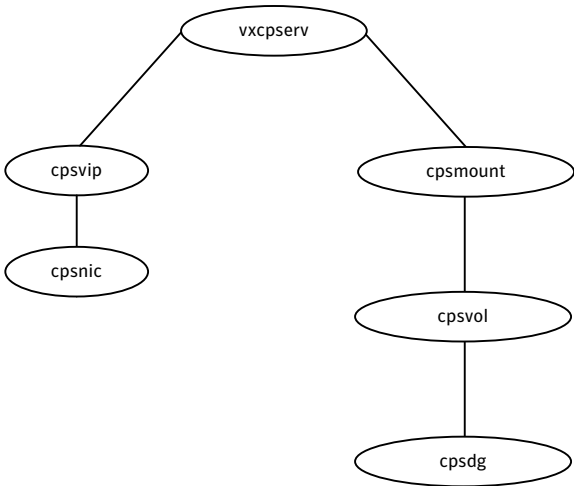
About the CP server service group

The CP server service group is a VCS service group. An I/O fencing configuration using CP server requires the CP server service group.

The following illustration displays a schematic of the service group and its dependencies when CP server is hosted on a single node VCS cluster.



The following illustration displays a schematic of the CPS service group and its dependencies when the CP server is hosted on an SFHA cluster.



About CP server user types and privileges

The CP server supports the following user types, each with a different access level privilege:

- CP server administrator (admin)
- CP server operator

Different access level privileges permit the user to issue different commands. If a user is neither a CP server admin nor a CP server operator user, then the user has guest status and can issue limited commands.

The user types and their access level privileges are assigned to individual users during VCS cluster configuration for fencing. During the installation process, you are prompted for a user name, password, and access level privilege (CP server admin or CP server operator).

To administer and operate a CP server, there must be at least one CP server admin.

A root user on a CP server is given all the administrator privileges and it can be used to perform all the CP server specific operations.

About secure communication between the VCS cluster and CP server

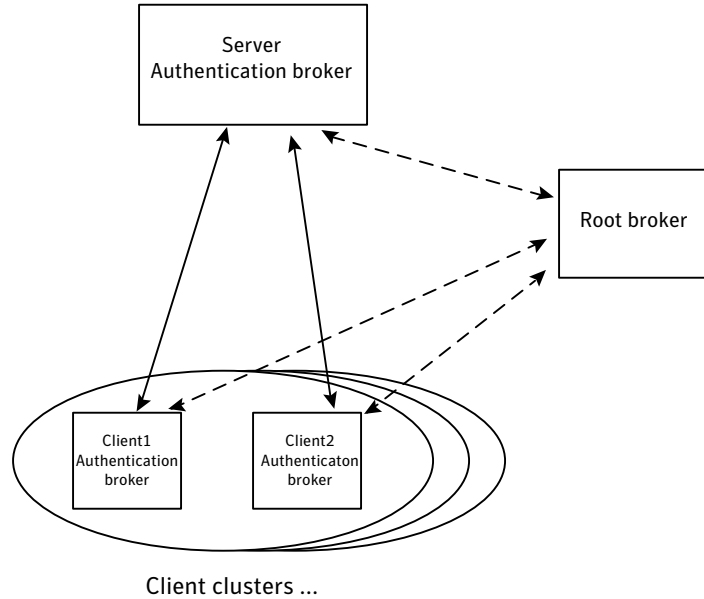
In a data center, TCP/IP communication between the VCS cluster and CP server must be made secure. The security of the communication channel involves encryption, authentication, and authorization.

The CP server node or cluster needs to confirm the authenticity of the VCS cluster nodes that communicate with it as a coordination point and only accept requests from known VCS cluster nodes. Requests from unknown clients are rejected as non-authenticated. Similarly, the fencing framework in VCS cluster must confirm that authentic users are conducting fencing operations with the CP server.

The encryption and authentication service for CP server is provided by Symantec Product Authentication Service. To enable Symantec™ Product Authentication Service, the VRTSat package is installed on the VCS clusters as well as CP server, as a part of VCS product installation.

The following figure displays a schematic of secure communication between the VCS cluster and CP server. An authentication broker is configured on CP server

and each VCS cluster node which authenticates clients such as users or services, and grants them a product credential.



Entities on behalf of which authentication is done, are referred to as principals. On the VCS cluster nodes (referred to as client clusters in this illustration), the current VCS installer creates the Authentication Server credentials on each node in the cluster, creates Web credentials for VCS users, and then sets up trust with the root broker. It also creates a VCS service group for the authentication broker. The installer then proceeds to start VCS in secure mode.

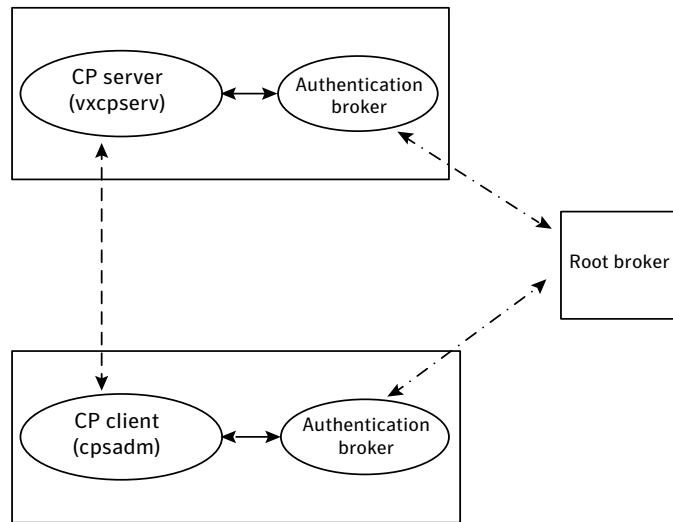
Typically, in an existing VCS cluster with security configured, a root broker would already have been configured and an authentication broker will be running on each cluster node.

How secure communication between CP servers and VCS clusters work

CP server and VCS cluster node communication involve the following entities:

- vxcpserv for the CP server
- cpsadm for the VCS cluster node

The following figure displays a schematic of the end-to-end communication flow with security enabled on CP server and VCS clusters.



Client cluster nodes

Communication flow between CP server and VCS cluster nodes with security configured on them is as follows:

- Initial setup

Identities of authentication brokers configured on CP server, as well as VCS cluster nodes are configured in the root broker's authentication private domain repository.

Note: If authentication brokers configured on CP server and VCS cluster nodes do not use the same root broker, then a trust should be established between the root brokers or authentication brokers, so that vxcperv process can authenticate requests from the VCS cluster nodes.

The cpsadm command gets the user name, domain type from the environment variables CPS_USERNAME, CPS_DOMAINTYPE. The user is expected to export these variables before running the cpsadm command manually. The customized fencing framework exports these environment variables internally before running the cpsadm commands.

The CP server process (vxcperv) uses its own user (_CPS_SERVER_) which is added to the local authentication broker during server startup.

- Getting credentials from authentication broker

The `cpsadm` command tries to get the existing credentials from authentication broker running on the local node. If this fails, it tries to authenticate itself with the local authentication broker.

The `vxcperv` process tries to get the existing credentials from authentication broker running on the local node. If this fails, it tries to authenticate itself with the local authentication broker and creates a principal for itself.

- Communication between CP server and VCS cluster nodes
Once the CP server is up after establishing its credential, it becomes ready to receive data from the clients. Once authenticated with the local authentication broker, `cpsadm` connects to the CP server. Data is passed over to the CP server.
- Validation
On receiving data from a particular VCS cluster node, `vxcperv` validates its credentials by consulting the local authentication broker. If validation fails, then the connection request data is rejected.

Security configuration details on CP server and VCS cluster: Settings in secure mode

If security is configured for communication between CP server and VCS cluster, the following are the settings:

- Settings on CP server:
A user gets created in the local authentication broker during CP server startup with the following values:
`username: _CPS_SERVER_`
`domainname: _CPS_SERVER_DOMAIN@FQHN`
`domaintype: vx`
where, FQHN is Fully Qualified Host Name of the client node
It can be verified using the following command on the CP server:
`/opt/VRTScps/bin/cpsat showcred`

Note: Note: The CP server configuration file (`/etc/vxcps.conf`) must not contain a line specifying `security=0`. If there is no line specifying “security” parameter or if there is a line specifying `security=1`, CP server with security is enabled (which is the default).

- Settings On VCS cluster nodes:
On VCS cluster, a user gets created for each cluster node in the local authentication broker during VCS security configuration with the following values:
`username: _HA_VCS_hostname`


```
domainname: HA_SERVICES@FQHN
domaintype: vx
```

where, FQHN is Fully Qualified Host Name of the client node

It can be verified using the following command on the VCS cluster node(s):

```
# /opt/VRTScps/bin/cpsat showcred
```

Users mentioned above are used only for authentication for the communication between:

- CP server and authentication broker configured on it, and
 - VCS cluster nodes and authentication brokers configured on them
- For CP server's authorization, the following user gets created and used by customized fencing framework on the VCS cluster, if security is configured:

```
_HA_VCS_hostname@HA_SERVICES@FQHN
```

where, hostname is the client node name without qualification and FQHN is Fully Qualified Host Name of the client node.

For each VCS cluster node, this user must be registered on the CP server database before fencing starts on the VCS cluster nodes. This can be verified by issuing the following command:

```
# cpsadm -s cp_server -a list_users
```

The following is an example of the command output:

```
Username/Domain Type Cluster Name / UUID Role
_HA_VCS_galaxy@HA_SERVICES@galaxy.symantec.com/vx cluster1/
{f0735332-e3709c1c73b9} Operator
```

Note: The configuration file (/etc/vxfenmode) on each client node must not contain a line specifying security=0. If there is no line specifying “security” parameter or if there is a line specifying security=1, client node starts with security enabled (which is the default).

Security configuration details on CP server and VCS cluster: Settings in non-secure mode

In non-secure mode, only in-built authorization is provided on the CP server without asking for any passwords. No authentication and encryption are provided. User credentials of “cpsclient@hostname” of “vx” domain type are used by the customized fencing framework for communication between CP server or VCS cluster nodes.

For each VCS cluster node, this user must be added on the CP server database before fencing starts on the VCS cluster nodes. The user can be verified by issuing the following command:

```
# cpsadm -s cpserver -a list_users
```

The following is an example of the command output:

```
Username/Domain Type Cluster Name / UUID Role
```

```
cpsclient@galaxy/vx cluster1 / {f0735332-e3709c1c73b9} Operator
```

Note: In this mode, CP server configuration file (/etc/vxcps.conf) must contain a line specifying security=0. Similarly, on each VCS cluster node the configuration file (/etc/vxfenmode) must contain a line specifying security=0.

About the vxfenmode file

At the end of the CP server configuration process, a vxfenmode file is created and saved to the /etc directory. The vxfenmode file contains configuration information related to fencing.

The file name format is “vxfenmode- date and time stamp”.

This file can be used as reference, if the fencing process fails for future configurations. The following table depicts sample values for parameters in the vxfenmode file.

Mixed mode (disk and CP server)	Pure CP server mode
vxfen_mode=customized	vxfen_mode=customized
vxfen_mechanism=cps	vxfen_mechanism=cps
scsi3_disk_policy=dmp	Not applicable
security=1	security=1
cps1=[thunderbolt.symantecexample.com]:14250	cps1=[thunderbolt.symantecexample.com]:14250
vxfendg=vxfencoorddg	cps2=[.symantecexample.com]:14250 cps3=[.symantecexample.com]:14250

Whenever coordinator disks are being used in an I/O fencing configuration in customized mode, a disk group has to be created and specified in the /etc/vxfenmode file. Additionally, the customized fencing framework also generates the /etc/vxfentab file which specifies the security setting and the coordination points (all the CP servers and disks from the disk group specified in /etc/vxfenmode file).

About data protection

Membership arbitration by itself is inadequate for complete data protection because it assumes that all systems will either participate in the arbitration or are already down.

Rare situations can arise which must also be protected against. Some examples are:

- A system hang causes the kernel to stop processing for a period of time.
- The system resources were so busy that the heartbeat signal was not sent.
- A break and resume function is supported by the hardware and executed. Dropping the system to a system controller level with a break command can result in the heartbeat signal timeout.

In these types of situations, the systems are not actually down, and may return to the cluster after cluster membership has been recalculated. This could result in data corruption as a system could potentially write to disk before it determines it should no longer be in the cluster.

Combining membership arbitration with data protection of the shared storage eliminates all of the above possibilities for data corruption.

Data protection fences off (removes access to) the shared data storage from any system that is not a current and verified member of the cluster. Access is blocked by the use of SCSI-3 persistent reservations.

SCSI-3 Persistent Reservation

SCSI-3 Persistent Reservation (SCSI-3 PR) supports device access from multiple systems, or from multiple paths from a single system. At the same time it blocks access to the device from other systems, or other paths.

VCS logic determines when to online a service group on a particular system. If the service group contains a disk group, the disk group is imported as part of the service group being brought online. When using SCSI-3 PR, importing the disk group puts registration and reservation on the data disks. Only the system that has imported the storage with SCSI-3 reservation can write to the shared storage. This prevents a system that did not participate in membership arbitration from corrupting the shared storage.

SCSI-3 PR ensures persistent reservations across SCSI bus resets.

Note: Use of SCSI 3 PR protects against all elements in the IT environment that might be trying to write illegally to storage, not only VCS related elements.

Membership arbitration combined with data protection is termed I/O fencing.

Examples of VCS operation with I/O fencing

This topic describes the general logic employed by the I/O fencing module along with some specific example scenarios.

About the I/O fencing algorithm

To ensure the most appropriate behavior is followed in both common and rare corner case events, the fencing algorithm works as follows:

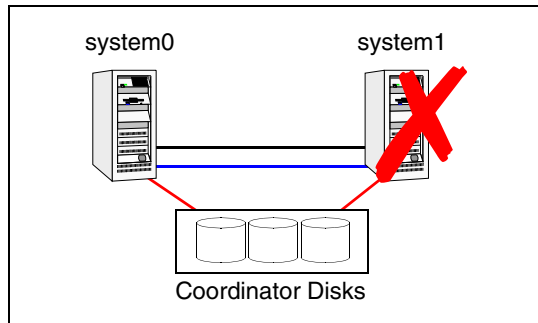
- The fencing module is designed to never have systems in more than one subcluster remain current and valid members of the cluster. In all cases, either one subcluster will survive, or in very rare cases, no systems will.
- The system with the lowest LLT ID in any subcluster of the original cluster races for control of the coordinator disks on behalf of the other systems in that subcluster.
- If a system wins the race for the first coordinator disk, that system is given priority to win the race for the other coordinator disks.
Any system that loses a race will delay a short period of time before racing for the next disk. Under normal circumstances, the winner of the race to the first coordinator disk will win all disks.
This ensures a clear winner when multiple systems race for the coordinator disk, preventing the case where three or more systems each win the race for one coordinator disk.
- If the cluster splits such that one of the subclusters has at least 51% of the members of the previous stable membership, that subcluster is given priority to win the race.
The system in the smaller subcluster(s) delay a short period before beginning the race.
This ensures that as many systems as possible will remain running in the cluster.
- If the vxfen module discovers on startup that the system that has control of the coordinator disks is not in the current GAB membership, an error message indicating a possible split brain condition is printed to the console. The administrator must clear this condition manually with the `vxfenclearpre` utility.

Two system cluster where one system fails

In this example, System1 fails, and System0 carries out the I/O fencing operation as follows:

- The GAB module on System0 determines System1 has failed due to loss of heartbeat signal reported from LLT.
- GAB passes the membership change to the fencing module on each system in the cluster.
The only system that is still running is System0
- System0 gains control of the coordinator disks by ejecting the key registered by System1 from each coordinator disk.
The ejection takes place one by one, in the order of the coordinator disk's serial number.
- When the fencing module on System0 successfully controls the coordinator disks, HAD carries out any associated policy connected with the membership change.
- System1 is blocked access to the shared storage, if this shared storage was configured in a service group that was now taken over by System0 and imported.

Figure 9-8 I/O Fencing example with system failure

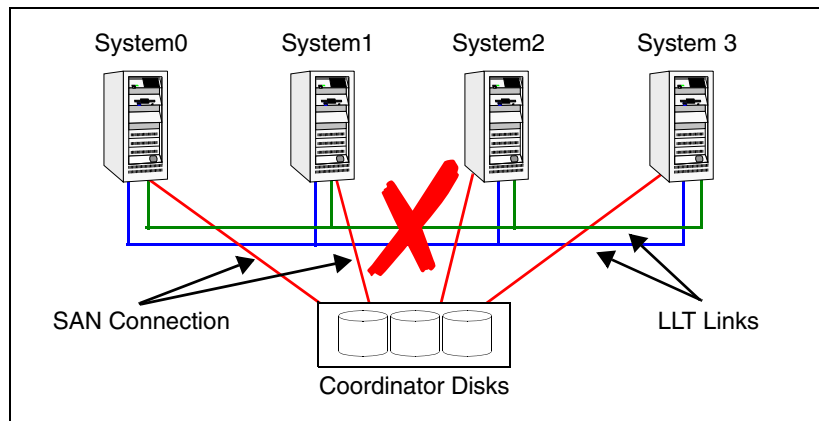


Four system cluster where cluster interconnect fails

In this example, the cluster interconnect fails in such a way as to split the cluster from one four-system cluster to two-system clusters. The cluster performs membership arbitration to ensure that only one subcluster remains.

Due to loss of heartbeat, System0 and System1 both believe System2 and System3 are down. System2 and System3 both believe System0 and System1 are down. The progression of I/O fencing operations are as follows:

Figure 9-9 Four system cluster where cluster interconnect fails



- LLT on each of the four systems no longer receives heartbeat messages from the systems on the other side of the interconnect failure on any of the configured LLT interfaces for the peer inactive timeout configured time.
- LLT on each machine passes to GAB that it has noticed a membership change. Specifically:
 - LLT on System0 passes to GAB that it no longer sees System2 and System3
 - LLT on System1 passes to GAB that it no longer sees System2 and System3
 - LLT on System2 passes to GAB that it no longer sees System0 and System1
 - LLT on System3 passes to GAB that it no longer sees System0 and System1
- After LLT informs GAB of a heartbeat loss, the systems that are remaining do a “GAB Stable Timeout (5 seconds). In this example:

- System0 and System1 agree that both of them do not see System2 and System3
- System2 and System3 agree that both of them do not see System0 and System1
- GAB marks the system as DOWN, and excludes the system from the cluster membership. In this example:
 - GAB on System0 and System1 mark System2 and System3 as DOWN and excludes them from cluster membership.
 - GAB on System2 and System3 mark System0 and System1 as DOWN and excludes them from cluster membership.
- GAB on each of the four systems passes the membership change to the vxfs driver for membership arbitration. Each subcluster races for control of the coordinator disks. In this example:
 - System0 has the lower LLT ID, and races on behalf of itself and System1.
 - System2 has the lower LLT ID, and races on behalf of itself and System3.
- GAB on each of the four systems also passes the membership change to HAD. HAD waits for the result of the membership arbitration from the fencing module before taking any further action.
- Assume System0 wins the race for the coordinator disks, and ejects the registration keys of System2 and System3 off the disks. The result is as follows:
 - System0 wins the race for the coordinator disk. The fencing module on System0 communicates race success to all other fencing modules in the current cluster, in this case System0 and System1. The fencing module on each system in turn communicates success to HAD. System0 and System1 remain valid and current members of the cluster.
 - System2 loses the race for control of the coordinator disks. The fencing module on System2 calls a kernel panic and the system restarts.
 - System3 sees another membership change from the kernel panic of System2. Because that was the system that was racing for control of the coordinator disks in this subcluster, System3 also panics.
- HAD carries out any associated policy or recovery actions based on the membership change.
- System2 and System3 are blocked access to the shared storage (if the shared storage was part of a service group that is now taken over by System0 or System 1).

Examples of VCS operation with I/O fencing

- To rejoin System2 and System3 to the cluster, the administrator must do the following:
 - Shut down System2 and System3
 - Fix the cluster interconnect links
 - Restart System2 and System3

How disk-based I/O fencing works in different event scenarios

Table 9-1 describes how I/O fencing works to prevent data corruption in different failure event scenarios. For each event, corrective operator actions are indicated.

Table 9-1 I/O fencing scenarios

Event	Node A: What happens?	Node B: What happens?	Operator action
Both private networks fail.	Node A races for majority of coordinator disks. If Node A wins race for coordinator disks, Node A ejects Node B from the shared disks and continues.	Node B races for majority of coordinator disks. If Node B loses the race for the coordinator disks, Node B removes itself from the cluster.	When Node B is ejected from cluster, repair the private networks before attempting to bring Node B back.
Both private networks function again after event above.	Node A continues to work.	Node B has crashed. It cannot start the database since it is unable to write to the data disks.	Restart Node B after private networks are restored.
One private network fails.	Node A prints message about an IOFENCE on the console but continues.	Node B prints message about an IOFENCE on the console but continues.	Repair private network. After network is repaired, both nodes automatically use it.

Table 9-1 I/O fencing scenarios

Event	Node A: What happens?	Node B: What happens?	Operator action
Node A hangs.	<p>Node A is extremely busy for some reason or is in the kernel debugger.</p> <p>When Node A is no longer hung or in the kernel debugger, any queued writes to the data disks fail because Node A is ejected. When Node A receives message from GAB about being ejected, it removes itself from the cluster.</p>	<p>Node B loses heartbeats with Node A, and races for a majority of coordinator disks.</p> <p>Node B wins race for coordinator disks and ejects Node A from shared data disks.</p>	<p>Verify private networks function and restart Node A.</p>

Table 9-1 I/O fencing scenarios

Event	Node A: What happens?	Node B: What happens?	Operator action
<p>Nodes A and B and private networks lose power. Coordinator and data disks retain power.</p> <p>Power returns to nodes and they restart, but private networks still have no power.</p>	<p>Node A restarts and I/O fencing driver (vxfen) detects Node B is registered with coordinator disks. The driver does not see Node B listed as member of cluster because private networks are down. This causes the I/O fencing device driver to prevent Node A from joining the cluster. Node A console displays:</p> <p>Potentially a preexisting split brain. Dropping out of the cluster. Refer to the user documentation for steps required to clear preexisting split brain.</p>	<p>Node B restarts and I/O fencing driver (vxfen) detects Node A is registered with coordinator disks. The driver does not see Node A listed as member of cluster because private networks are down. This causes the I/O fencing device driver to prevent Node B from joining the cluster. Node B console displays:</p> <p>Potentially a preexisting split brain. Dropping out of the cluster. Refer to the user documentation for steps required to clear preexisting split brain.</p>	<p>Resolve preexisting split brain condition.</p> <p>See “System panics to prevent potential data corruption” on page 602.</p>

Table 9-1 I/O fencing scenarios

Event	Node A: What happens?	Node B: What happens?	Operator action
Node A crashes while Node B is down. Node B comes up and Node A is still down.	Node A is crashed.	Node B restarts and detects Node A is registered with the coordinator disks. The driver does not see Node A listed as member of the cluster. The I/O fencing device driver prints message on console: Potentially a preexisting split brain. Dropping out of the cluster. Refer to the user documentation for steps required to clear preexisting split brain.	Resolve preexisting split brain condition. See “System panics to prevent potential data corruption” on page 602.

Table 9-1 I/O fencing scenarios

Event	Node A: What happens?	Node B: What happens?	Operator action
The disk array containing two of the three coordinator disks is powered off.	Node A continues to operate as long as no nodes leave the cluster.	Node B continues to operate as long as no nodes leave the cluster.	
Node B leaves the cluster and the disk array is still powered off.	Node A races for a majority of coordinator disks. Node A fails because only one of three coordinator disks is available. Node A removes itself from the cluster.	Node B leaves the cluster.	Power on failed disk array and restart I/O fencing driver to enable Node A to register with all coordinator disks.

About cluster membership and data protection without I/O fencing

Proper seeding of the cluster and the use of low priority heartbeat cluster interconnect links are best practices with or without the use of I/O fencing. Best practice also recommends multiple cluster interconnect links between systems in the cluster. This allows GAB to differentiate between:

- A loss of all heartbeat links simultaneously, which is interpreted as a system failure. In this case, depending on failover configuration, HAD may attempt to restart the services that were running on that system on another system.
- A loss of all heartbeat links over time, which is interpreted as an interconnect failure. In this case, the assumption is made that there is a high probability that the system is not down, and HAD does not attempt to restart the services on another system.

In order for this differentiation to have meaning, it is important to ensure the cluster interconnect links do not have a single point of failure, such as a network hub or ethernet card.

About jeopardy

In all cases, when LLT on a system no longer receives heartbeat messages from another system on any of the configured LLT interfaces, GAB reports a change in membership.

When a system has only one interconnect link remaining to the cluster, GAB can no longer reliably discriminate between loss of a system and loss of the network. The reliability of the system's membership is considered at risk. A special membership category takes effect in this situation, called a jeopardy membership. This provides the best possible split-brain protection without membership arbitration and SCSI-3 capable devices.

When a system is placed in jeopardy membership status, two actions occur

- If the system loses the last interconnect link, VCS places service groups running on the system in autodisabled state. A service group in autodisabled state may failover on a resource or group fault, but can not fail over on a system fault until the autodisabled flag is manually cleared by the administrator.
- VCS operates the system as a single system cluster. Other systems in the cluster are partitioned off in a separate cluster membership.

About Daemon Down Node Alive (DDNA)

Daemon Down Node Alive (DDNA) is a condition in which the VCS high availability daemon (HAD) on a node fails, but the node is running. When HAD fails, the hashadow process tries to bring HAD up again. If the hashadow process succeeds in bringing HAD up, the system leaves the DDNA membership and joins the regular membership.

In a DDNA condition, VCS does not have information about the state of service groups on the node. So, VCS places all service groups that were online on the affected node in the autodisabled state. The service groups that were online on the node cannot fail over.

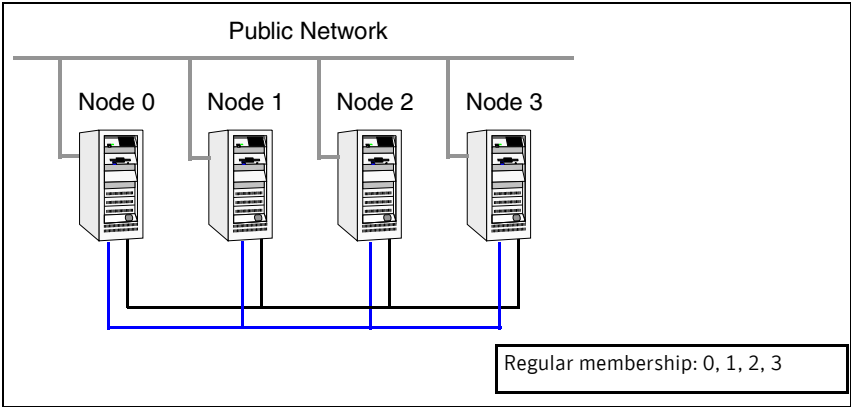
Manual intervention is required to enable failover of autodisabled service groups. The administrator must release the resources running on the affected node, clear resource faults, and bring the service groups online on another node. You can use the GAB registration monitoring feature to detect DDNA conditions. See “[Registration monitoring](#)” on page 572.

Examples of VCS operation without I/O fencing

The following scenarios describe events, and how VCS responds, in a cluster without I/O fencing.

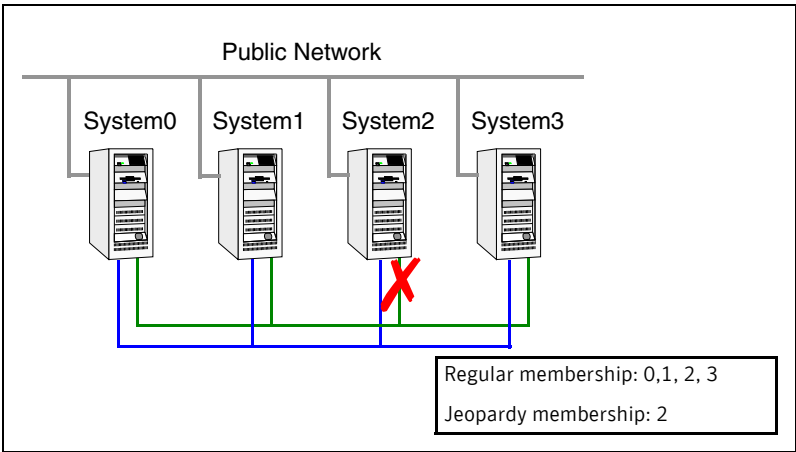
Four system cluster without a low priority link

Consider a four-system cluster that has two private cluster interconnect heartbeat links. The cluster does not have any low priority link.



Cluster interconnect link failure

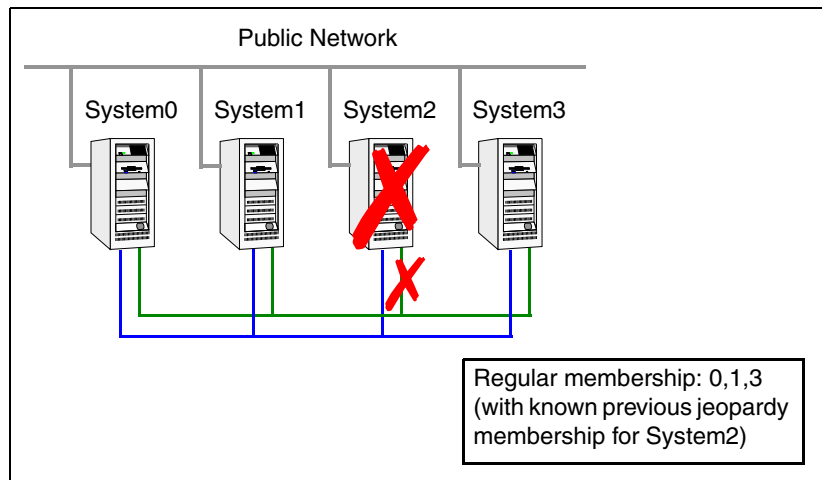
In this example, a link to System2 fails, leaving System2 with only one cluster interconnect link remaining.



The cluster is reformed. Systems 0, 1, 2, and 3 are in the regular membership and System2 in a jeopardy membership. Service groups on System2 are autodisabled. All normal cluster operations continue, including normal failover of service groups due to resource fault.

Cluster interconnect link failure followed by system failure

In this example, the link to System2 fails, and System2 is put in the jeopardy membership. Subsequently, System2 fails due to a power fault.

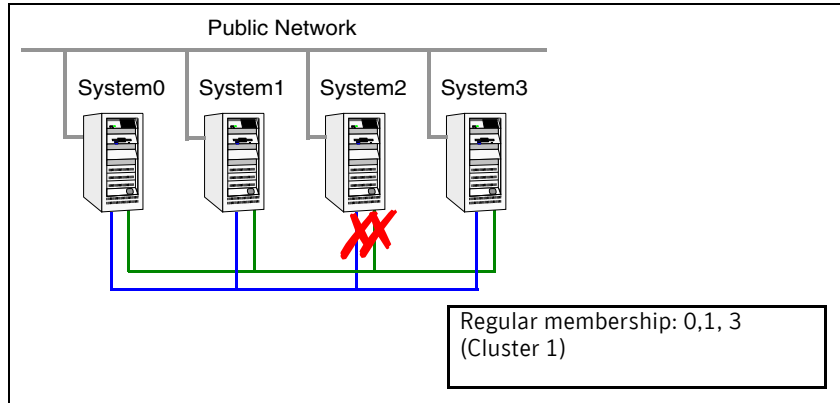


Systems 0, 1, and 3 recognize that System2 has faulted. The cluster is reformed. Systems 0, 1, and 3 are in a regular membership. When System2 went into jeopardy membership, service groups running on System2 were autodisabled. Even though the system is now completely failed, no other system can assume ownership of these service groups unless the system administrator manually clears the AutoDisabled flag on the service groups that were running on System2.

However, after the flag is cleared, these service groups can be manually brought online on other systems in the cluster.

All high priority cluster interconnect links fail

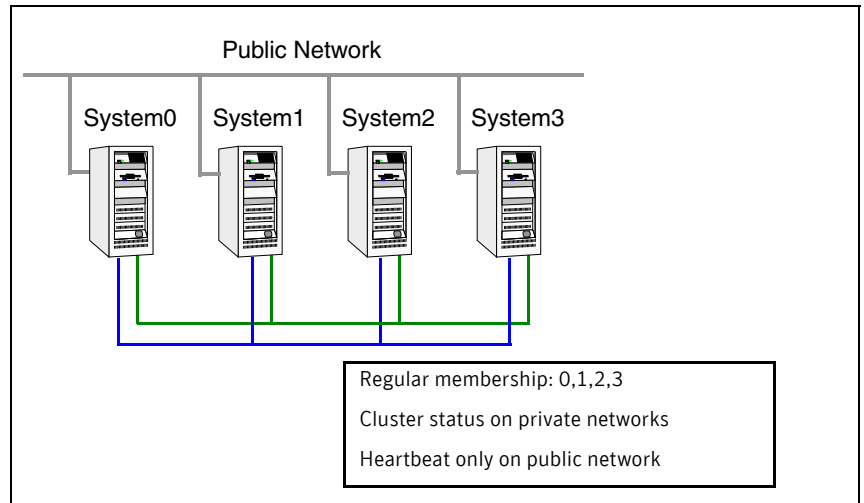
In this example, all high priority links to System2 fail. This can occur two ways:



- Both links to System2 fail at the same time
System2 was never in jeopardy membership. Without a low priority link, the cluster splits into two subclusters, where System0, 1 and 3 are in one subcluster, and System2 is in another. This is a split brain scenario.
- Both links to System2 fail at different times
System2 was in a jeopardy membership when the second link failed, and therefore the service groups that were online on System2 were autodisabled. No other system can online these service groups without administrator intervention.
Systems 0, 1 and 3 form a mini-cluster. System2 forms another single-system mini-cluster. All service groups that were present on systems 0, 1 and 3 are autodisabled on System2.

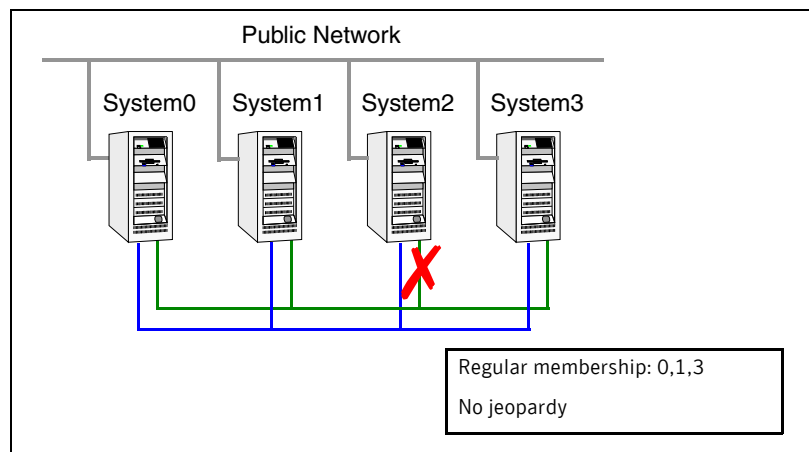
Four system cluster with low priority link

Consider a four-system cluster that has two private cluster interconnect heartbeat links, and one public low priority link.



Cluster interconnect link failure

In this example, a link to System2 fails, leaving System2 with one cluster interconnect link and the low priority link remaining.

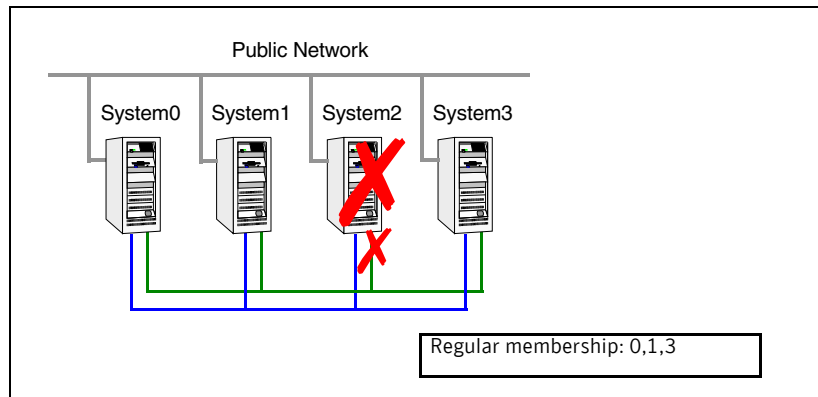


Other systems send all cluster status traffic to System2 over the remaining private link and use both private links for traffic between themselves. The low

priority link continues carrying the heartbeat signal only. No jeopardy condition is in effect because two links remain to determine system failure.

Cluster interconnect link failure followed by system failure

In this example, the link to System2 fails. Because there is a low priority heartbeat link, System2 is not put in the jeopardy membership. Subsequently, System2 fails due to a power fault.



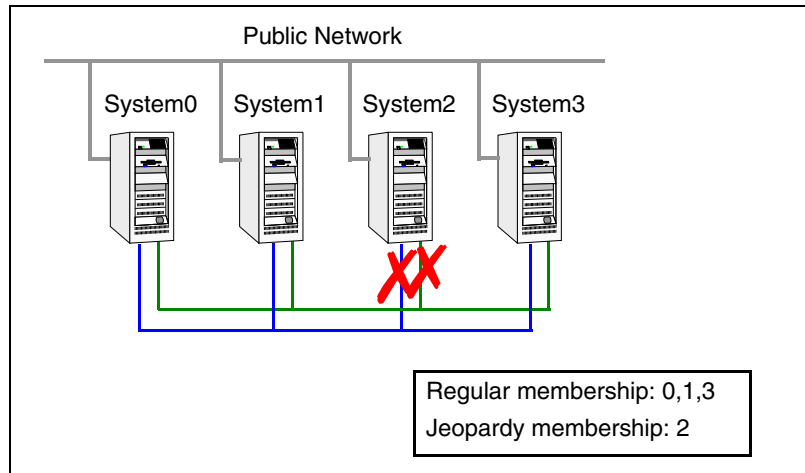
Systems 0, 1, and 3 recognize that System2 has faulted. The cluster is reformed. Systems 0, 1, and 3 are in a regular membership. The service groups on System2 that are configured for failover on system fault are attempted to be brought online on another target system, if one exists.

All high priority cluster interconnect links fail

In this example, both high priority cluster interconnect links to System2 fail, leaving System2 with only the low priority link remaining.

Cluster status communication is now routed over the low priority link to System2. System2 is placed in a jeopardy membership. The service groups on

System2 are autotdisabled, and the service group attribute AutoFailOver is set to 0, meaning the service group will not fail over on a system fault.



When a cluster interconnect link is re-established, all cluster status communications revert back to the cluster interconnect and the low priority link returns to sending heartbeat signal only. At this point, System2 is placed back in regular cluster membership.

Summary of best practices for cluster communications

The following are the recommended best practices for cluster communications to best support proper cluster membership and data protection.

- Properly seed the cluster by requiring all systems, and not just a subset of systems, to be present in the GAB membership before the cluster will automatically seed.
 If every system is not present, manual intervention by the administrator must eliminate the possibility of a split brain condition before manually seeding the cluster.
- Configure multiple independent communication network links between cluster systems.
 Networks should not have a single point of failure, such as a shared hub or ethernet card.
- The links for all nodes for LLT must be of the same link speed

- Low-priority LLT links in clusters with or without I/O fencing is recommended. In clusters without I/O fencing, this is critical.

Note: An exception to this is if the cluster uses fencing along with Cluster File Systems (CFS) or Oracle Real Application Clusters (RAC).

The reason for this is that low priority links are usually shared public network links. In the case where the main cluster interconnects fail, and the low priority link was the only remaining link, large amounts of data would be moved to the low priority link. This would potentially slow down the public network to unacceptable performance. Without a low priority link configured, membership arbitration would go into effect in this case, and some systems may be taken down, but the remaining systems would continue to run without impact to the public network.

It is not recommended to have a cluster with CFS or RAC without I/O fencing configured.

- Disable the console-abort sequence

Most UNIX systems provide a console-abort sequence that enables the administrator to halt and continue the processor. Continuing operations after the processor has stopped may corrupt data and is therefore unsupported by VCS.

When a system is halted with the abort sequence, it stops producing heartbeats. The other systems in the cluster consider the system failed and take over its services. If the system is later enabled with another console sequence, it continues writing to shared storage as before, even though its applications have been restarted on other systems.

Symantec recommends disabling the console-abort sequence or creating an alias to force the go command to perform a restart on systems not running I/O fencing.

- Select the smallest possible LUNs for use as coordinator disks. No more than three coordinator disks are needed in any configuration.
- Do not reconnect the cluster interconnect after a network partition without shutting down one side of the split cluster.

A common example of this happens during testing, where the administrator may disconnect the cluster interconnect and create a network partition. Depending on when the interconnect cables are reconnected, unexpected behavior can occur.

Administering I/O fencing

- [About administering I/O fencing](#)
- [About the vxfentsthdw utility](#)
- [About the vxfenadm utility](#)
- [About the vxfenclearpre utility](#)
- [About the vxfenswap utility](#)
- [Administering the coordination point server](#)

About administering I/O fencing

The I/O fencing feature provides the following utilities that are available through the VRTSvxfen package:

vxfcntlsthdw	Tests hardware for I/O fencing See “ About the vxfcntlsthdw utility ” on page 328.
vxfenconfig	Configures and unconfigures I/O fencing Checks the list of coordinator disks used by the vxfen driver.
vxfenadm	Displays information on I/O fencing operations and manages SCSI-3 disk registrations and reservations for I/O fencing See “ About the vxfenadm utility ” on page 336.
vxfenclearpre	Removes SCSI-3 registrations and reservations from disks See “ About the vxfenclearpre utility ” on page 340.
vxfenswap	Replaces coordinator disks without stopping I/O fencing See “ About the vxfenswap utility ” on page 341.
vxfendisk	Generates the list of paths of disks in the diskgroup. This utility requires that Veritas Volume Manager is installed and configured.
cpsadm	Administer and maintain CP servers in server-based fencing configurations. See “ Administering the coordination point server ” on page 349.

The I/O fencing commands reside in the /opt/VRTS/bin folder. Make sure you added this folder path to the PATH environment variable.

Refer to the corresponding manual page for more information on the commands.

About the vxfcntlsthdw utility

You can use the vxfcntlsthdw utility to verify that shared storage arrays to be used for data support SCSI-3 persistent reservations and I/O fencing. During the I/O fencing configuration, the testing utility is used to test a single disk. The utility has other options that may be more suitable for testing storage devices in other configurations. You also need to test coordinator disk groups.

See *Veritas Cluster Server Installation Guide* to set up I/O fencing.

The utility, which you can run from one system in the cluster, tests the storage used for data by setting and verifying SCSI-3 registrations on the disk or disks you specify, setting and verifying persistent reservations on the disks, writing data to the disks and reading it, and removing the registrations from the disks. Refer also to the vxfcntlsthaw(1M) manual page.

General guidelines for using vxfcntlsthaw

- The utility requires two systems connected to the shared storage.

Caution: The tests overwrite and destroy data on the disks, unless you use the `-r` option.

- The two nodes must have `ssh` (default) or `rsh` communication. If you use `rsh`, launch the vxfcntlsthaw utility with the `-n` option.
After completing the testing process, you can remove permissions for communication and restore public network connections.
- To ensure both systems are connected to the same disk during the testing, you can use the `vxfenadm -i diskpath` command to verify a disk's serial number.
See [“Verifying the nodes see the same disk”](#) on page 339.
- For disk arrays with many disks, use the `-m` option to sample a few disks before creating a disk group and using the `-g` option to test them all.
- The utility indicates a disk can be used for I/O fencing with a message resembling:

```
The disk /dev/rhdisk75 is ready to be configured for I/O Fencing  
on node galaxy
```


If the utility does not show a message stating a disk is ready, verification has failed.
- If the disk you intend to test has existing SCSI-3 registration keys, the test issues a warning before proceeding.

vxfcntlsthdlw options

Table 10-2 describes three methods the utility provides to test storage devices.

Table 10-2 vxfcntlsthdlw options

vxfcntlsthdlw option	Description	When to use
-n	Utility uses rsh for communication.	Use when rsh is used for communication.
-r	Non-destructive testing. Testing of the disks for SCSI-3 persistent reservations occurs in a non-destructive way; that is, there is only testing for reads, not writes. May be used with -m, -f, or -g options.	Use during non-destructive testing. See “Performing non-destructive testing on the disks using the -r option” on page 334.
-t	Testing of the return value of SCSI TEST UNIT (TUR) command under SCSI-3 reservations. A warning is printed on failure of TUR testing.	When you want to perform TUR testing.
-d	Use DMP devices. May be used with -c or -g options.	By default, the script picks up the DMP paths for disks in the diskgroup. If you want the script to use the raw paths for disks in the diskgroup, use the -w option.
-w	Use raw devices. May be used with -c or -g options.	With the -w option, the script picks the operating system paths for disks in the diskgroup. By default, the script uses the -d option to pick up the DMP paths for disks in the disk group.
-c	Utility tests the coordinator disk group prompting for systems and devices, and reporting success or failure.	For testing disks in coordinator disk group. See “Testing the coordinator disk group using vxfcntlsthdlw -c” on page 332.

Table 10-2 vxfststhdw options

vxfststhdw option	Description	When to use
-m	Utility runs manually, in interactive mode, prompting for systems and devices, and reporting success or failure. May be used with -r and -t options. -m is the default option.	For testing a few disks or for sampling disks in larger arrays. See “Testing the shared disks using the -m option” on page 331.
-f <i>filename</i>	Utility tests system/device combinations listed in a text file. May be used with -r and -t options.	For testing several disks. See “Testing the shared disks listed in a file using the -f option” on page 334.
-g <i>disk_group</i>	Utility tests all disk devices in a specified disk group. May be used with -r and -t options.	For testing many disks and arrays of disks. Disk groups may be temporarily created for testing purposes and destroyed (ungrouped) after testing. See “Testing all the disks in a diskgroup using the -g option” on page 334.

Testing the shared disks using the -m option

Review the procedure to test the shared disks. By default, the utility uses the -m option.

This procedure uses the /dev/rhdisk75 disk in the steps.

If the utility does not show a message stating a disk is ready, verification has failed. Failure of verification can be the result of an improperly configured disk array. It can also be caused by a bad disk.

If the failure is due to a bad disk, remove and replace it. The vxfststhdw utility indicates a disk can be used for I/O fencing with a message resembling:

```
The disk /dev/rhdisk75 is ready to be configured for I/O Fencing on
node galaxy
```

Note: For A/P arrays, run the vxfststhdw command only on secondary paths.

To test disks using vxfcntlshdw script

- 1 Make sure system-to-system communication is functioning properly.
- 2 From one node, start the utility.

```
# vxfcntlshdw [-n]
```
- 3 After reviewing the overview and warning that the tests overwrite data on the disks, confirm to continue the process and enter the node names.

```
***** WARNING!!!!!!!!!! *****
THIS UTILITY WILL DESTROY THE DATA ON THE DISK!!
...
Do you still want to continue : [y/n] (default: n) y
Enter the first node of the cluster: galaxy
Enter the second node of the cluster: nebula
```
- 4 Enter the names of the disks you are checking. For each node, the disk may be known by the same name:

```
Enter the disk name to be checked for SCSI-3 PGR on node galaxy
in the format:
    for dmp: /dev/vx/rdmp/DiskXX
    for raw: /dev/rhdiskx
/dev/rhdisk75
Enter the disk name to be checked for SCSI-3 PGR on node nebula
in the format:
    for dmp: /dev/vx/rdmp/DiskXX
    for raw: /dev/rhdiskx
Make sure it's the same disk as seen by nodes galaxy and nebula
/dev/rhdisk75
If the serial numbers of the disks are not identical, then the test terminates.
```
- 5 Review the output as the utility performs the checks and report its activities.
- 6 If a disk is ready for I/O fencing on each node, the utility reports success:

```
ALL tests on the disk /dev/rhdisk75 have PASSED
The disk is now ready to be configured for I/O Fencing on node
galaxy
```
- 7 Run the vxfcntlshdw utility for each disk you intend to verify.

Testing the coordinator disk group using vxfcntlshdw -c

Use the vxfcntlshdw utility to verify disks are configured to support I/O fencing. In this procedure, the vxfcntlshdw utility tests the three disks one disk at a time from each node.

- From the node galaxy, the disks are /dev/rhdisk75, /dev/rhdisk76, and /dev/rhdisk77.
- From the node nebula, the same disks are /dev/rhdisk80, /dev/rhdisk81, and /dev/rhdisk82.

Note: To test the coordinator disk group using the vxfcntlsthdw utility, the utility requires that the coordinator disk group, vxfcntlcoorddg, be accessible from two nodes.

To test the coordinator disk group using vxfcntlsthdw -c

- 1 Use the vxfcntlsthdw command with the -c option. For example:

```
# vxfcntlsthdw -c vxfcntlcoorddg
```
- 2 Enter the nodes you are using to test the coordinator disks:
Enter the first node of the cluster:
galaxy
Enter the second node of the cluster:
nebula
- 3 Review the output of the testing process for both nodes for all disks in the coordinator disk group. Each disk should display output that resembles:

```
ALL tests on the disk /dev/rhdisk75 have PASSED.  
The disk is now ready to be configured for I/O Fencing on node  
galaxy as a COORDINATOR DISK.  
  
ALL tests on the disk /dev/rhdisk80 have PASSED.  
The disk is now ready to be configured for I/O Fencing on node  
nebula as a COORDINATOR DISK.
```
- 4 After you test all disks in the disk group, the vxfcntlcoorddg disk group is ready for use.

Removing and replacing a failed disk

If a disk in the coordinator disk group fails verification, remove the failed disk or LUN from the vxfcntlcoorddg disk group, replace it with another, and retest the disk group.

To remove and replace a failed disk

- 1 Use the vxdiskadm utility to remove the failed disk from the disk group. Refer to the *Veritas Volume Manager Administrator's Guide*.
- 2 Add a new disk to the node, initialize it, and add it to the coordinator disk group.
See *Veritas Cluster Server Installation Guide* for instructions to initialize disks for I/O fencing and to set up coordinator disk groups.
- 3 Retest the disk group.
See [“Testing the coordinator disk group using vxfcntlsthdw -c”](#) on page 332.

Performing non-destructive testing on the disks using the `-r` option

You can perform non-destructive testing on the disk devices when you want to preserve the data.

To perform non-destructive testing on disks

- ◆ To test disk devices containing data you want to preserve, you can use the `-r` option with the `-m`, `-f`, or `-g` options.

For example, to use the `-m` option and the `-r` option, you can run the utility as follows:

```
# vxfsentsthdw -rm
```

When invoked with the `-r` option, the utility does not use tests that write to the disks. Therefore, it does not test the disks for all of the usual conditions of use.

Testing the shared disks listed in a file using the `-f` option

Use the `-f` option to test disks that are listed in a text file. Review the following example procedure.

To test the shared disks listed in a file

- 1 Create a text file `disks_test` to test two disks shared by systems `galaxy` and `nebula` that might resemble:

```
galaxy /dev/rhdisk75 nebula /dev/rhdisk77  
galaxy /dev/rhdisk76 nebula /dev/rhdisk78
```

Where the first disk is listed in the first line and is seen by `galaxy` as `/dev/rhdisk75` and by `nebula` as `/dev/rhdisk77`. The other disk, in the second line, is seen as `/dev/rhdisk76` from `galaxy` and `/dev/rhdisk78` from `nebula`. Typically, the list of disks could be extensive.

- 2 To test the disks, enter the following command:

```
# vxfsentsthdw -f disks_test
```

The utility reports the test results one disk at a time, just as for the `-m` option.

Testing all the disks in a diskgroup using the `-g` option

Use the `-g` option to test all disks within a disk group. For example, you create a temporary disk group consisting of all disks in a disk array and test the group.

Note: Do not import the test disk group as shared; that is, do not use the `-s` option.

After testing, destroy the disk group and put the disks into disk groups as you need.

To test all the disks in a diskgroup

- 1 Create a diskgroup for the disks that you want to test.
- 2 Enter the following command to test the diskgroup test_disks_dg:

```
# vxfststhdw -g test_disks_dg
```

The utility reports the test results one disk at a time.

Testing a disk with existing keys

If the utility detects that a coordinator disk has existing keys, you see a message that resembles:

```
There are Veritas I/O fencing keys on the disk. Please make sure
that I/O fencing is shut down on all nodes of the cluster before
continuing.
```

```
***** WARNING!!!!!!!!!! *****
```

```
THIS SCRIPT CAN ONLY BE USED IF THERE ARE NO OTHER ACTIVE NODES
IN THE CLUSTER!  VERIFY ALL OTHER NODES ARE POWERED OFF OR
INCAPABLE OF ACCESSING SHARED STORAGE.
```

If this is not the case, data corruption will result.

```
Do you still want to continue : [y/n] (default: n) y
```

The utility prompts you with a warning before proceeding. You may continue as long as I/O fencing is not yet configured.

About the vxfenadm utility

Administrators can use the `vxfenadm` command to troubleshoot and test fencing configurations.

The command’s options for use by administrators are as follows:

- s

read the keys on a disk and display the keys in numeric, character, and node format
- Note:** The -g and -G options are deprecated. Use the -s option.
- i

read SCSI inquiry information from device
- m

register with disks
- n

make a reservation with disks
- p

remove registrations made by other systems
- r

read reservations
- x

remove registrations

Refer to the `vxfenadm(1m)` manual page for a complete list of the command options.

About the I/O fencing registration key format

The keys that the `vxfen` driver registers on the data disks and the coordinator disks consists of eight bytes. The key format is different for the coordinator disks and data disks.

The key format of the coordinator disks is as follows:

Byte	0	1	2	3	4	5	6	7
Value	V	F	cID 0x	cID 0x	cID 0x	cID 0x	nID 0x	nID 0x

where

- VF is the unique identifier that carves out a namespace for the keys (consumes two bytes)
- cID 0x is the LLT cluster ID in hexadecimal (consumes four bytes)
- nID 0x is the LLT node ID in hexadecimal (consumes two bytes)

The `vxfen` driver uses this key format in both `scsi3` mode and customized mode of I/O fencing.

The key format of the data disks that are configured as failover disk groups under VCS is as follows:

Byte	0	1	2	3	4	5	6	7
Value	A+nID	V	C	S				

where nID is the LLT node ID

For example:

If the node ID is 1, then the first byte has the value as B ('A' + 1 = B).

The key format of the data disks configured as parallel disk groups under CVM is as follows:

Byte	0	1	2	3	4	5	6	7
Value	A+nID	P	G	R	DGcount	DGcount	DGcount	DGcount

where DGcount is the count of disk group in the configuration

Displaying the I/O fencing keys on the disks

You can display the keys that are currently assigned to the disks using the vxfenadm command.

To display the I/O fencing keys on the disks

- 1 To display the key for the disks, run the following command:

```
# vxfenadm -s disk_name
```

For example:

- To display the key for the coordinator disk /dev/rhdisk74 from the system with node ID 1, enter the following command:

```
# vxfenadm -s /dev/rhdisk74
```

```
key[1]:
```

```
[Numeric Format]: 86,70,68,69,69,68,48,48
```

```
[Character Format]: VFDEED00
```

```
* [Node Format]: Cluster ID: 57069 Node ID: 0 Node Name: galaxy
```

The -s option of vxfenadm displays all eight bytes of a key value in three formats. In the numeric format,

- The first two bytes, represent the identifier VF, contains the ASCII value 86, 70.
- The next four bytes contain the ASCII value of the cluster ID 57069 encoded in hex (0xDEED) which are 68, 69, 69, 68.

- The remaining bytes contain the ASCII value of the node ID 0 (0x00) which are 48, 48. Node ID 1 would be 01 and node ID 10 would be 0A.

An asterisk before the Node Format indicates that the vxfenadm command is run from the node of a cluster where LLT is configured and running.

- To display the keys on a CVM parallel disk group:

```
# vxfenadm -s /dev/vx/rdmp/disk_7
```

```
Reading SCSI Registration Keys...
```

```
Device Name: /dev/vx/rdmp/disk_7
```

```
Total Number Of Keys: 1
```

```
key[0]:
```

```
[Numeric Format]: 66,80,71,82,48,48,48,48
```

```
[Character Format]: BPGR0001
```

```
[Node Format]: Cluster ID: unknown Node ID: 1 Node Name: nebula
```

- To display the keys on a VCS failover disk group:

```
# vxfenadm -s /dev/vx/rdmp/disk_8
```

```
Reading SCSI Registration Keys...
```

```
Device Name: /dev/vx/rdmp/disk_8
```

```
Total Number Of Keys: 1
```

```
key[0]:
```

```
[Numeric Format]: 65,86,67,83,0,0,0,0
```

```
[Character Format]: AVCS
```

```
[Node Format]: Cluster ID: unknown Node ID: 0 Node Name: galaxy
```

- 2 To display the keys that are registered in all the disks specified in a disk file:

```
# vxfenadm -s all -f disk_filename
```

For example:

To display all the keys on coordinator disks:

```
# vxfenadm -s all -f /etc/vxfentab
```

```
Device Name: /dev/vx/rdmp/disk_9
```

```
Total Number Of Keys: 2
```

```
key[0]:
```

```
[Numeric Format]: 86,70,66,69,65,68,48,50
```

```
[Character Format]: VFBEAD02
```

```
[Node Format]: Cluster ID: 48813 Node ID: 2 Node Name: unknown
```

```
key[1]:
```

```
[Numeric Format]: 86,70,68,69,69,68,48,48
```

```
[Character Format]: VFDEED00
```

```
* [Node Format]: Cluster ID: 57069 Node ID: 0 Node Name: galaxy
```

You can verify the cluster ID using the lltstat -C command and the node ID using the lltstat -N command. For example:

```
# lltstat -C
```

57069

- If the disk has keys which do not belong to a specific cluster, then the vxfenadm command cannot lookup the node name for the node ID and hence prints the node name as unknown. For example:

```
Device Name: /dev/vx/rdmp/disk_7
Total Number Of Keys: 1
key[0]:
[Numeric Format]: 86,70,45,45,45,45,48,49
[Character Format]: VF---01
[Node Format]: Cluster ID: unknown Node ID: 1 Node Name: nebula
```

- For disks with arbitrary format of keys, the vxfenadm command prints all the fields as unknown. For example:

```
[Numeric Format]: 65,66,67,68,49,50,51,45
[Character Format]: ABCD123-
[Node Format]: Cluster ID: unknown Node ID: unknown Node Name:
unknown
```

Verifying the nodes see the same disk

To confirm whether a disk (or LUN) supports SCSI-3 persistent reservations, two nodes must simultaneously have access to the same disks. Because a shared disk is likely to have a different name on each node, check the serial number to verify the identity of the disk. Use the vxfenadm command with the -i option to verify that the same serial number for the LUN is returned on all paths to the LUN.

For example, an EMC disk is accessible by the /dev/rhdisk75 path on node A and the /dev/rhdisk76 path on node B.

To verify that the nodes see the same disks

- 1 Verify the connection of the shared storage for data to two of the nodes on which you installed Veritas Cluster Server.
- 2 From node A, enter the following command:

```
# vxfenadm -i /dev/rhdisk75
Vendor id      : EMC
Product id     : SYMMETRIX
Revision      : 5567
Serial Number  : 42031000a
```

The same serial number information should appear when you enter the equivalent command on node B using the /dev/rhdisk76 path.

On a disk from another manufacturer, Hitachi Data Systems, the output is different and may resemble:

```
# vxfenadm -i /dev/rhdisk77
Vendor id      : HITACHI
Product id     : OPEN-3
Revision      : 0117
Serial Number  : 0401EB6F0002
```

Refer to the `vxfenadm(1M)` manual page.

About the `vxfenclearpre` utility

You can use the `vxfenclearpre` utility to remove SCSI-3 registrations and reservations on the disks.

This utility currently does not support server-based fencing. You must manually resolve any preexisting split-brain with server-based fencing configuration.

See [“Issues during fencing start up on VCS cluster node”](#) on page 607.

Removing preexisting keys

If you encountered a split brain condition, use the `vxfenclearpre` utility to remove SCSI-3 registrations and reservations on the coordinator disks as well as on the data disks in all shared disk groups.

You can also use this procedure to remove the registration and reservation keys created by another node from a disk.

To clear keys after split brain

- 1 Stop VCS and I/O fencing on all nodes.

```
# hastop -all
```

Enter the following command on each node:

```
# /etc/init.d/vxfen.rc stop
```

If you have any applications that run outside of VCS control that have access to the shared storage, then shut down all other nodes in the cluster that have access to the shared storage. This prevents data corruption.

- 2 Start the script:

```
# vxfenclearpre [ -t all | -t vxfen | -t cluster -c clusterid ]
```

The `-t` option applies only to the coordinator disks. If you do not use the `-t` option, the script clears all the keys from coordinator disks and data disks.

If you use the `-t` option, the script behavior is as follows:

<code>-t all</code>	Clears all keys from the coordinator disks
<code>-t vxfen</code>	Clears all vxfen keys (with the VF prefix) from the coordinator disks
<code>-t cluster -c clusterid</code>	Clears only vxfen keys from the coordinator disks that matches the <i>clusterid</i> you have specified.

The *clusterid* value is the cluster id that you provided during VCS configuration and ranges between 0 and 65535.

- 3 Read the script's introduction and warning. Review the list of keys for the disks that the script will clear before you enter **y** at the prompt to proceed.

```
Do you still want to continue: [y/n] (default : n) y  
Cleaning up the coordinator disks...
```

```
Cleaning up the data disks for all shared disk groups...
```

```
Successfully removed SCSI-3 persistent registration and  
reservations from the coordinator disks as well as the shared  
data disks.
```

```
You can retry starting fencing module. In order to restart the  
whole  
product, you might want to restart the system.
```

```
#
```

- 4 Restart all nodes in the cluster.

About the vxfsnwap utility

The vxfsnwap utility allows you to replace coordinator disks in a cluster that is online. The utility verifies that the serial number of the new disks are identical on all the nodes and the new disks can support I/O fencing.

This utility also supports server-based fencing.

Refer to the vxfsnwap(1M) manual page.

See *Veritas Cluster Server Installation Guide* for details on the coordinator disk requirements.

You can replace the coordinator disks without stopping I/O fencing in the following cases:

- The disk becomes defective or inoperable and you want to switch to a new diskgroup.
See [“Replacing I/O fencing coordinator disks when the cluster is online”](#) on page 342.
See [“Replacing the coordinator diskgroup in a cluster that is online”](#) on page 344.
If you want to replace the coordinator disks when the cluster is offline, you cannot use the vxfsnwap utility. You must manually perform the steps that the utility does to replace the coordinator disks.
See [“Replacing defective disks when the cluster is offline”](#) on page 604.
- You want to switch the disk interface between raw devices and DMP devices.
See [“Changing the disk interaction policy in a cluster that is online”](#) on page 346.

- New disks are available to act as coordinator disks.
See [“Adding disks from a recovered site to the coordinator diskgroup”](#) on page 347.
- The keys that are registered on the coordinator disks are lost.
In such case, the cluster might panic when a split-brain occurs. You can replace the coordinator disks with the same disks using the vxfenswap command. During the disk replacement, the missing keys register again without any risk of data corruption.
See [“Refreshing lost keys on coordinator disks”](#) on page 348.

In server-based fencing configuration, you can use the vxfenswap utility to perform the following tasks:

- Perform a planned replacement of customized coordination points (CP servers or SCSI-3 disks).
See [“Replacing coordination points for server-based fencing in an online cluster”](#) on page 355.
- Refresh the I/O fencing keys that are registered on the coordination points.
See [“Refreshing registration keys on the coordination points for server-based fencing”](#) on page 357.

If the vxfenswap operation is unsuccessful, then you can use the `-a cancel` option of the vxfenswap command to manually roll back the changes that the vxfenswap utility does.

- For disk-based fencing, use the `vxfenswap -g diskgroup -a cancel` command to cancel the vxfenswap operation.
You must run this command if a node fails during the process of disk replacement, or if you aborted the disk replacement.
- For server-based fencing, use the `vxfenswap -a cancel` command to cancel the vxfenswap operation.

Replacing I/O fencing coordinator disks when the cluster is online

Review the procedures to add, remove, or replace one or more coordinator disks in a cluster that is operational.

Warning: The cluster might panic if any node leaves the cluster membership before the vxfenswap script replaces the set of coordinator disks.

To replace a disk in an active coordinator diskgroup

- 1 Make sure system-to-system communication is functioning properly.

2 Make sure that the cluster is online.

```
# vxfenadm -d
I/O Fencing Cluster Information:
=====
Fencing Protocol Version: 201
Fencing Mode: SCSI3
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
    * 0 (galaxy)
    1 (nebula)
RFSM State Information:
    node 0 in state 8 (running)
    node 1 in state 8 (running)
```

3 Import the coordinator disk group.

The file `/etc/vxfendg` includes the name of the disk group (typically, `vxfencoordg`) that contains the coordinator disks, so use the command:

```
# vxdg -tfc import `cat /etc/vxfendg`
```

where:

- t specifies that the disk group is imported only until the node restarts.
- f specifies that the import is to be done forcibly, which is necessary if one or more disks is not accessible.
- C specifies that any import locks are removed.

4 Turn off the coordinator attribute value for the coordinator disk group.

```
# vxdg -g vxfencoordg set coordinator=off
```

5 To remove disks from the disk group, use the VxVM disk administrator utility `vxdiskadm`.**6 Perform the following steps to add new disks to the coordinator disk group:**

- Add new disks to the node.
- Initialize the new disks as VxVM disks.
- Check the disks for I/O fencing compliance.
- Add new disks to the `vxfencoordg` disk group and set the coordinator attribute value as “on” for the coordinator disk group.

See the *Veritas Cluster Server Installation Guide* for detailed instructions.

Note that though the diskgroup content changes, the I/O fencing remains in the same state.

7 Make sure that the `/etc/vxfenmode` file is updated to specify the correct disk policy.

See the *Veritas Cluster Server Installation Guide* for detailed instructions.

8 From one node, start the `vxfenswap` utility. You must specify the diskgroup to the utility:

Do one of the following:

- If you use `ssh` for communication:

```
# vxfenswap -g diskgroup
```

- If you use `rsh` for communication:

```
# vxfenswap -g diskgroup -n
```

The utility performs the following tasks:

- Backs up the existing `/etc/vxfentab` file.
 - Creates a test file `/etc/vxfentab.test` for the diskgroup that is modified on each node of the cluster.
 - Reads the diskgroup you specified in the `vxfenwap` command and adds the diskgroup to the `/etc/vxfentab.test` file on each node.
 - Verifies that the serial number of the new disks are identical on all the nodes. The script terminates if the check fails.
 - Verifies that the new disks can support I/O fencing on each node.
- 9 If the disk verification passes, the script reports success and asks if you want to commit the new set of coordinator disks.
 - 10 Review the message that the utility displays and confirm that you want to commit the new set of coordinator disks. Else skip to step 13.
Do you wish to commit this change? [y/n] (default: n) **y**
If the utility successfully commits, the script moves the `/etc/vxfentab.test` file to the `/etc/vxfentab` file.
 - 11 If you do not want to commit the new set of coordinator disks, answer `n`.
The `vxfenswap` utility rolls back the disk replacement operation.

Replacing the coordinator diskgroup in a cluster that is online

You can also replace the coordinator diskgroup using the `vxfenswap` utility. The following example replaces the coordinator disk group `vxfencoorddg` with a new disk group `vx fendg`.

To replace the coordinator diskgroup

- 1 Make sure system-to-system communication is functioning properly.
- 2 Make sure that the cluster is online.

```
# vxfenadm -d
I/O Fencing Cluster Information:
=====
Fencing Protocol Version: 201
Fencing Mode: SCSI3
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
    * 0 (galaxy)
```



```

1 (nebula)
RFSM State Information:
  node 0 in state 8 (running)
  node 1 in state 8 (running)

```

- 3 Find the name of the current coordinator diskgroup (typically vxfencoorddg) that is in the /etc/vxfendg file.

```

# cat /etc/vxfendg
vxfencoorddg

```

- 4 Find the alternative disk groups available to replace the current coordinator diskgroup.

```

# vxdisk -o alldgs list

```

DEVICE	TYPE	DISK	GROUP	STATUS
rhdisk64	auto:cdsdisk	-	(vxfendg)	online
rhdisk65	auto:cdsdisk	-	(vxfendg)	online
rhdisk66	auto:cdsdisk	-	(vxfendg)	online
rhdisk75	auto:cdsdisk	-	(vxfencoorddg)	online
rhdisk76	auto:cdsdisk	-	(vxfencoorddg)	online
rhdisk77	auto:cdsdisk	-	(vxfencoorddg)	online

- 5 Validate the new disk group for I/O fencing compliance. Run the following command:

```

# vxfentsthdw -c vxfendg
vxfentsthdw -c vxfendg

```

See “[Testing the coordinator disk group using vxfentsthdw -c](#)” on page 332.

- 6 If the new disk group is not already deported, run the following command to deport the disk group:

```

# vxdg deport vxfendg

```

- 7 Make sure that the /etc/vxfenmode file is updated to specify the correct disk policy.

See the *Veritas Cluster Server Installation Guide* for detailed instructions.

- 8 From any node, start the vxfenswap utility. For example, if vxfendg is the new diskgroup that you want to use as the coordinator diskgroup:

```

# vxfenswap -g vxfendg [-n]

```

The utility performs the following tasks:

- Backs up the existing /etc/vxfentab file.
- Creates a test file /etc/vxfentab.test for the diskgroup that is modified on each node of the cluster.
- Reads the diskgroup you specified in the vxfenwap command and adds the diskgroup to the /etc/vxfentab.test file on each node.
- Verifies that the serial number of the new disks are identical on all the nodes. The script terminates if the check fails.
- Verifies that the new disk group can support I/O fencing on each node.

- 9 If the disk verification passes, the script reports success and asks if you want to replace the coordinator disk group.
- 10 Review the message that the utility displays and confirm that you want to replace the diskgroup. Else skip to step 13.
Do you wish to commit this change? [y/n] (default: n) **y**
If the utility successfully commits, the script moves the /etc/vxfentab.test file to the /etc/vxfentab file.
The utility also updates the /etc/vxfendg file with this new diskgroup.
- 11 Set the coordinator attribute value as “on” for the new coordinator disk group.

```
# vxldg -g vxfendg set coordinator=on
```

Set the coordinator attribute value as “off” for the old disk group.

```
# vxldg -g vxfencoordg set coordinator=off
```
- 12 Verify that the coordinator disk group has changed.

```
# cat /etc/vxfendg
```

vxfendg
- 13 If you do not want to replace the diskgroup, answer n.
The vxfenswap utility rolls back the disk replacement operation.

Changing the disk interaction policy in a cluster that is online

In a cluster that is online, you can change the disk interaction policy from raw to dmp using the vxfenswap utility.

To change the disk interaction policy

- 1 Make sure system-to-system communication is functioning properly.
- 2 Make sure that the cluster is online, and check the disk policy that is used.

```
# vxfenadm -d
```

I/O Fencing Cluster Information:
=====
Fencing Protocol Version: 201
Fencing Mode: SCSI3
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
* 0 (galaxy)
1 (nebula)
RFSM State Information:
node 0 in state 8 (running)
node 1 in state 8 (running)
- 3 On each node in the cluster, edit the /etc/vxfenmode file to change the disk policy.

```
cat /etc/vxfenmode
```

vxfen_mode=scsi3

- ```
scsi3_disk_policy=raw
```
- 4 From any node, start the vxfenswap utility:  

```
vxfenswap -g vxfencoorddg [-n]
```
  - 5 Verify the change in the disk policy.  

```
vxfenadm -d
```

I/O Fencing Cluster Information:  
=====
Fencing Protocol Version: 201  
Fencing Mode: SCSI3  
Fencing SCSI3 Disk Policy: raw

## Adding disks from a recovered site to the coordinator diskgroup

In a campus cluster environment, consider a case where the primary site goes down and the secondary site comes online with a limited set of disks. When the primary site restores, the primary site's disks are also available to act as coordinator disks. You can use the vxfenswap utility to add these disks to the coordinator diskgroup.

See [“About I/O fencing in campus clusters”](#) on page 554.

### To add new disks from a recovered site to the coordinator diskgroup

- 1 Make sure system-to-system communication is functioning properly.
- 2 Make sure that the cluster is online.

```
vxfenadm -d
```

I/O Fencing Cluster Information:  
=====
Fencing Protocol Version: 201  
Fencing Mode: SCSI3  
Fencing SCSI3 Disk Policy: dmp  
Cluster Members:  
\* 0 (galaxy)  
1 (nebula)  
RFSM State Information:  
node 0 in state 8 (running)  
node 1 in state 8 (running)

- 3 Verify the name of the coordinator diskgroup.

```
cat /etc/vxfendg
vxfencoorddg
```

- 4 Run the following command:

```
vxdisk -o alldgs list
```

| DEVICE   | TYPE         | DISK | GROUP          | STATUS  |
|----------|--------------|------|----------------|---------|
| rhdisk75 | auto:cdsdisk | -    | (vxfencoorddg) | online  |
| rhdisk75 | auto         | -    | -              | offline |
| rhdisk75 | auto         | -    | -              | offline |

- 5 Verify the number of disks used in the coordinator diskgroup.

```
vxfenconfig -l
I/O Fencing Configuration Information:
=====
Count : 1
Disk List
Disk Name Major Minor Serial Number Policy
/dev/vx/rdmp/rhdisk75 32 48 R450 00013154 0312 dmp
```

- 6 When the primary site comes online, start the vxfenswap utility on any node in the cluster:

```
vxfenswap -g vxfencoordg [-n]
```

- 7 Verify the count of the coordinator disks.

```
vxfenconfig -l
I/O Fencing Configuration Information:
=====
Single Disk Flag : 0
Count : 3
Disk List
Disk Name Major Minor Serial Number Policy
/dev/vx/rdmp/rhdisk75 32 48 R450 00013154 0312 dmp
/dev/vx/rdmp/rhdisk76 32 32 R450 00013154 0313 dmp
/dev/vx/rdmp/rhdisk77 32 16 R450 00013154 0314 dmp
```

## Refreshing lost keys on coordinator disks

If the coordinator disks lose the keys that are registered, the cluster might panic when a split-brain occurs.

You can use the vxfenswap utility to replace the coordinator disks with the same disks. The vxfenswap utility registers the missing keys during the disk replacement.

### To refresh lost keys on coordinator disks

- 1 Make sure system-to-system communication is functioning properly.
- 2 Make sure that the cluster is online.

```
vxfenadm -d
I/O Fencing Cluster Information:
=====
Fencing Protocol Version: 201
Fencing Mode: SCSI3
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
 * 0 (galaxy)
 1 (nebula)
RFSM State Information:
 node 0 in state 8 (running)
 node 1 in state 8 (running)
```

- 3 Run the following command to view the coordinator disks that do not have keys:  

```
vxfenadm -s all -f /etc/vxfentab
Device Name: /dev/vx/rdmp/rhdisk75
Total Number of Keys: 0
No keys...
...
```
- 4 On any node, run the following command to start the vxfereswap utility:  

```
vxfereswap -g vxferescoorddg [-n]
```
- 5 Verify that the keys are atomically placed on the coordinator disks.  

```
vxfenadm -s all -f /etc/vxfentab
Device Name: /dev/vx/rdmp/rhdisk75
Total Number of Keys: 4
...
```

# Administering the coordination point server

This section describes how to perform administrative and maintenance tasks on the coordination point server.

For more information about the `cpsadm` command and the associated command options, see the `cpsadm(1M)` manual page.

## Environment variables associated with the coordination point server

This section describes the environment variables that are required for the `cpsadm` command. The `cpsadm` command detects these environment variables and uses their value when communicating with the CP server.

These environment variables are not required when the `cpsadm` command is run on the CP server, they are required when the `cpsadm` command is run on the client cluster nodes.

| Environment variable | Description                                                                                                                                              |
|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| CPS_USERNAME         | Fully qualified username as configured in VxSS ( <code>vssat showcred</code> ).                                                                          |
| CPS_DOMAINTYPE       | One of the following values: <ul style="list-style-type: none"> <li>■ vx</li> <li>■ unixpwd</li> <li>■ nis</li> <li>■ nisplus</li> <li>■ ldap</li> </ul> |

You must export these variables directly on the shell before running the `cpsadm` command.

For example,

```
export CPS_USERNAME=
export CPS_DOMAINTYPE=
```

The username and domain type values are the same as those added onto the CP server. To view these values run the following command:

```
cpsadm -s cp_server -a list_users
```

## Adding and removing a VCS cluster from the CP server database

To add a VCS cluster to a CP server

- ◆ Type the following command:

```
cpsadm -s cp_server -a add_clus -c cluster_name -u uuid
```

To remove a VCS cluster from a CP server

- ◆ Type the following command:

```
cpsadm -s cp_server -a rm_clus -u uuid
```

|                           |                                                          |
|---------------------------|----------------------------------------------------------|
| <code>cp_server</code>    | The CP server's virtual host name or virtual IP address. |
| <code>cluster_name</code> | The name of the VCS cluster.                             |
| <code>uuid</code>         | The UUID (Universally Unique ID) of the VCS cluster.     |

## Adding and removing a VCS cluster node from the CP server database

To add a node

- ◆ Type the following command:

```
cpsadm -s cp_server -a add_node -u uuid -n nodeid -h host
```

To remove a node

- ◆ Type the following command:

```
cpsadm -s cp_server -a rm_node -u uuid -n nodeid
```

|                        |                                                          |
|------------------------|----------------------------------------------------------|
| <code>cp_server</code> | The CP server's virtual host name or virtual IP address. |
| <code>node_id</code>   | The node id of the VCS cluster node.                     |
| <code>uuid</code>      | The UUID (Universally Unique ID) of the VCS cluster.     |

## Adding and removing a CP server user

### To add a user

- ◆ Type the following command:

```
cpsadm -s cp_server -a add_user -e user_name -f user_role\
-g domain_type -u uuid
```

### To remove a user

- ◆ Type the following command:

```
cpsadm -s cp_server -a rm_user -e user_name -g domain_type
```

|             |                                                          |
|-------------|----------------------------------------------------------|
| cp_server   | The CP server's virtual host name or virtual IP address. |
| user_name   | The user to be added to the CP server configuration.     |
| uuid        | The UUID (Universally Unique ID) of the VCS cluster.     |
| user_role   | The user role, either cps_admin or cps_operator.         |
| domain_type | The domain type, for example vx, unixpwd, nis, etc.      |

## Listing the CP server users

### To list the CP server users

- ◆ Type the following command:

```
cpsadm -s cp_server -a list_users
```

## Listing the nodes in all the client clusters

### To list the nodes in all the client cluster

- ◆ Type the following command:

```
cpsadm -s cp_server -a list_nodes
```

## Listing the membership of nodes in the VCS cluster

### To list the membership of nodes in client cluster

- ◆ Type the following command:

```
cpsadm -s cp_server -a list_membership -c cluster_name
```

## Preempting a node

### To preempt a node

- ◆ Type the following command:  

```
cpsadm -s cp_server -a preempt_node -u uuid -n nodeid \
-v victim_node_id
```

*victim\_node\_id*—The victim node's host name.

## Registering and unregistering a node

### To register a node

- ◆ Type the following command:  

```
cpsadm -s cp_server -a reg_node -u uuid -n nodeid
```

### To unregister a node

- ◆ Type the following command:  

```
cpsadm -s cp_server -a unreg_node -u uuid -n nodeid
```

## Enabling and disabling access for a user to a client cluster

### To enable access for a user to a client cluster

- ◆ Type the following command:  

```
cpsadm -s cp_server -a add_clus_to_user -e user_name \
-f user_role -g domain_type -u uuid
```

### To disable access for a user to a client cluster

- ◆ Type the following command:  

```
cpsadm -s cp_server -a rm_clus_from_user -e user_name \
-f user_role -g domain_type -u uuid
```

|                    |                                                                             |
|--------------------|-----------------------------------------------------------------------------|
| <i>cp_server</i>   | The CP server's FQHN or virtual IP address.                                 |
| <i>user_name</i>   | The user to be added to the CP server configuration.                        |
| <i>user_role</i>   | The user role, either <i>cps_admin</i> or <i>cps_operator</i> .             |
| <i>domain_type</i> | The domain type, for example <i>vx</i> , <i>unixpwd</i> , <i>nis</i> , etc. |
| <i>uuid</i>        | The UUID (Universally Unique ID) of the VCS cluster.                        |



## Stopping the CP server

### To stop the CP server

- ◆ Type the following command:  
# **cpsadm -s cp\_server -a halt\_cps**

## Checking the connectivity of CP server

### To check the connectivity of CP server

- ◆ Type the following command:  
# **cpsadm -s cp\_server -a ping\_cps**

## Configuring server-based fencing on a new node

This section describes the procedures to configure server-based fencing on a new node. Perform these procedures if your existing cluster uses server-based I/O fencing.

See the *Veritas Cluster Server Installation Guide* for a detailed procedure to add a new node to the VCS cluster and to set up server-based fencing on the new node.

### To configure server-based fencing on the new node

- 1 Log in to each CP server as the root user.
- 2 Update each CP server configuration with the new node information:  
# **cpsadm -s cp\_server -a add\_node -u uuid \**  
# **-h host -n nodeid**  
For example:  
# **cpsadm -s mycps1.symantecexample.com \**  
# **-a add\_node -u {f0735332-1dd1-11b2} -h nodeC -n 2**
- 3 Verify that the new node is added to the CP server configuration:  
# **cpsadm -s mycps1.symantecexample.com -a list\_nodes**  
The new node must be listed in the command output.
- 4 Depending on the security setting, add the corresponding user on each CP server.  
For example:  
# **cpsadm -s mycps1.symantecexample.com \**  
# **-a add\_user -e cpsclient@nodeC \**  
# **-f cps\_operator -g vx**

User cpsclient@nodeC successfully added

## Taking a snapshot of the CP server database

Use the `cpsadm` command to take a snapshot of the shared database.

```
cpsadm -s cp_server -a db_snapshot
```

For example:

```
cpsadm -s mycps@symantecexample.com -a db_snapshot
```

The resulting CP server database snapshot is stored at the location that is pointed out by `db` parameter in the file `/etc/vxcps.conf`. By default this location is `/etc/VRTScps/db`.

## Migrating from a non-secure to secure setup for CP server and VCS cluster communication

The following procedure describes how to migrate from a non-secure to secure set up for the CP server and VCS cluster.

### To migrate from non-secure to secure setup for CP server and VCS cluster

- 1 Stop fencing on all the VCS cluster nodes of all the clusters (which are using the CP servers).

```
/etc/init.d/vxfen.rc stop
```

- 2 Stop all the CP servers using the following command on each CP server:

```
hagrps -offline CPSSG -any
```

- 3 Ensure that security is configured for communication between CP servers and VCS cluster nodes.

See the *Veritas Cluster Server Installation Guide* for more information.

- 4 Modify `/etc/vxcps.conf` on each CP server to set `security=1`.

- 5 Start CP servers using the following command:

```
hagrps -online CPSSG -any
```

- 6 Add the following user for each client node on each CP server:

```
_HA_VCS_hostname@HA_SERVICES@FQHN
```

where, *hostname* is the client node name without qualification, and *FQHN* is Fully Qualified Host Name of the VCS node.

- 7 Add the users to the CP server database.

For example, issue the following commands on the CP server (mycps@symantecexample.com):

```
cpsadm -s mycps@symantecexample.com -a add_user \
-e _HA_VCS_galaxy@HA_SERVICES@galaxy.symantec.com \
-f cps_operator -g vx
```

```
User _HA_VCS_galaxy@HA_SERVICES@galaxy.symantec.com successfully
added
```

```
cpsadm -s mycps@symantecexample.com -a add_user \
```

```
-e _HA_VCS_nebula6@HA_SERVICES@nebula.symantec.com \
-f cps_operator -g vx
```

```
User _HA_VCS_nebula6@HA_SERVICES@nebula.symantec.com
successfully added
```

**8** Authorize the user to administer the cluster.

For example, issue the following command on the CP server (mycps@symantecexample.com):

```
cpsadm -s mycps@symantecexample.com -a add_clus_to_user \
-c cpcluster -u {f0735332-1dd1-11b2-a3cb-e3709c1c73b9} \
-e _HA_VCS_galaxy@HA_SERVICES@galaxy.symantec.com \
-f cps_operator -g vx
```

```
Cluster successfully added to user
```

```
_HA_VCS_galaxy@HA_SERVICES@galaxy.symantec.com privileges.
```

**9** Modify /etc/vxfenmode file on each VCS cluster node to set security=1.

**10** Start fencing on VCS cluster node using the following command:

```
/etc/init.d/vxfen.rc start
```

## Replacing coordination points for server-based fencing in an online cluster

Use the following procedure to perform a planned replacement of customized coordination points (CP servers or SCSI-3 disks) without incurring application downtime on an online client cluster.

---

**Note:** If multiple clusters share the same CP server, you must perform this replacement procedure in each cluster.

---

You can use the vxfenswap utility to replace coordination points when fencing is running in customized mode in an online cluster, with vxfen\_mechanism=cps. The utility does not support migration from server-based fencing (vxfen\_mode=customized) to disk-based fencing (vxfen\_mode=scsi3) and vice-versa in an online cluster.

However, in a cluster that is offline you can migrate from disk-based fencing to server-based fencing and vice-versa:

- **Disk-based to server-based fencing:**  
Perform the tasks on the CP server and on the VCS cluster nodes as described in the “Enable fencing in a client cluster with a new CP server” scenario.
- **Server-based to disk-based fencing:**

Perform the tasks on the VCS cluster nodes as described in the “Enable fencing in a client cluster with a new CP server” scenario.

See [“Deployment and migration scenarios for CP server”](#) on page 358.

You can cancel the coordination point replacement operation at any time using the `vxferaswap -a cancel` command.

See [“About the vxferaswap utility”](#) on page 341.

### To replace coordination points for an online cluster

- 1 Ensure that the client cluster nodes and users have been added to the new CP server(s).

This can be checked by running the following commands:

```
cpsadm -s cpserver -a list_nodes
cpsadm -s cpserver -a list_users
```

If the client cluster nodes are not present here, prepare the new CP server(s) for use by the client cluster.

- 2 Ensure that fencing is running on the cluster using the old set of coordination points and in customized mode.

For example, enter the following command:

```
vxferasadm -d
```

The command returns:

```
I/O Fencing Cluster Information:
=====
Fencing Protocol Version: <version>
Fencing Mode: CUSTOMIZED
Cluster Members:
* 0 (galaxy)
1 (nebula)
RFSM State Information:
node 0 in state 8 (running)
node 1 in state 8 (running)
```

- 3 Back up the `/etc/vxferasmode` file on each of the client cluster nodes.
- 4 Use a text editor to access `/etc/vxferasmode` and update the values to the new CP server (coordination points).

The values of the `/etc/vxferasmode` file have to be updated on all the nodes in the VCS cluster.

Review and if necessary, update the `vxferasmode` parameters for security, the coordination points, and if applicable to your configuration, `vxferasdg`.

Refer to the text information within the `vxferasmode` file for additional information about these parameters and their new possible values.

- 5 Run the `vxferaswap` utility from one of the nodes of the cluster.

The `vxferswap` utility requires secure ssh connection to all the cluster nodes.

Use `-n` to use rsh instead of default ssh.

```
vxferswap -n
```

- 6 If validation of coordination points from all the nodes fails, the `vxferswap` utility rollbacks the coordination point replacement operation. Proceed to restore `/etc/vxfermode` with the backed up file on all the VCS cluster nodes.
- 7 You are then prompted to commit the change. Enter `y` for yes.
- 8 Confirm the successful execution of the `vxferswap` utility by checking the coordination points currently used by the `vxfer` driver.  
For example, run the following command:

```
vxferconfig -l
```

## Refreshing registration keys on the coordination points for server-based fencing

You can perform a planned refreshing of registrations on a CP server without incurring application downtime on the VCS cluster. Registration refreshing on a CP server should be performed if the CP server agent issues an alert on the loss of such registrations on the CP server database.

The following procedure describes how to refresh the coordination point registrations.

### To refresh the registration keys on the coordination points for server-based fencing

- 1 Ensure that the VCS cluster nodes and users have been added to the new CP server.

This can be checked by running the following commands:

```
cpsadm -s cp_server -a list_nodes
cpsadm -s cp_server -a list_users
```

- 2 Ensure that fencing is running on the cluster in customized mode using the coordination points mentioned in `/etc/vxfermode` file.

For example, enter the following command:

```
vxferadm -d
```

- 3 List the coordination points currently used by I/O fencing.

```
vxferconfig -l
```

- 4 Run the `vxferswap` utility from one of the nodes of the cluster.

The `vxferswap` utility requires secure ssh connection to all the cluster nodes.

Use `-n` to use rsh instead of default ssh.

For example:

```
vxfenswap -n
```

The command returns:

```
VERITAS vxfenswap version <version> <platform>
The logfile generated for vxfenswap is
/var/VRTSvcs/log/vxfen/vxfenswap.log.
Please Wait...
VXFEN vxfenconfig NOTICE Driver will use customized fencing
- mechanism cps
Validation of coordination points change has succeeded on
all nodes.
You may commit the changes now.
WARNING: This may cause the whole cluster to panic
if a node leaves membership before the change is complete.
```

- 5
- You are then prompted to commit the change. Enter y for yes.  
The command returns a confirmation of successful coordination point replacement.
- 6
- Confirm the successful execution of the vxfenswap utility. If CP agent is configured, it should report ONLINE as it succeeds to find the registrations on coordination points. The registrations on the CP server and coordinator disks can be viewed using the cpsadm and vxfenadm utilities respectively. Note that a running online coordination point refreshment operation can be canceled at any time using the command:  

```
vxfenswap -a cancel
```

## Deployment and migration scenarios for CP server

Table 10-3 describes the supported deployment and migration scenarios, as well as the required procedures to be performed on the VCS cluster and CP server node or cluster.

Table 10-3 CP server deployment and migration scenarios

| Scenario           | CP server     | VCS cluster                                           | Action required                                                                                                                                                                                                                                                                           |
|--------------------|---------------|-------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Setup of CP server | New CP server | New VCS cluster using CP server as coordination point | On the designated CP server, perform the following tasks:<br>1 Prepare to configure the new CP server.<br>2 Configure the new CP server.<br>On the VCS cluster nodes, configure server-based I/O fencing.<br>See the <i>Veritas Cluster Server Installation Guide</i> for the procedures. |

**Table 10-3** CP server deployment and migration scenarios

| Scenario                                                                              | CP server                          | VCS cluster                                                | Action required                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|---------------------------------------------------------------------------------------|------------------------------------|------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Add a new VCS cluster to an existing and operational CP server                        | Existing and operational CP server | New VCS cluster                                            | On the VCS cluster nodes, configure server-based I/O fencing.<br>See the <i>Veritas Cluster Server Installation Guide</i> for the procedures.                                                                                                                                                                                                                                                                                                                                                                                                          |
| Replace the coordination point from an existing CP server to a new CP server          | New CP server                      | Existing VCS cluster using CP server as coordination point | On the designated CP server, perform the following tasks:<br><ol style="list-style-type: none"> <li>1 Prepare to configure the new CP server.</li> <li>2 Configure the new CP server.</li> <li>3 Prepare the new CP server for use by the VCS cluster</li> </ol> See the <i>Veritas Cluster Server Installation Guide</i> for the procedures.<br>On the VCS cluster nodes run the vxfsnwap command to move to replace the CP server.<br>See <a href="#">“Replacing coordination points for server-based fencing in an online cluster”</a> on page 355. |
| Replace the coordination point from an existing CP server to an operational CP server | Operational CP server              | Existing VCS cluster using CP server as coordination point | On the designated CP server, prepare to configure the new CP server manually.<br>See the <i>Veritas Cluster Server Installation Guide</i> for the procedures.<br>On the VCS cluster run the vxfsnwap command to move to replace the CP server:<br>See <a href="#">“Replacing coordination points for server-based fencing in an online cluster”</a> on page 355.                                                                                                                                                                                       |

Table 10-3 CP server deployment and migration scenarios

| Scenario                                                                                                                           | CP server             | VCS cluster                                                            | Action required                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|------------------------------------------------------------------------------------------------------------------------------------|-----------------------|------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Enabling fencing in a VCS cluster with a new CP server<br><br>Note: This procedure incurs application downtime on the VCS cluster. | New CP server         | Existing VCS cluster with fencing configured in disabled or scsi3 mode | <p>On the designated CP server, perform the following tasks:</p> <ol style="list-style-type: none"><li>1 Prepare to configure the new CP server.</li><li>2 Configure the new CP server.</li><li>3 Prepare the new CP server for use by the VCS cluster</li></ol> <p>See the <i>Veritas Cluster Server Installation Guide</i> for the procedures.</p> <p>On the VCS cluster nodes, perform the following:</p> <ol style="list-style-type: none"><li>1 Stop all applications, VCS, and fencing on the VCS cluster.</li><li>2 To stop VCS, use the following command (to be run on all the VCS cluster nodes):<br/>hastop -local</li><li>3 Stop fencing using the following command:<br/># <b>/etc/init.d/vxfen.rc stop</b></li><li>4 Reconfigure I/O fencing on the VCS cluster.<br/>See the <i>Veritas Cluster Server Installation Guide</i> for the procedures.</li></ol> |
| Refreshing registrations of VCS cluster nodes on coordination points without incurring application downtime                        | Operational CP server | Existing VCS cluster using the CP server as coordination point         | <p>On the VCS cluster run the vxfsnwap command to refresh the keys on the CP server.</p> <p>See “<a href="#">Refreshing registration keys on the coordination points for server-based fencing</a>” on page 357.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |



# Controlling VCS behavior

- [About VCS behavior on resource faults](#)
- [Controlling VCS behavior at the service group level](#)
- [Controlling VCS behavior at the resource level](#)
- [Changing agent file paths and binaries](#)
- [VCS behavior on loss of storage connectivity](#)
- [Service group workload management](#)
- [Sample configurations depicting workload management](#)

# About VCS behavior on resource faults

VCS considers a resource faulted in the following situations:

- When the resource state changes unexpectedly. For example, an online resource going offline.
- When a required state change does not occur. For example, a resource failing to go online or offline when commanded to do so.

In many situations, VCS agents take predefined actions to correct the issue before reporting resource failure to the engine. For example, the agent may try to bring a resource online several times before declaring a fault.

When a resource faults, VCS takes automated actions to “clean up the faulted resource. The Clean function makes sure the resource is completely shut down before bringing it online on another node. This prevents concurrency violations.

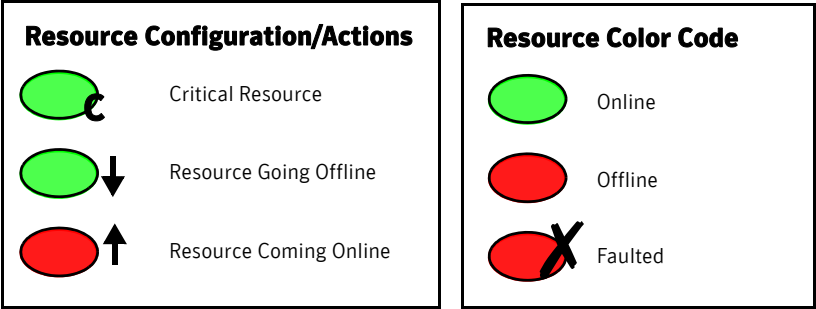
When a resource faults, VCS takes all resources dependent on the faulted resource offline. The fault is thus propagated in the service group

## About critical and non-critical resources

The Critical attribute for a resource defines whether a service group fails over when the resource faults. If a resource is configured as non-critical (by setting the Critical attribute to 0) and no resources depending on the failed resource are critical, the service group will not fail over. VCS takes the failed resource offline and updates the group status to ONLINE|PARTIAL. The attribute also determines whether a service group tries to come online on another node if, during the group’s online process, a resource fails to come online.

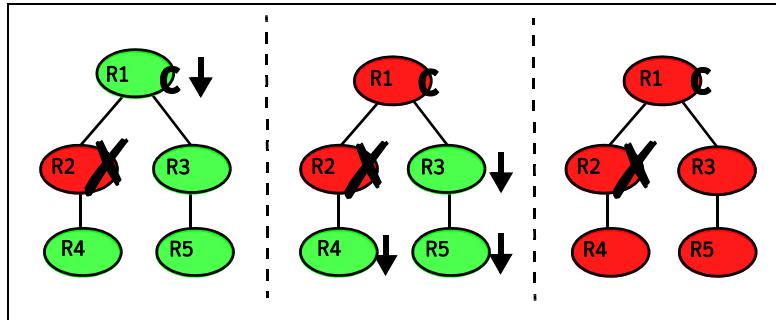
## VCS behavior diagrams

This section describes the default functionality of VCS when resources fault. The illustration displays the symbols used in this section.



### Scenario: Resource with critical parent faults

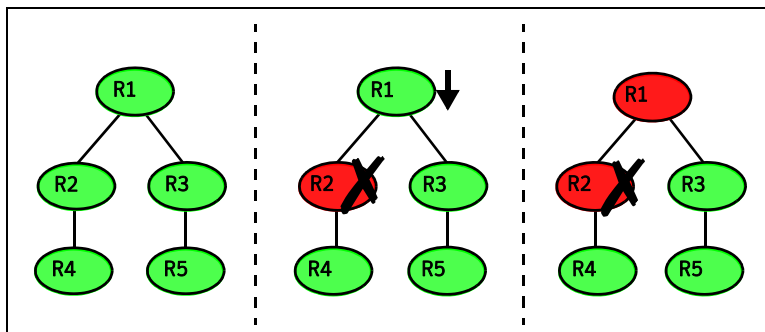
The service group in the following example has five resources, of which resource R1 is configured as a critical resource.



When resource R2 faults, the fault is propagated up the dependency tree to resource R1. When the critical resource R1 goes offline, VCS must fault the service group and fail it over elsewhere in the cluster. VCS takes other resources in the service group offline in the order of their dependencies. After taking resources R3, R4, and R5 offline, VCS fails over the service group to another node.

### Scenario: Resource with non-critical parent faults

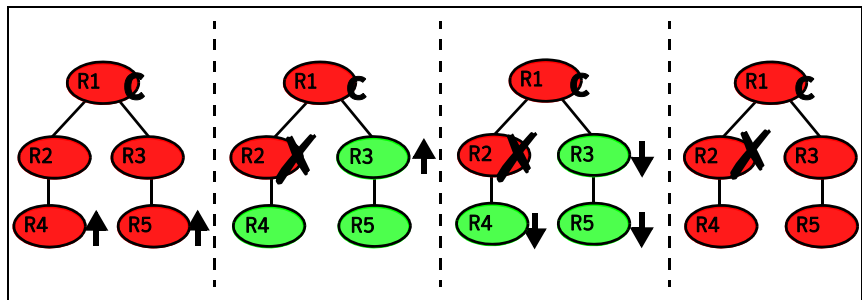
The service group in the following example does not have any critical resources.



When resource R2 faults, the engine propagates the failure up the dependency tree. Neither resource R1 nor resource R2 are critical, so the fault does not result in the tree going offline or in service group failover.

Scenario: Resource with critical parent fails to come online

In the following example, when a command is issued to bring the service group online, resource R2 fails to come online.



VCS calls the Clean function for resource R2 and propagates the fault up the dependency tree. Resource R1 is set to critical, so the service group is taken offline and failed over to another node in the cluster.

# Controlling VCS behavior at the service group level

This section describes how you can configure service group attributes to modify VCS behavior in response to resource faults.

## About the AutoRestart attribute

If a persistent resource on a service group (GROUP\_1) faults, VCS fails the service group over to another system if the following conditions are met:

- The AutoFailOver attribute is set.
- Another system in the cluster exists to which GROUP\_1 can fail over.

If neither of these conditions is met, GROUP\_1 remains offline and faulted, even after the faulted resource becomes online.

Setting the AutoRestart attribute enables a service group to be brought back online without manual intervention. If no failover targets are available, setting the AutoRestart attribute enables VCS to bring the group back online on the first available system after the group's faulted resource came online on that system.

For example, NIC is a persistent resource. In some cases, when a system boots and VCS starts, VCS probes all resources on the system. When VCS probes the NIC resource, the resource may not be online because the networking is not up and fully operational. In such situations, VCS marks the NIC resource as faulted, and does not bring the service group online. However, when the NIC resource becomes online and if AutoRestart is enabled, the service group is brought online.

## Controlling failover on service group or system faults

The AutoFailOver attribute configures service group behavior in response to service group and system faults.

The possible values include 0, 1, and 2. You can set the value of this attribute as 2 if you have enabled the HA/DR license and if the service group is a non-hybrid service group.

0 VCS does not fail over the service group when a system or service group faults.

If a fault occurs in a service group, the group is taken offline, depending on whether any of its resources are configured as critical. If a system faults, the service group is not failed over to another system.

- 1

VCS automatically fails over the service group when a system or a service group faults, provided a suitable node exists for failover.

The service group attributes `SystemZones` and `FailOverPolicy` impact the failover behavior of the service group. For global clusters, the failover decision is also based on the `ClusterFailOverPolicy`.

See [“Service group attributes”](#) on page 663.
- 2

VCS automatically fails over the service group only if another suitable node exists in the same system zone.

If a suitable node does not exist in the same system zone, VCS brings the service group offline, and generates an alert for administrator's intervention. You can manually bring the group online using the `hagrp -online` command.

**Note:** If `SystemZones` attribute is not defined, the failover behavior is similar to `AutoFailOver=1`.

## Defining failover policies

The service group attribute `FailOverPolicy` governs how VCS calculates the target system for failover. There are three possible values for `FailOverPolicy`:

- Priority

VCS selects the system with the lowest priority as the failover target. The Priority failover policy is ideal for simple two-node clusters or small clusters with few service groups.

Priority is set in the `SystemList` attribute implicitly via ordering, such as `SystemList = {SystemA, SystemB}` or explicitly, such as `SystemList = {SystemA=0, SystemB=1}`. Priority is the default behavior.
- RoundRobin

VCS selects the system running the fewest service groups as the failover target. This policy is ideal for large clusters running many service groups with similar server load characteristics (for example, similar databases or applications)
- Load

The Load failover policy comprises the following components:

System capacity and service group load, represented by the attributes `Capacity` and `Load` respectively.

System limits and service group prerequisites, represented by the attributes `Limits` and `Prerequisites`, respectively.

## About system zones

The `SystemZones` attribute enables you to create a subset of systems to use in an initial failover decision. This feature allows fine-tuning of application failover decisions, and yet retains the flexibility to fail over anywhere in the cluster.

If the attribute is configured, a service group tries to stay within its zone before choosing a host in another zone. For example, in a three-tier application infrastructure with Web, application, and database servers, you could create two system zones: one each for the application and the database. In the event of a failover, a service group in the application zone will try to fail over to another node within the zone. If no nodes are available in the application zone, the group will fail over to the database zone, based on the configured load and limits.

In this configuration, excess capacity and limits on the database backend are kept in reserve to handle the larger load of a database failover. The application servers handle the load of service groups in the application zone. During a cascading failure, the excess capacity in the cluster is available to all service groups.

## About load-based autostart

VCS provides a method to determine where a service group comes online when the cluster starts. Setting the `AutoStartPolicy` to `Load` instructs the VCS engine, HAD, to determine the best system on which to start the groups. VCS places service groups in an `AutoStart` queue for load-based startup as soon as the groups probe all running systems. VCS creates a subset of systems that meet all prerequisites and then chooses the system with the highest `AvailableCapacity`.

Set `AutoStartPolicy = Load` and configure the `SystemZones` attribute to establish a list of preferred systems on which to initially run a group.

## Freezing service groups

Freezing a service group prevents VCS from taking any action when the service group or a system faults. Freezing a service group prevents dependent resources from going offline when a resource faults. It also prevents the `Clean` function from being called on a resource fault.

You can freeze a service group when performing operations on its resources from outside VCS control. This prevents VCS from taking actions on resources while your operations are on. For example, freeze a database group when using database controls to stop and start a database.

## Controlling Clean behavior on resource faults

The ManageFaults attribute specifies whether VCS calls the Clean function when a resource in the service group faults. ManageFaults is a service group attribute; you can configure each service group to operate as desired.

- If the ManageFaults attribute is set to ALL, VCS calls the Clean function when a resource faults.
- If the ManageFaults attribute is set to NONE, VCS takes no action on a resource fault; it hangs the service group until administrative action can be taken. VCS marks the resource state as ADMIN\_WAIT and does not fail over the service group until the resource fault is removed and the ADMIN\_WAIT state is cleared.

VCS calls the resadminwait trigger when a resource enters the ADMIN\_WAIT state due to a resource fault if the ManageFaults attribute is set to NONE. You can customize this trigger to provide notification about the fault. See “[resadminwait event trigger](#)” on page 456.

When ManageFaults is set to none and one of the following events occur, the resource enters the admin\_wait state:

| Event                                                                                                  | Resource state                         |
|--------------------------------------------------------------------------------------------------------|----------------------------------------|
| The offline function did not complete within the expected time.                                        | ONLINE ADMIN_WAIT                      |
| The offline function was ineffective.                                                                  | ONLINE ADMIN_WAIT                      |
| The online function did not complete within the expected time.                                         | OFFLINE ADMIN_WAIT                     |
| The online function was ineffective.                                                                   | OFFLINE ADMIN_WAIT                     |
| The resource was taken offline unexpectedly.                                                           | OFFLINE ADMIN_WAIT                     |
| For the online resource the monitor function consistently failed to complete within the expected time. | ONLINE MONITOR_TIM<br>EDOUT ADMIN_WAIT |



## Clearing resources in the ADMIN\_WAIT state

When VCS sets a resource in the ADMIN\_WAIT state, it invokes the resadminwait trigger according to the reason the resource entered the state.

See “[resadminwait event trigger](#)” on page 456.

### To clear a resource

- 1 Take the necessary actions outside VCS to bring all resources into the required state.
- 2 Verify that resources are in the required state by issuing the command:  

```
hagrp -clearadminwait group -sys system
```

This command clears the ADMIN\_WAIT state for all resources. If VCS continues to detect resources that are not in the required state, it resets the resources to the ADMIN\_WAIT state.
- 3 If resources continue in the ADMIN\_WAIT state, repeat [step 1](#) and [step 2](#), or issue the following command to stop VCS from setting the resource to the ADMIN\_WAIT state:  

```
hagrp -clearadminwait -fault group -sys system
```

This command has the following results:

  - If the resadminwait trigger was called for reasons 0 or 1, the resource state is set as ONLINE|UNABLE\_TO\_OFFLINE.
  - If the resadminwait trigger was called for reasons 2, 3, or 4, the resource state is set as FAULTED. Please note that when resources are set as FAULTED for these reasons, the clean function is not called. Verify that resources in ADMIN-WAIT are in clean, OFFLINE state prior to invoking this command.

When a service group has a resource in the ADMIN\_WAIT state, the following service group operations cannot be performed on the resource: online, offline, switch, and flush. Also, you cannot use the hastop command when resources are in the ADMIN\_WAIT state. When this occurs, you must issue the hastop command with -force option only.

## Controlling fault propagation

The `FaultPropagation` attribute defines whether a resource fault is propagated up the resource dependency tree. It also defines whether a resource fault causes a service group failover.

- If the `FaultPropagation` attribute is set to 1 (default), a resource fault is propagated up the dependency tree. If a resource in the path is critical, the service group is taken offline and failed over, provided the `AutoFailOver` attribute is set to 1.
- If the `FaultPropagation` is set to 0, resource faults are contained at the resource level. VCS does not take the dependency tree offline, thus preventing failover. If the resources in the service group remain online, the service group remains in the `PARTIAL|FAULTED` state. If all resources are offline or faulted, the service group remains in the `OFFLINE| FAULTED` state.

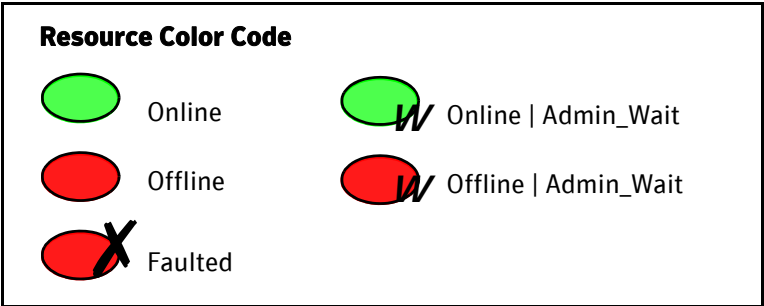
You can set this attribute for a service group or for a resource. The default value for this attribute is 1.

When a resource faults, VCS fires the `resfault` trigger and sends an SNMP trap. The trigger is called on the system where the resource faulted and includes the name of the faulted resource.

See “[resfault event trigger](#)” on page 457.

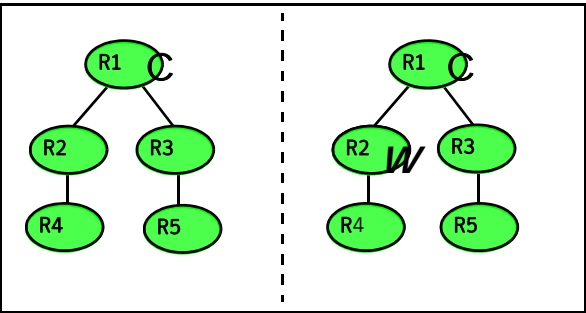
## Customized behavior diagrams

The illustrations in this section depict how the ManageFaults and FaultPropagation attributes change VCS behavior when handling resource faults. The following illustration depicts the legends used in the section.



### Scenario: Resource with a critical parent and ManageFaults=NONE

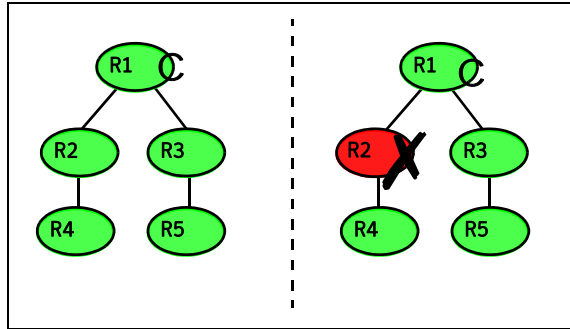
The service group in the following example has five resources. The ManageFaults attribute is set to NONE for resource R2.



If resource R2 fails, the resource is marked as ONLINE|ADMIN\_WAIT. The Clean function is not called for the resource. VCS does not take any other resource offline.

## Scenario: Resource with a critical parent and FaultPropagation=0

In the following example, the FaultPropagation attribute is set to 0.



When resource R2 faults, the Clean function is called and the resource is marked as faulted. The fault is not propagated up the tree, and the group is not taken offline.

## VCS behavior for resources that support the intentional offline functionality

Certain agents can identify when an application has been intentionally shut down outside of VCS control.

For agents that support this functionality, if an administrator intentionally shuts down an application outside of VCS control, VCS does not treat it as a fault. VCS sets the service group state as offline or partial, depending on the state of other resources in the service group.

This feature allows administrators to stop applications without causing a failover. The feature is available for V51 agents.

### About the IntentionalOffline attribute

To configure a resource to recognize an intentional offline of configured application, set the IntentionalOffline attribute to 1. Set the attribute to its default value of 0 to disable this functionality.

- If you set the attribute to 1: When the application is intentionally stopped outside of VCS control, the resource enters an OFFLINE state. This attribute does not affect VCS behavior on application failure. VCS continues to fault resources if managed corresponding applications fail.

- If you set the attribute to 0: When the application is intentionally stopped outside of VCS control, the resource enters a FAULTED state.

### About the ExternalStateChanged attribute

Use the ExternalStateChanged attribute to control service group behavior in response to a configured application is intentionally started or stopped outside of VCS control.

The attribute defines how VCS handles service group state when resources are intentionally brought online or taken offline outside of VCS control.

The attribute can take the following values:

- OnlineGroup: If the configured application is started outside of VCS control, VCS brings the corresponding service group online. If you attempt to start the application on a frozen node or service group, VCS brings the corresponding service group online once the node or service group is unfrozen.
- OfflineGroup: If the configured application is stopped outside of VCS control, VCS takes the corresponding service group offline.
- OfflineHold: If a configured application is stopped outside of VCS control, VCS sets the state of the corresponding VCS resource as offline. VCS does not take any parent resources or the service group offline.

OfflineHold and OfflineGroup are mutually exclusive.

## Controlling VCS behavior at the resource level

This section describes how you can control VCS behavior at the resource level. Note that a resource is not considered faulted until the agent framework declares the fault to the VCS engine.

### About resource type attributes that control resource behavior

The following attributes affect how the VCS agent framework reacts to problems with individual resources before informing the fault to the VCS engine.

#### About the RestartLimit attribute

The RestartLimit attribute defines whether VCS attempts to restart a failed resource before informing the engine of the fault.

If the RestartLimit attribute is set to a non-zero value, the agent attempts to restart the resource before declaring the resource as faulted. When restarting a failed resource, the agent framework calls the Clean function before calling the

Online function. However, setting the `ManageFaults` attribute to `NONE` prevents the `Clean` function from being called and prevents the `Online` function from being retried.

### About the `OnlineRetryLimit` attribute

The `OnlineRetryLimit` attribute specifies the number of times the `Online` function is retried if the initial attempt to bring a resource online is unsuccessful.

When the `OnlineRetryLimit` set to a non-zero value, the agent framework calls the `Clean` function before rerunning the `Online` function. Setting the `ManageFaults` attribute to `NONE` prevents the `Clean` function from being called and also prevents the `Online` operation from being retried.

### About the `ConfInterval` attribute

The `ConfInterval` attribute defines how long a resource must remain online without encountering problems before previous problem counters are cleared. The attribute controls when VCS clears the `RestartCount`, `ToleranceCount` and `CurrentMonitorTimeoutCount` values.

### About the `ToleranceLimit` attribute

The `ToleranceLimit` attribute defines the number of times the `Monitor` routine should return an offline status before declaring a resource offline. This attribute is typically used when a resource is busy and appears to be offline. Setting the attribute to a non-zero value instructs VCS to allow multiple failing monitor cycles with the expectation that the resource will eventually respond. Setting a non-zero `ToleranceLimit` also extends the time required to respond to an actual fault.

### About the `FaultOnMonitorTimeouts` attribute

The `FaultOnMonitorTimeouts` attribute defines whether VCS interprets a `Monitor` function timeout as a resource fault.

If the attribute is set to 0, VCS does not treat `Monitor` timeouts as a resource faults. If the attribute is set to 1, VCS interprets the timeout as a resource fault and the agent calls the `Clean` function to shut the resource down.

By default, the `FaultOnMonitorTimeouts` attribute is set to 4. This means that the `Monitor` function must time out four times in a row before the resource is marked faulted.

## How VCS handles resource faults

This section describes the process VCS uses to determine the course of action when a resource faults.

### VCS behavior when an online resource faults

In the following example, a resource in an online state is reported as being offline without being commanded by the agent to go offline.

- VCS first verifies the Monitor routine completes successfully in the required time. If it does, VCS examines the exit code returned by the Monitor routine. If the Monitor routine does not complete in the required time, VCS looks at the FaultOnMonitorTimeouts (FOMT) attribute.
- If FOMT=0, the resource will not fault when the Monitor routine times out. VCS considers the resource online and monitors the resource periodically, depending on the monitor interval.  
If FOMT=1 or more, VCS compares the CurrentMonitorTimeoutCount (CMTC) with the FOMT value. If the monitor timeout count is not used up, CMTC is incremented and VCS monitors the resource in the next cycle.
- If FOMT= CMTC, this means that the available monitor timeout count is exhausted and VCS must now take corrective action.
- If the ManageFaults attribute is set to NONE, VCS marks the resource as ONLINE|ADMIN\_WAIT and fires the resadminwait trigger. If the ManageFaults attribute is set to ALL, the resource enters a GOING OFFLINE WAIT state. VCS invokes the Clean function with the reason *Monitor Hung*.
- If the Clean function is successful (that is, Clean exit code = 0), VCS examines the value of the RestartLimit attribute. If Clean fails (exit code = 1), the resource remains online with the state UNABLE TO OFFLINE. VCS fires the resnotoff trigger and monitors the resource again.
- If the Monitor routine does not time out, it returns the status of the resource as being online or offline.
- If the ToleranceLimit (TL) attribute is set to a non-zero value, the Monitor cycle returns offline (exit code = 100) for a number of times specified by the ToleranceLimit and increments the ToleranceCount (TC). When the ToleranceCount equals the ToleranceLimit (TC = TL), the agent declares the resource as faulted.
- If the Monitor routine returns online (exit code = 110) during a monitor cycle, the agent takes no further action. The ToleranceCount attribute is reset to 0 when the resource is online for a period of time specified by the ConfInterval attribute.

If the resource is detected as being offline a number of times specified by the `ToleranceLimit` before the `ToleranceCount` is reset ( $TC = TL$ ), the resource is considered failed.

- After the agent determines the resource is not online, VCS checks the `Frozen` attribute for the service group. If the service group is frozen, VCS declares the resource faulted and calls the `resfault` trigger. No further action is taken.
- If the service group is not frozen, VCS checks the `ManageFaults` attribute. If `ManageFaults=NONE`, VCS marks the resource state as `ONLINE|ADMIN_WAIT` and calls the `resadminwait` trigger. If `ManageFaults=ALL`, VCS calls the `Clean` function with the `CleanReason` set to `Unexpected Offline`.
- If the `Clean` function fails (exit code = 1) the resource remains online with the state `UNABLE TO OFFLINE`. VCS fires the `resnotoff` trigger and monitors the resource again. The resource enters a cycle of alternating `Monitor` and `Clean` functions until the `Clean` function succeeds or a user intervenes.
- If the `Clean` function is successful, VCS examines the value of the `RestartLimit` (RL) attribute. If the attribute is set to a non-zero value, VCS increments the `RestartCount` (RC) attribute and invokes the `Online` function. This continues till the value of the `RestartLimit` equals that of the `RestartCount`. At this point, VCS attempts to monitor the resource.

If the `Monitor` returns an online status, VCS considers the resource online and resumes periodic monitoring. If the monitor returns an offline status, the resource is faulted and VCS takes actions based on the service group configuration.

## VCS behavior when a resource fails to come online

In the following example, the agent framework invokes the `Online` function for an offline resource. The resource state changes to `WAITING TO ONLINE`.

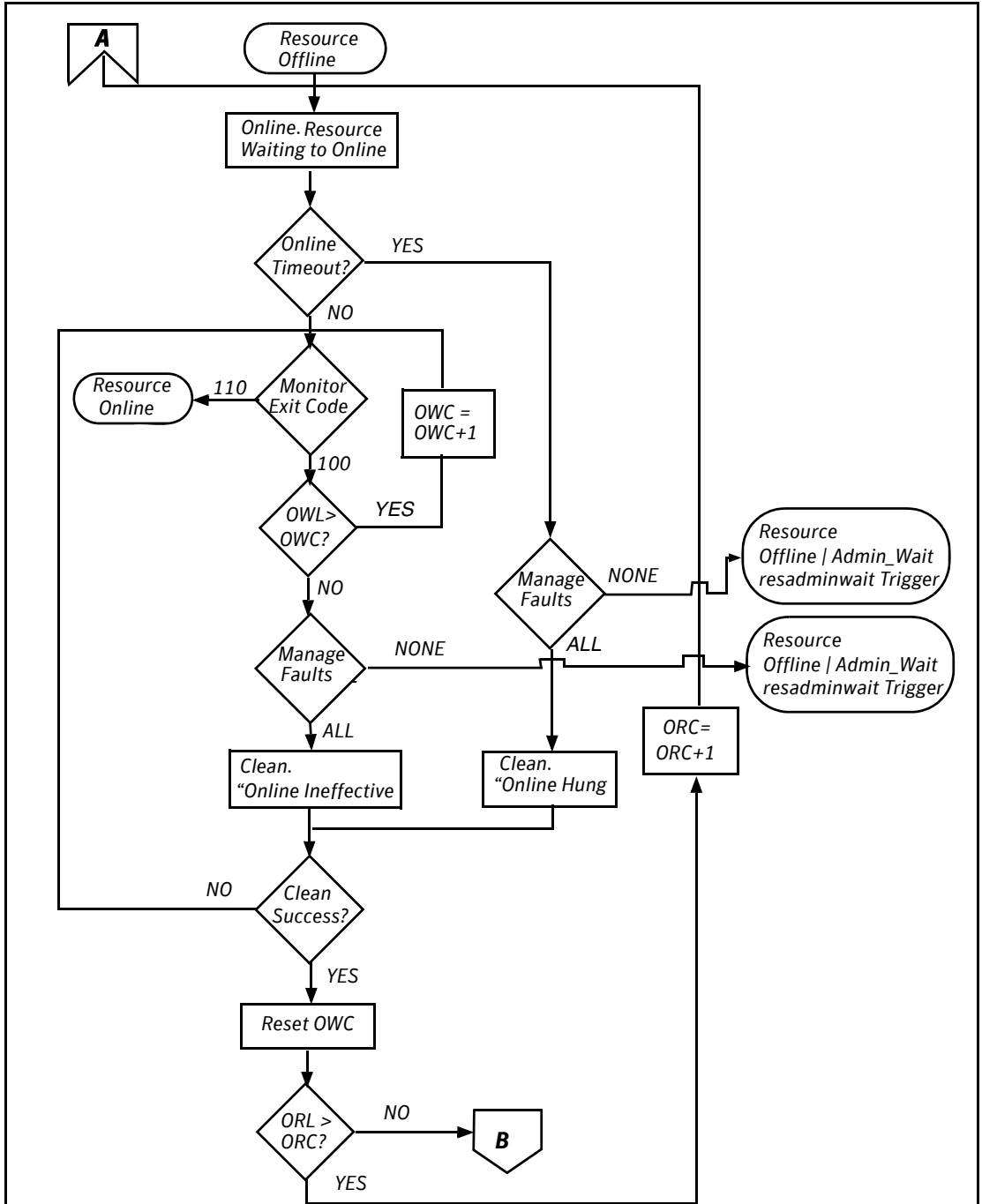
- If the `Online` function times out, VCS examines the value of the `ManageFaults` attribute.
- If `ManageFaults` is set to `NONE`, the resource state changes to `OFFLINE|ADMIN_WAIT`.  
 If `ManageFaults` is set to `ALL`, VCS calls the `Clean` function with the `CleanReason` set to `Online Hung`.
- If the `Online` function does not time out, VCS invokes the `Monitor` function. The `Monitor` routine returns an exit code of 110 if the resource is online. Otherwise, the `Monitor` routine returns an exit code of 100.
- VCS examines the value of the `OnlineWaitLimit` (OWL) attribute. This attribute defines how many monitor cycles can return an offline status



before the agent framework declares the resource faulted. Each successive Monitor cycle increments the OnlineWaitCount (OWC) attribute. When  $OWL = OWC$  (or if  $OWL = 0$ ), VCS determines the resource has faulted.

- VCS then examines the value of the ManageFaults attribute. If the ManageFaults is set to NONE, the resource state changes to OFFLINE|ADMIN\_WAIT.  
If the ManageFaults is set to ALL, VCS calls the Clean function with the CleanReason set to Online Ineffective.
- If the Clean function is not successful (exit code = 1), the agent monitors the resource. It determines the resource is offline, and calls the Clean function with the Clean Reason set to Online Ineffective. This cycle continues till the Clean function is successful, after which VCS resets the OnlineWaitCount value.
- If the OnlineRetryLimit (ORL) is set to a non-zero value, VCS increments the OnlineRetryCount (ORC) and invokes the Online function. This starts the

cycle all over again. If  $ORL = ORC$ , or if  $ORL = 0$ , VCS assumes that the Online operation has failed and declares the resource as faulted.



## VCS behavior after a resource is declared faulted

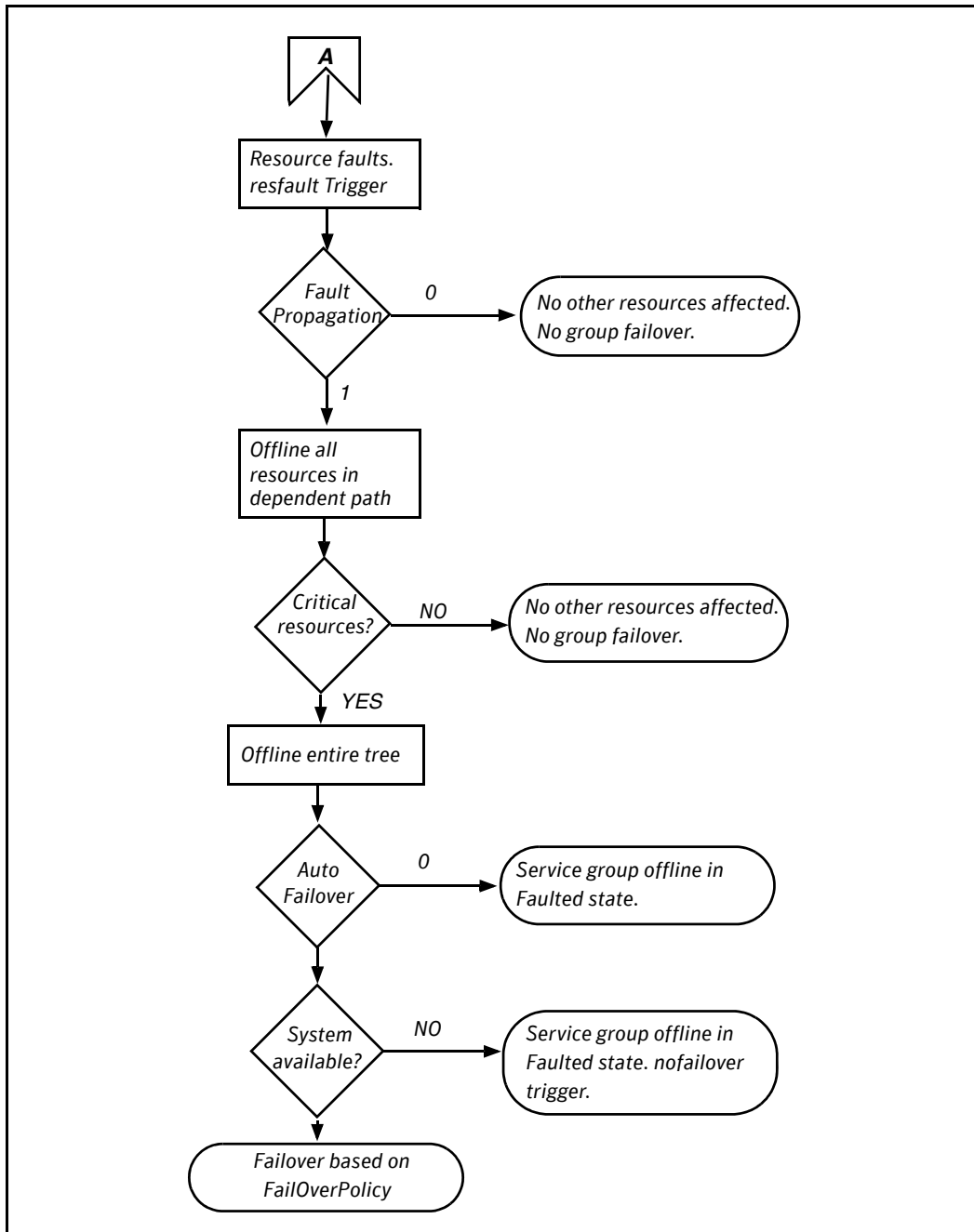
After a resource is declared faulted, VCS fires the resfault trigger and examines the value of the FaultPropagation attribute.

- If FaultPropagation is set to 0, VCS does not take other resources offline, and changes the group state to OFFLINE|FAULTED or PARTIAL|FAULTED. The service group does not fail over.

If FaultPropagation is set to 1, VCS takes all resources in the dependent path of the faulted resource offline, up to the top of the tree.

- VCS then examines if any resource in the dependent path is critical. If no resources are critical, the service group is left in its OFFLINE|FAULTED or PARTIAL|FAULTED state. If a resource in the path is critical, VCS takes the all resources in the service group offline in preparation of a failover.
- If the AutoFailOver attribute is set to 0, the service group is not failed over; it remains in a faulted state. If AutoFailOver is set to 1, VCS examines if any systems in the service group's SystemList are possible candidates for failover. If no suitable systems exist, the group remains faulted and VCS calls the nofailover trigger. If eligible systems are available, VCS examines the FailOverPolicy to determine the most suitable system to which to fail over the service group.

- If FailOverPolicy is set to Load, a NoFailover situation may occur because of restrictions placed on service groups and systems by Service Group Workload Management.



## Disabling resources

Disabling a resource means that the resource is no longer monitored by a VCS agent, and that the resource cannot be brought online or taken offline. The agent starts monitoring the resource after the resource is enabled. The resource attribute `Enabled` determines whether a resource is enabled or disabled. A persistent resource can be disabled when all its parents are offline. A non-persistent resource can be disabled when the resource is in an `OFFLINE` state.

### When to disable a resource

Typically, resources are disabled when one or more resources in the service group encounter problems and disabling the resource is required to keep the service group online or to bring it online.

---

**Note:** Disabling a resource is not an option when the entire service group requires disabling. In that case, set the service group attribute `Enabled` to 0.

---

#### To disable a resource

To disable the resource when VCS is running:

```
hares -modify resource_name Enabled 0
```

To have the resource disabled initially when VCS is started, set the resource's `Enabled` attribute to 0 in `main.cf`.

### Limitations of disabling resources

When VCS is running, there are certain prerequisites to be met before the resource is disabled successfully.

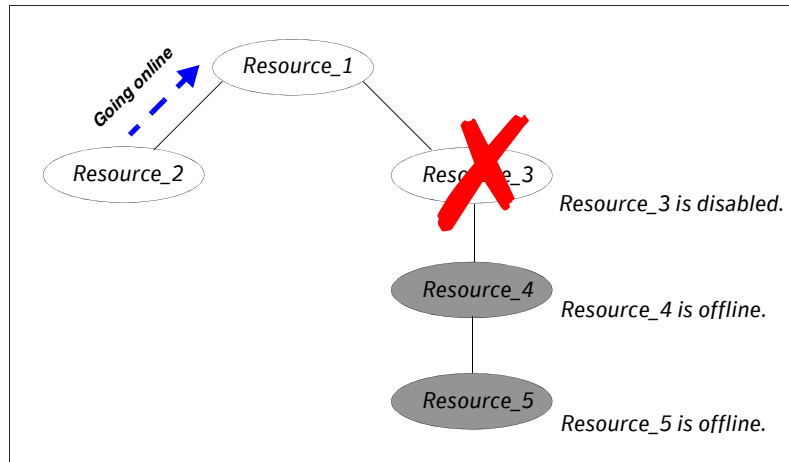
- An online non-persistent resource cannot be disabled. It must be in a clean `OFFLINE` state. (The state must be `OFFLINE` and `IState` must be `NOT WAITING`.)
- If it is a persistent resource and the state is `ONLINE` on some of the systems, all dependent resources (parents) must be in clean `OFFLINE` state. (The state must be `OFFLINE` and `IState` must be `NOT WAITING`)

Therefore, before disabling the resource you may be required to take it offline (if it is non-persistent) and take other resources offline in the service group.

## Additional considerations for disabling resources

- When a group containing disabled resources is brought online, the online transaction is not propagated to the disabled resources. Children of the disabled resource are brought online by VCS only if they are required by another enabled resource.
- You can bring children of disabled resources online if necessary.
- When a group containing disabled resources is taken offline, the offline transaction is propagated to the disabled resources.

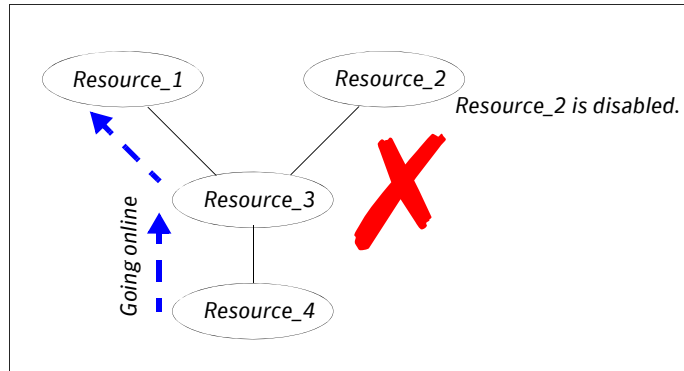
The following figures show how a service group containing disabled resources is brought online.



In the figure above, Resource\_3 is disabled. When the service group is brought online, the only resources brought online by VCS are Resource\_1 and Resource\_2 (Resource\_2 is brought online first) because VCS recognizes Resource\_3 is disabled. In accordance with online logic, the transaction is not propagated to the disabled resource.



In the figure below, Resource\_2 is disabled. When the service group is brought online, resources 1, 3, 4 are also brought online (Resource\_4 is brought online first). Note Resource\_3, the child of the disabled resource, is brought online because Resource\_1 is enabled and is dependent on it.



## How disabled resources affect group states

When a service group is brought online containing non-persistent, disabled resources whose AutoStart attributes are set to 1, the group state is PARTIAL, even though enabled resources with Autostart=1 are online. This is because the disabled resource is considered for the group state.

To have the group in the ONLINE state when enabled resources with AutoStart set to 1 are in ONLINE state, set the AutoStart attribute to 0 for the disabled, non-persistent resources.

## Changing agent file paths and binaries

By default, VCS runs agent binaries from the path `$VCS_HOME/bin/AgentName/AgentNameAgent`. For example, `/opt/VRTSvcs/bin/FileOnOff/FileOnOffAgent`.

You can instruct VCS to run a different set of agent binaries or scripts by specifying values for the `AgentFile` and `AgentDirectory` attributes.

- **AgentFile**— Specify a value for this attribute if the name of the agent binary is not the same as that of the resource type.

For example, if the resource type is `MyApplication` and the agent binary is called `MyApp`, set the `AgentFile` attribute to `MyApp`. For a script-based agent, you could configure `AgentFile` as `/opt/VRTSvcs/bin/ScriptAgent`.

- **AgentDirectory**— Specify a value for this attribute if the agent is not installed at the default location.

When you specify the agent directory, VCS looks for the agent file (*AgentNameAgent*) in the agent directory. If the agent file name does not conform to the *AgentNameAgent* convention, configure the `AgentFile` attribute.

For example, if the `MyApplication` agent is installed at `/opt/VRTSvcs/bin/CustomAgents/MyApplication`, specify this path as the attribute value. If the agent file is not named `MyApplicationAgent`, configure the `AgentFile` attribute.

If you do not set these attributes and the agent is not available at its default location, VCS looks for the agent at the `/opt/VRTSagents/ha/bin/AgentName/AgentNameAgent`.

### To change the path of an agent

- ◆ Before configuring a resource for the agent, add `AgentFile` and `AgentDirectory` as static attributes to the agent's resource type.

```
haattr -add -static resource_type AgentFile \
 "binary_name"
haattr -add -static resource_type AgentDirectory \
 "complete_path_to_agent_binary"
```

# VCS behavior on loss of storage connectivity

When a node loses connectivity to shared storage, input-output operations (I/O) to volumes return errors and the disk group gets disabled. In this situation, VCS must fail the service groups over to another node. This failover is to ensure that applications have access to shared storage. The failover involves deporting disk groups from one node and importing them to another node. However, pending I/O operations must complete before the disabled disk group can be deported.

Pending I/O operations cannot complete without storage connectivity. VCS assumes data is being read from or written to disks and does not declare the DiskGroup resource as offline. This behavior prevents potential data corruption that may be caused by the disk group being imported on two hosts. However, this also means that service groups remain online on a node that does not have storage connectivity and the service groups cannot be failed over unless an administrator intervenes. This affects application availability.

Some Fibre Channel (FC) drivers have a configurable parameter called *failover*, which defines the number of seconds for which the driver retries I/O commands before returning an error. If you set the failover parameter to 0, the FC driver retries I/O infinitely and *does not return an error* even when storage connectivity is lost. This also causes the Monitor function for the DiskGroup to time out and prevents failover of the service group unless an administrator intervenes.

## About disk group configuration and VCS behavior

Table 11-4 describes how the disk group state and the failover attribute define VCS behavior when a node loses connectivity to shared storage.

Table 11-4 Disk group state and VCS behavior

| Case | DiskGroup State | Failover Attribute | VCS Behavior on Loss of Storage Connectivity                           |
|------|-----------------|--------------------|------------------------------------------------------------------------|
| 1    | Enabled         | N seconds          | Fails over service groups to another node.                             |
| 2    | Disabled        | N seconds          | DiskGroup resource remains online.<br>No failover.                     |
| 3    | Enabled         | 0                  | DiskGroup resource remains in monitor timed out state.<br>No failover. |
| 4    | Disabled        | 0                  | DiskGroup resource remains online.<br>No failover.                     |

## How VCS attributes control behavior on loss of storage connectivity

If you use I/O fencing, you can configure VCS attributes to ensure that a node panics on losing connectivity to shared storage. The panic causes service groups to fail over to another node.

A system reboot or shutdown could leave the system in a hung state because the operating system cannot dump the buffer cache to the disk. The panic operation ensures that VCS does not wait for I/Os to complete before triggering the failover mechanism, thereby ensuring application availability. However, you might have to perform a file system check when you restart the node.

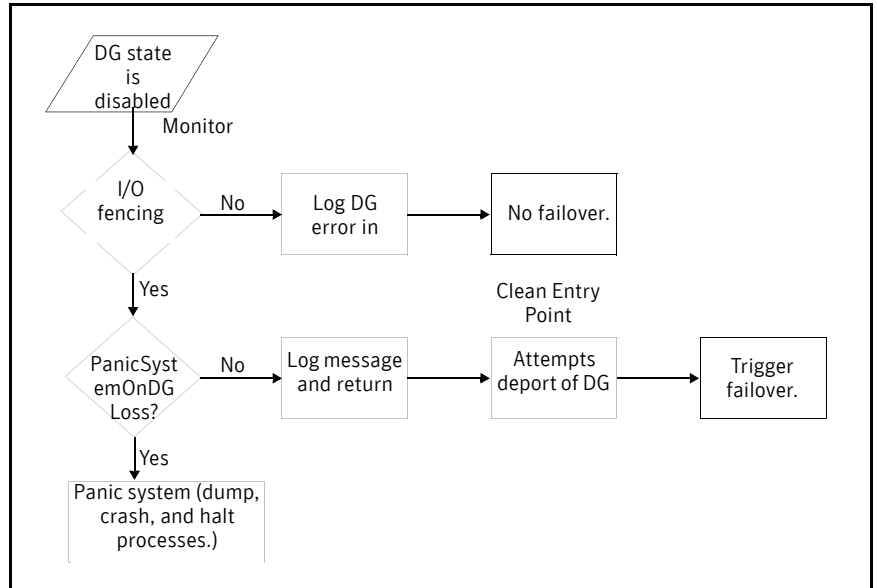
The following attributes define VCS behavior on loss of storage connectivity:–

|                        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PanicSystemOnDGLoss    | Applies to the DiskGroup resource and defines whether the agent panics the system when storage connectivity is lost.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| FaultOnMonitorTimeouts | <p>The number of consecutive monitor timeouts after which VCS calls the Clean function to mark the resource as FAULTED or restarts the resource.</p> <p>If you set the attribute to 0, VCS does not treat a Monitor timeout as a resource fault. By default, the attribute is set to 4. This means that the Monitor function must time out four times in a row before VCS marks the resource faulted.</p> <p>When the Monitor function for the DiskGroup agent times out, (case 3 in the table above), the FaultOnMonitorTimeouts attribute defines when VCS interprets the resource as faulted and invokes the Clean function. If the CleanReason is "monitor hung", the system panics.</p> |

## About VCS behavior when a disk group is disabled

Figure 11-10 describes VCS behavior for a disabled diskgroup.

**Figure 11-10** VCS behavior when a disk group is disabled



## Recommendations to ensure application availability

Symantec makes the following recommendations to ensure application availability and data integrity when a node loses connectivity to shared storage.

- Do not set the failover attribute for the FC driver to 0. However, if you do set the failover attribute to 0, set the FaultOnMonitorTimeouts value for the DiskGroup resource type to a finite value.
- If you use I/O fencing, set the PanicSystemOnDGLoss attribute for the DiskGroup resource to 1. This ensures that the system panics when it loses connectivity to shared storage, and causes applications to fail over to another node. The failover ensures application availability, while I/O fencing ensures data integrity.

## Service group workload management

Workload management is a load-balancing mechanism that determines which system hosts an application during startup, or after an application or server fault.

Service Group Workload Management provides tools for making intelligent decisions about startup and failover locations, based on system capacity and resource availability.

### Enabling service group workload management

The service group attribute `FailOverPolicy` governs how VCS calculates the target system for failover. Set `FailOverPolicy` to `Load` to enable service group workload management.

See “[Controlling VCS behavior at the resource level](#)” on page 373

### About system capacity and service group load

The `Load` and `Capacity` construct allows the administrator to define a fixed amount of resources a server provides (`Capacity`), and a fixed amount of resources a specific service group is expected to utilize (`Load`).

The system attribute `Capacity` sets a fixed load-handling capacity for servers. Define this attribute based on system requirements.

The service group attribute `Load` sets a fixed demand for service groups. Define this attribute based on application requirements.

When a service group is brought online, its load is subtracted from the system’s capacity to determine available capacity. VCS maintains this info in the attribute `AvailableCapacity`.

When a failover occurs, VCS determines which system has the highest available capacity and starts the service group on that system. During a failover involving multiple service groups, VCS makes failover decisions serially to facilitate a proper load-based choice.

System capacity is a *soft* restriction; in some situations, value of the `Capacity` attribute could be less than zero. During some operations, including cascading failures, the value of the `AvailableCapacity` attribute could be negative.

### Static load versus dynamic load

Dynamic load is an integral component of the Service Group Workload Management framework. Typically, HAD sets remaining capacity with the function:

$\text{AvailableCapacity} = \text{Capacity} - (\text{sum of Load values of all online service groups})$

If the `DynamicLoad` attribute is defined, its value overrides the calculated Load values with the function:

$\text{AvailableCapacity} = \text{Capacity} - \text{DynamicLoad}$

This enables better control of system loading values than estimated service group loading (static load). However, this requires setting up and maintaining a load estimation package outside VCS. It also requires modifying the configuration file `main.cf` manually.

Note that the `DynamicLoad` (specified with `hasys -load`) is subtracted from the Capacity as an integer and not a percentage value. For example, if a system's capacity is 200 and the load estimation package determines the server is 80 percent loaded, it must inform VCS that the `DynamicLoad` value is 160 (not 80).

## About overload warning

Overload warning provides the notification component of the Load policy. When a server sustains the preset load level (set by the attribute `LoadWarningLevel`) for a preset time (set by the attribute `LoadTimeThreshold`), VCS invokes the loadwarning trigger.

See [“Using event triggers”](#) on page 446

See [“System attributes”](#) on page 680.

The loadwarning trigger is a user-defined script or application designed to carry out specific actions. It is invoked once, when system load exceeds the `LoadWarningLevel` for the `LoadTimeThreshold`. It is not invoked again until the `LoadTimeCounter`, which determines how many seconds system load has been above `LoadWarningLevel`, is reset.

## About system limits and service group prerequisites

`Limits` is a system attribute and designates which resources are available on a system, including shared memory segments and semaphores.

`Prerequisites` is a service group attribute and helps manage application requirements. For example, a database may require three shared memory segments and 10 semaphores. VCS Load policy determines which systems meet the application criteria and then selects the least-loaded system.

If the prerequisites defined for a service group are not met on a system, the service group cannot be brought online on the system.

When configuring these attributes, define the service group's prerequisites first, then the corresponding system limits. Each system can have a different limit

and there is no cap on the number of group prerequisites and system limits. Service group prerequisites and system limits can appear in any order.

You can also use these attributes to configure the cluster as N-to-1 or N-to-N. For example, to ensure that only one service group can be online on a system at a time, add the following entries to the definition of each group and system:

```
Prerequisites = { GroupWeight = 1 }
Limits = { GroupWeight = 1 }
```

System limits and group prerequisites work independently of FailOverPolicy. Prerequisites determine the eligible systems on which a service group can be started. When a list of systems is created, HAD then follows the configured FailOverPolicy.

## Using capacity and limits

When selecting a node as a failover target, VCS selects the system that meets the service group's prerequisites and has the highest available capacity. If multiple systems meet the prerequisites and have the same available capacity, VCS selects the system appearing lexically first in the SystemList.

Systems having an available capacity of less than the percentage set by the LoadWarningLevel attribute, and those remaining at that load for longer than the time specified by the LoadTimeThreshold attribute invoke the loadwarning trigger.



# Sample configurations depicting workload management

This section lists some sample configurations that use the concepts described in this chapter.

## System and Service group definitions

The main.cf in this example shows various Service Group Workload Management attributes in a system definition and a service group definition. See “[VCS attributes](#)” on page 643.

```
include "types.cf"
cluster SGWM-demo (
)

system LargeServer1 (
 Capacity = 200
 Limits = { ShrMemSeg=20, Semaphores=10, Processors=12 }
 LoadWarningLevel = 90
 LoadTimeThreshold = 600
)

group G1 (
 SystemList = { LargeServer1, LargeServer2, MedServer1,
 MedServer2 }
 SystemZones = { LargeServer1=0, LargeServer2=0,
 MedServer1=1, MedServer2=1 }
 AutoStartPolicy = Load
 AutoStartList = { MedServer1, MedServer2 }
 FailOverPolicy = Load
 Load = 100
 Prerequisites = { ShrMemSeg=10, Semaphores=5, Processors=6 }
)
```

## Sample configuration: Basic four-node cluster

```
include "types.cf"
cluster SGWM-demo

system Server1 (
 Capacity = 100
)

system Server2 (
 Capacity = 100
)

system Server3 (
 Capacity = 100
)

system Server4 (
 Capacity = 100
)

group G1 (
 SystemList = { Server1, Server2, Server3, Server4 }
 AutoStartPolicy = Load
 AutoStartList = { Server1, Server2, Server3, Server4 }
 FailOverPolicy = Load
 Load = 20
)

group G2 (
 SystemList = { Server1, Server2, Server3, Server4 }
 AutoStartPolicy = Load
 AutoStartList = { Server1, Server2, Server3, Server4 }
 FailOverPolicy = Load
 Load = 40
)

group G3 (
 SystemList = { Server1, Server2, Server3, Server4 }
 AutoStartPolicy = Load
 AutoStartList = { Server1, Server2, Server3, Server4 }
 FailOverPolicy = Load
 Load = 30
)
```

```
group G4 (
 SystemList = { Server1, Server2, Server3, Server4 }
 AutoStartPolicy = Load
 AutoStartList = { Server1, Server2, Server3, Server4 }
 FailOverPolicy = Load
 Load = 10
)

group G5 (
 SystemList = { Server1, Server2, Server3, Server4 }
 AutoStartPolicy = Load
 AutoStartList = { Server1, Server2, Server3, Server4 }
 FailOverPolicy = Load
 Load = 50
)

group G6 (
 SystemList = { Server1, Server2, Server3, Server4 }
 AutoStartPolicy = Load
 AutoStartList = { Server1, Server2, Server3, Server4 }
 FailOverPolicy = Load
 Load = 30
)

group G7 (
 SystemList = { Server1, Server2, Server3, Server4 }
 AutoStartPolicy = Load
 AutoStartList = { Server1, Server2, Server3, Server4 }
 FailOverPolicy = Load
 Load = 20
)

group G8 (
 SystemList = { Server1, Server2, Server3, Server4 }
 AutoStartPolicy = Load
 AutoStartList = { Server1, Server2, Server3, Server4 }
 FailOverPolicy = Load
 Load = 40
)
```

### AutoStart operation

In this configuration, assume that groups probe in the same order they are described, G1 through G8. Group G1 chooses the system with the highest AvailableCapacity value. All systems have the same available capacity, so G1 starts on Server1 because this server is lexically first. Groups G2 through G4 follow on Server2 through Server4. With the startup decisions made for the initial four groups, the cluster configuration resembles:

| Server  | AvailableCapacity | Online Groups |
|---------|-------------------|---------------|
| Server1 | 80                | G1            |
| Server2 | 60                | G2            |
| Server3 | 70                | G3            |
| Server4 | 90                | G4            |

As the next groups come online, group G5 starts on Server4 because this server has the highest AvailableCapacity value. Group G6 then starts on Server1 with AvailableCapacity of 80. Group G7 comes online on Server3 with AvailableCapacity of 70 and G8 comes online on Server2 with AvailableCapacity of 60.

The cluster configuration now resembles:

| Server  | AvailableCapacity | Online Groups |
|---------|-------------------|---------------|
| Server1 | 50                | G1 and G6     |
| Server2 | 20                | G2 and G8     |
| Server3 | 50                | G3 and G7     |
| Server4 | 40                | G4 and G5     |

In this configuration, Server2 fires the loadwarning trigger after 600 seconds because it is at the default LoadWarningLevel of 80 percent.

### Failure scenario

In the first failure scenario, Server4 fails. Group G4 chooses Server1 because Server1 and Server3 have AvailableCapacity of 50 and Server1 is lexically first. Group G5 then comes online on Server3. Serializing the failover choice allows complete load-based control and adds less than one second to the total failover time.

Following the first failure, the configuration now resembles:

| Server  | AvailableCapacity | Online Groups  |
|---------|-------------------|----------------|
| Server1 | 40                | G1, G6, and G4 |
| Server2 | 20                | G2 and G8      |
| Server3 | 0                 | G3, G7, and G5 |

In this configuration, Server3 fires the loadwarning trigger to notify that the server is overloaded. An administrator can then switch group G7 to Server1 to balance the load across groups G1 and G3. When Server4 is repaired, it rejoins the cluster with an AvailableCapacity value of 100, making it the most eligible target for a failover group.

### Cascading failure scenario

If Server3 fails before Server4 can be repaired, group G3 chooses Server1, group G5 chooses Server2, and group G7 chooses Server1. This results in the following configuration:

| Server  | AvailableCapacity | Online Groups          |
|---------|-------------------|------------------------|
| Server1 | -10               | G1, G6, G4, G3, and G7 |
| Server2 | -30               | G2, G8, and G5         |

Server1 fires the loadwarning trigger to notify that it is overloaded.

## Sample configuration: Complex four-node cluster

The cluster in this example has two large enterprise servers (LargeServer1 and LargeServer2) and two medium-sized servers (MedServer1 and MedServer2). It has four service groups, G1 through G4, with various loads and prerequisites. Groups G1 and G2 are database applications with specific shared memory and semaphore requirements. Groups G3 and G4 are middle-tier applications with no specific memory or semaphore requirements.

```
include "types.cf"
cluster SGWM-demo (
)

system LargeServer1 (
 Capacity = 200
 Limits = { ShrMemSeg=20, Semaphores=10, Processors=12 }
 LoadWarningLevel = 90
 LoadTimeThreshold = 600
)

system LargeServer2 (
 Capacity = 200
 Limits = { ShrMemSeg=20, Semaphores=10, Processors=12 }
 LoadWarningLevel=70
 LoadTimeThreshold=300
)

system MedServer1 (
 Capacity = 100
 Limits = { ShrMemSeg=10, Semaphores=5, Processors=6 }
)

system MedServer2 (
 Capacity = 100
 Limits = { ShrMemSeg=10, Semaphores=5, Processors=6 }
)
```

```
group G1 (
 SystemList = { LargeServer1, LargeServer2, MedServer1,
MedServer2 }
 SystemZones = { LargeServer1=0, LargeServer2=0, MedServer1=1,
MedServer2=1 }
 AutoStartPolicy = Load
 AutoStartList = { LargeServer1, LargeServer2 }
 FailOverPolicy = Load
 Load = 100
 Prerequisites = { ShrMemSeg=10, Semaphores=5, Processors=6 }
)

group G2 (
 SystemList = { LargeServer1, LargeServer2, MedServer1,
MedServer2 }
 SystemZones = { LargeServer1=0, LargeServer2=0, MedServer1=1,
MedServer2=1 }
 AutoStartPolicy = Load
 AutoStartList = { LargeServer1, LargeServer2 }
 FailOverPolicy = Load
 Load = 100
 Prerequisites = { ShrMemSeg=10, Semaphores=5, Processors=6 }
)

group G3 (
 SystemList = { LargeServer1, LargeServer2, MedServer1,
MedServer2 }
 SystemZones = { LargeServer1=0, LargeServer2=0, MedServer1=1,
MedServer2=1 }
 AutoStartPolicy = Load
 AutoStartList = { MedServer1, MedServer2 }
 FailOverPolicy = Load
 Load = 30
)

group G4 (
 SystemList = { LargeServer1, LargeServer2, MedServer1,
MedServer2 }
 SystemZones = { LargeServer1=0, LargeServer2=0, MedServer1=1,
MedServer2=1 }
 AutoStartPolicy = Load
 AutoStartList = { MedServer1, MedServer2 }
 FailOverPolicy = Load
 Load = 20
)
```

## AutoStart operation

In this configuration, the AutoStart sequence resembles:

- G1—LargeServer1
- G2—LargeServer2
- G3—MedServer1
- G4—MedServer2

All groups begin a probe sequence when the cluster starts. Groups G1 and G2 have an AutoStartList of LargeServer1 and LargeServer2. When these groups probe, they are queued to go online on one of these servers, based on highest AvailableCapacity value. If G1 probes first, it chooses LargeServer1 because LargeServer1 and LargeServer2 both have an AvailableCapacity of 200, but LargeServer1 is lexically first. Groups G3 and G4 use the same algorithm to determine their servers.

## Normal operation

The configuration resembles:

| Server       | AvailableCapacity | CurrentLimits | Online Groups |
|--------------|-------------------|---------------|---------------|
| LargeServer1 | 100               | ShrMemSeg=10  | G1            |
|              |                   | Semaphores=5  |               |
|              |                   | Processors=6  |               |
| LargeServer2 | 100               | ShrMemSeg=10  | G2            |
|              |                   | Semaphores=5  |               |
|              |                   | Processors=6  |               |
| MedServer1   | 70                | ShrMemSeg=10  | G3            |
|              |                   | Semaphores=5  |               |
|              |                   | Processors=6  |               |
| MedServer2   | 80                | ShrMemSeg=10  | G4            |
|              |                   | Semaphores=5  |               |
|              |                   | Processors=6  |               |

## Failure scenario

In this scenario, if LargeServer2 fails, VCS scans all available systems in group G2’s SystemList that are in the same SystemZone and creates a subset of systems that meet the group’s prerequisites. In this case, LargeServer1 meets all



required Limits. Group G2 is brought online on LargeServer1. This results in the following configuration:

| Server       | AvailableCapacity | CurrentLimits | Online Groups |
|--------------|-------------------|---------------|---------------|
| LargeServer1 | 0                 | ShrMemSeg=0   | G1, G2        |
|              |                   | Semaphores=0  |               |
|              |                   | Processors=0  |               |
| MedServer1   | 70                | ShrMemSeg=10  | G3            |
|              |                   | Semaphores=5  |               |
|              |                   | Processors=6  |               |
| MedServer2   | 80                | ShrMemSeg=10  | G4            |
|              |                   | Semaphores=5  |               |
|              |                   | Processors=6  |               |

After 10 minutes (LoadTimeThreshold = 600) VCS fires the loadwarning trigger on LargeServer1 because the LoadWarningLevel exceeds 90 percent.

### Cascading failure scenario

In this scenario, another system failure can be tolerated because each system has sufficient Limits to accommodate the service group running on its peer. If MedServer1 fails, its groups can fail over to MedServer2.

If LargeServer1 fails, the failover of the two groups running on it is serialized. The first group lexically, G1, chooses MedServer2 because the server meets the required Limits and has AvailableCapacity value. Group G2 chooses MedServer1 because it is the only remaining system that meets the required Limits.

## Sample configuration: Server consolidation

The following configuration has a complex eight-node cluster running multiple applications and large databases. The database servers, LargeServer1, LargeServer2, and LargeServer3, are enterprise systems. The middle-tier servers running multiple applications are MedServer1, MedServer2, MedServer3, MedServer4, and MedServer5.

In this configuration, the database zone (system zone 0) can handle a maximum of two failures. Each server has Limits to support a maximum of three database service groups. The application zone has excess capacity built into each server.

The servers running the application groups specify Limits to support one database, even though the application groups do not run prerequisites. This allows a database to fail over across system zones and run on the least-loaded server in the application zone.

```
include "types.cf"
cluster SGWM-demo (
)

system LargeServer1 (
 Capacity = 200
 Limits = { ShrMemSeg=15, Semaphores=30, Processors=18 }
 LoadWarningLevel = 80
 LoadTimeThreshold = 900
)

system LargeServer2 (
 Capacity = 200
 Limits = { ShrMemSeg=15, Semaphores=30, Processors=18 }
 LoadWarningLevel=80
 LoadTimeThreshold=900
)

system LargeServer3 (
 Capacity = 200
 Limits = { ShrMemSeg=15, Semaphores=30, Processors=18 }
 LoadWarningLevel=80
 LoadTimeThreshold=900
)

system MedServer1 (
 Capacity = 100
 Limits = { ShrMemSeg=5, Semaphores=10, Processors=6 }
)
```

```
system MedServer2 (
 Capacity = 100
 Limits = { ShrMemSeg=5, Semaphores=10, Processors=6 }
)

system MedServer3 (
 Capacity = 100
 Limits = { ShrMemSeg=5, Semaphores=10, Processors=6 }
)

system MedServer4 (
 Capacity = 100
 Limits = { ShrMemSeg=5, Semaphores=10, Processors=6 }
)

system MedServer5 (
 Capacity = 100
 Limits = { ShrMemSeg=5, Semaphores=10, Processors=6 }
)

group Database1 (
 SystemList = { LargeServer1, LargeServer2, LargeServer3,
 MedServer1, MedServer2, MedServer3, MedServer4,
 MedServer5 }
 SystemZones = { LargeServer1=0, LargeServer2=0,
 LargeServer3=0,
 MedServer1=1, MedServer2=1, MedServer3=1,
 MedServer4=1,
 MedServer5=1 }
 AutoStartPolicy = Load
 AutoStartList = { LargeServer1, LargeServer2, LargeServer3 }
 FailOverPolicy = Load
 Load = 100
 Prerequisites = { ShrMemSeg=5, Semaphores=10, Processors=6 }
)

group Database2 (
 SystemList = { LargeServer1, LargeServer2, LargeServer3,
 MedServer1, MedServer2, MedServer3, MedServer4,
 MedServer5 }
 SystemZones = { LargeServer1=0, LargeServer2=0,
 LargeServer3=0,
 MedServer1=1, MedServer2=1, MedServer3=1,
 MedServer4=1,
 MedServer5=1 }
 AutoStartPolicy = Load
 AutoStartList = { LargeServer1, LargeServer2, LargeServer3 }
 FailOverPolicy = Load
 Load = 100
 Prerequisites = { ShrMemSeg=5, Semaphores=10, Processors=6 }
)
```

```

group Database3 (
 SystemList = { LargeServer1, LargeServer2, LargeServer3,
 MedServer1, MedServer2, MedServer3, MedServer4,
MedServer5 }
 SystemZones = { LargeServer=0, LargeServer2=0,
LargeServer3=0,
 MedServer1=1, MedServer2=1, MedServer3=1,
MedServer4=1,
 MedServer5=1 }
 AutoStartPolicy = Load
 AutoStartList = { LargeServer1, LargeServer2, LargeServer3 }
 FailOverPolicy = Load
 Load = 100
 Prerequisites = { ShrMemSeg=5, Semaphores=10, Processors=6 }
)

group Application1 (
 SystemList = { LargeServer1, LargeServer2, LargeServer3,
 MedServer1, MedServer2, MedServer3, MedServer4,
MedServer5 }
 SystemZones = { LargeServer1=0, LargeServer2=0,
LargeServer3=0,
 MedServer1=1, MedServer2=1, MedServer3=1,
MedServer4=1,
 MedServer5=1 }
 AutoStartPolicy = Load
 AutoStartList = { MedServer1, MedServer2, MedServer3,
MedServer4,
 MedServer5 }
 FailOverPolicy = Load
 Load = 50
)

group Application2 (
 SystemList = { LargeServer1, LargeServer2, LargeServer3,
 MedServer1, MedServer2, MedServer3, MedServer4,
MedServer5 }
 SystemZones = { LargeServer1=0, LargeServer2=0,
LargeServer3=0,
 MedServer1=1, MedServer2=1, MedServer3=1,
MedServer4=1,
 MedServer5=1 }
 AutoStartPolicy = Load
 AutoStartList = { MedServer1, MedServer2, MedServer3,
MedServer4,
 MedServer5 }
 FailOverPolicy = Load
 Load = 50
)

```

```
group Application3 (
 SystemList = { LargeServer1, LargeServer2, LargeServer3,
 MedServer1, MedServer2, MedServer3, MedServer4,
MedServer5 }
 SystemZones = { LargeServer1=0, LargeServer2=0,
LargeServer3=0,
 MedServer1=1, MedServer2=1, MedServer3=1,
MedServer4=1,
 MedServer5=1 }
 AutoStartPolicy = Load
 AutoStartList = { MedServer1, MedServer2, MedServer3,
MedServer4,
 MedServer5 }
 FailOverPolicy = Load
 Load = 50
)

group Application4 (
 SystemList = { LargeServer1, LargeServer2, LargeServer3,
 MedServer1, MedServer2, MedServer3, MedServer4,
MedServer5 }
 SystemZones = { LargeServer1=0, LargeServer2=0,
LargeServer3=0,
 MedServer1=1, MedServer2=1, MedServer3=1,
MedServer4=1,
 MedServer5=1 }
 AutoStartPolicy = Load
 AutoStartList = { MedServer1, MedServer2, MedServer3,
MedServer4,
 MedServer5 }
 FailOverPolicy = Load
 Load = 50
)

group Application5 (
 SystemList = { LargeServer1, LargeServer2, LargeServer3,
 MedServer1, MedServer2, MedServer3, MedServer4,
MedServer5 }
 SystemZones = { LargeServer1=0, LargeServer2=0,
LargeServer3=0,
 MedServer1=1, MedServer2=1, MedServer3=1,
MedServer4=1,
 MedServer5=1 }
 AutoStartPolicy = Load
 AutoStartList = { MedServer1, MedServer2, MedServer3,
MedServer4,
 MedServer5 }
 FailOverPolicy = Load
 Load = 50
)
```

## AutoStart operation

Based on the preceding main.cf example, the AutoStart sequence resembles:

|              |              |
|--------------|--------------|
| Database1    | LargeServer1 |
| Database2    | LargeServer2 |
| Database3    | LargeServer3 |
| Application1 | MedServer1   |
| Application2 | MedServer2   |
| Application3 | MedServer3   |
| Application4 | MedServer4   |
| Application5 | MedServer5   |

## Normal operation

The configuration resembles:

| Server       | AvailableCapacity | CurrentLimits                                  | Online Groups |
|--------------|-------------------|------------------------------------------------|---------------|
| LargeServer1 | 100               | ShrMemSeg=10<br>Semaphores=20<br>Processors=12 | Database1     |
| LargeServer2 | 100               | ShrMemSeg=10<br>Semaphores=20<br>Processors=12 | Database2     |
| LargeServer3 | 100               | ShrMemSeg=10<br>Semaphores=20<br>Processors=12 | Database3     |
| MedServer1   | 50                | ShrMemSeg=5<br>Semaphores=10<br>Processors=6   | Application1  |
| MedServer2   | 50                | ShrMemSeg=5<br>Semaphores=10<br>Processors=6   | Application2  |
| MedServer3   | 50                | ShrMemSeg=5<br>Semaphores=10<br>Processors=6   | Application3  |

| Server     | AvailableCapacity | CurrentLimits                                | Online Groups |
|------------|-------------------|----------------------------------------------|---------------|
| MedServer4 | 50                | ShrMemSeg=5<br>Semaphores=10<br>Processors=6 | Application4  |
| MedServer5 | 50                | ShrMemSeg=5<br>Semaphores=10<br>Processors=6 | Application5  |

Failure scenario

In the following example, LargeServer3 fails. VCS scans all available systems in the SystemList for the Database3 group for systems in the same SystemZone and identifies systems that meet the group’s prerequisites. In this case, LargeServer1 and LargeServer2 meet the required Limits. Database3 is brought online on LargeServer1. This results in the following configuration:

| Server       | AvailableCapacity | CurrentLimits                                  | Online Groups          |
|--------------|-------------------|------------------------------------------------|------------------------|
| LargeServer1 | 0                 | ShrMemSeg=5<br>Semaphores=10<br>Processors=6   | Database1<br>Database3 |
| LargeServer2 | 100               | ShrMemSeg=10<br>Semaphores=20<br>Processors=12 | Database2              |

In this scenario, further failure of either system can be tolerated because each has sufficient Limits available to accommodate the additional service group.

Cascading failure scenario

If the performance of a database is unacceptable with two database groups running on a single server, the SystemZones policy can help expedite performance. Failing over a database group into the application zone has the effect of resetting the group’s preferred zone. For example, in the above scenario Database3 was moved to LargeServer1. The administrator could reconfigure the application zone to move two application groups to a single system. The database application can then be switched to the empty application server (MedServer1–MedServer5), which would put Database3 in Zone1 (application zone). If a failure occurs in Database3, the group selects the least-loaded server in the application zone for failover.





# The role of service group dependencies

- [About service group dependencies](#)
- [Service group dependency configurations](#)
- [Group Dependency FAQs](#)
- [Linking service groups](#)
- [VCS behavior with service group dependencies](#)

## About service group dependencies

Service groups can be dependent on each other. The dependent group is the *parent* and the other group is the *child*. For example a finance application (parent) may require that the database application (child) is online before it comes online. While service group dependencies offer more features to manage application service groups, they create more complex failover configurations.

A service group may function both as a parent and a child. Veritas Cluster Server supports five levels of service group dependencies.

## Dependency links

The dependency relationship between a parent and a child is called a *link*. The link is characterized by the dependency category, the location of the service groups, and the rigidity of dependency.

- A dependency may be *online*, or *offline*.
- A dependency may be *local*, *global*, or *remote*.
- A dependency may be *soft*, *firm*, or *hard* with respect to the rigidity of the constraints between parent and child service group.

You can customize the behavior of service groups by choosing the right combination of the dependency category, location, and rigidity.

### Dependency categories: online or offline dependencies

Dependency categories determine the relationship of the parent group with the state of the child group.

|                          |                                                                                                                                                                                                                                                                                                                        |
|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Online group dependency  | <p>The parent group must wait for the child group to be brought online before it can start.</p> <p>For example, to configure a database application and a database service as two separate groups, specify the database application as the parent, and the database service as the child.</p>                          |
| Offline group dependency | <p>The parent group can be started only if the child group is offline and vice versa. This behavior prevents conflicting applications from running on the same system.</p> <p>For example, configure a test application on one system as the parent and the production application on another system as the child.</p> |

### Dependency location: local, global, or remote dependencies

The relative location of the parent and child service groups determines whether the dependency between them is a local, global, or remote.

|                   |                                                                                                                                                                 |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Local dependency  | The parent group depends on the child group being online or offline on the same system.                                                                         |
| Global dependency | An instance of the parent group depends on one or more instances of the child group being online on any system.                                                 |
| Remote dependency | An instance of parent group depends on one or more instances of the child group being online on any system other than the system on which the parent is online. |

## Dependency rigidity: soft, firm, or hard dependencies

The type of dependency defines the rigidity of the link between parent and child groups. A soft dependency means minimum constraints, whereas a hard dependency means maximum constraints

**Soft dependency** Specifies the minimum constraints while bringing parent and child groups online. The only constraint is that the child group must be online before the parent group is brought online.

Note that VCS may not honor soft dependencies when either the `hastop -local` or the `/etc/init.d/vcs stop` are run. In this scenario, VCS takes both parent and child service groups offline in any order, without considering the group dependency.

For example, in an online local soft dependency, an instance of the child group must be online on the same system before the parent group can come online.

Soft dependency provides the following flexibility:

- If the child group faults, VCS does not immediately take the parent offline. If the child group cannot fail over, the parent remains online.
- When both groups are online, either group, child or parent, may be taken offline while the other remains online.
- If the parent group faults, the child group may remain online.
- When the link is created, the child group need not be online if the parent is online. However, when both groups are online, their online state must not conflict with the type of link.

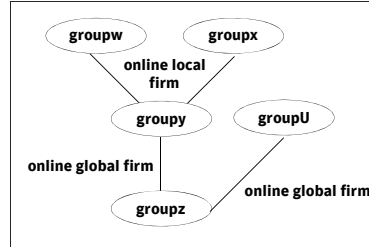
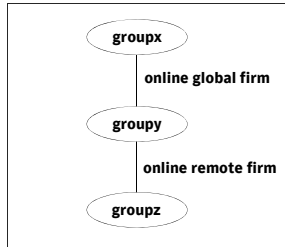
**Firm dependency** Imposes more constraints when VCS brings the parent or child groups online or takes them offline. In addition to the constraint that the child group must be online before the parent group is brought online, the constraints include:

- If the child group faults, the parent is taken offline. If the parent is frozen at the time of the fault, the parent remains in its original state. If the child cannot fail over to another system, the parent remains offline.
- If the parent group faults, the child group may remain online.
- The child group cannot be taken offline if the parent group is online. The parent group can be taken offline while the child is online.
- When the link is created, the parent group must be offline. However, if both groups are online, their online state must not conflict with the type of link.

|                 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Hard dependency | <p data-bbox="685 223 1330 286">Imposes the maximum constraints when VCS brings the parent of child service groups online or takes them offline. For example:</p> <ul data-bbox="685 286 1330 529" style="list-style-type: none"> <li data-bbox="685 286 1330 442">■ If a child group faults, the parent is taken offline before the child group is taken offline. If the child group fails over, the parent fails over to another system (or the same system for a local dependency). If the child group cannot fail over, the parent group remains offline.</li> <li data-bbox="685 442 1330 529">■ If the parent faults, the child is taken offline. If the child fails over, the parent fails over. If the child group cannot fail over, the parent group remains offline.</li> </ul> <p data-bbox="685 538 1330 598"><b>Note:</b> When the child faults, if the parent group is frozen, the parent remains online. The faulted child does not fail over.</p> <p data-bbox="685 607 1330 668">The following restrictions apply when configuring a hard dependency:</p> <ul data-bbox="685 668 1330 982" style="list-style-type: none"> <li data-bbox="685 668 1330 703">■ Only online local hard dependencies are supported.</li> <li data-bbox="685 703 1330 763">■ Only a single-level, parent-child relationship can be configured as a hard dependency.</li> <li data-bbox="685 763 1330 824">■ Only one parent and one child group can be configured in a hard dependency.</li> <li data-bbox="685 824 1330 885">■ Bringing the child group online does not automatically bring the parent online.</li> <li data-bbox="685 885 1330 946">■ Taking the parent group offline does not automatically take the child offline.</li> <li data-bbox="685 946 1330 982">■ Bringing the parent online is prohibited if the child is offline.</li> </ul> |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

## Dependency limitations

- Multiple parent service groups may depend on a child service group, although a parent group may depend on only one child group.
- A group dependency tree may be at most five levels deep.



- You cannot link two service groups whose current states violate the relationship.  
For example, all link requests are accepted if all instances of parent group are offline.  
All link requests are rejected if parent group is online and child group is offline, except in offline dependencies.  
All online global/online remote link requests to link two parallel groups are rejected.  
All online local link requests to link a parallel parent group to a failover child group are rejected.

# Service group dependency configurations

In the following tables, the term instance applies to parallel groups only. If a parallel group is online on three systems, for example, an instance of the group is online on each system. For failover groups, only one instance of a group is online at any time. The default dependency type is Firm.

## Failover parent / Failover child

| Link              | Failover Parent Depends on ...        | Failover Parent is Online If ... | If Failover Child Faults, then ...                                                                                                                                                              | If Failover Parent Faults, then ...                                                                                                            |
|-------------------|---------------------------------------|----------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| online local soft | Failover Child online on same system. | Child is online on same system.  | Parent stays online.<br><br>If Child fails over to another system, Parent migrates to the same system.<br><br>If Child cannot fail over, Parent remains online.                                 | Child stays online.                                                                                                                            |
| online local firm | Failover Child online on same system. | Child is online on same system.  | Parent taken offline.<br><br>If Child fails over to another system, Parent migrates to the same system.<br><br>If Child cannot fail over, Parent remains offline.                               | Child stays online.                                                                                                                            |
| online local hard | Failover Child online on same system. | Child is online on same system.  | Parents taken offline before Child is taken offline.<br><br>If Child fails over to another system, Parent migrates to another system.<br><br>If Child cannot fail Over, Parent remains offline. | Child taken offline.<br><br>If Child fails over, Parent migrates to the same system.<br><br>If Child cannot fail over, Parent remains offline. |

| Link               | Failover Parent Depends on ...                  | Failover Parent is Online If ...          | If Failover Child Faults, then ...                                                                                                                                                                                                                  | If Failover Parent Faults, then ...                                                                                                                                                                                                     |
|--------------------|-------------------------------------------------|-------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| online global soft | Failover Child online somewhere in the cluster. | Child is online somewhere in the cluster. | Parent stays online.<br><br>If Child fails over to another system, Parent remains online.<br><br>If Child cannot fail over, Parent remains online.                                                                                                  | Child stays online.<br><br>Parent fails over to any available system.<br><br>If no system is available, Parent remains offline.                                                                                                         |
| online global firm | Failover Child online somewhere in the cluster. | Child is online somewhere in the cluster. | Parent taken offline after Child is offlined.<br><br>If Child fails over to another system, Parent is brought online on any system.<br><br>If Child cannot fail over, Parent remains offline.                                                       | Child stays online.<br><br>Parent fails over to any available system.<br><br>If no system is available, Parent remains offline.                                                                                                         |
| online remote soft | Failover Child online on another system.        | Child is online on another system.        | If Child fails over to the system on which Parent was online, Parent migrates to another system.<br><br>If Child fails over to another system, Parent continues to run on original system.<br><br>If Child cannot fail over, Parent remains online. | Child stays online.<br><br>Parent fails over to a system where Child is not online.<br><br>If the only system available is where Child is online, Parent is not brought online.<br><br>If no system is available, Child remains online. |



| Link               | Failover Parent Depends on ...            | Failover Parent is Online If ...     | If Failover Child Faults, then ...                                                                                                                                                                                                                                                                         | If Failover Parent Faults, then ...                                                                                                                                                                                 |
|--------------------|-------------------------------------------|--------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| online remote firm | Failover Child online on another system.  | Child is online on another system.   | <p>If Child fails over to the system on which Parent was online, Parent switches to another system.</p> <p>If Child fails over to another system, Parent restarts on original system.</p> <p>If Child cannot fail over, VCS takes the parent offline.</p>                                                  | <p>Parent fails over to a system where Child is not online.</p> <p>If the only system available is where Child is online, Parent is not brought online.</p> <p>If no system is available, Child remains online.</p> |
| offline local      | Failover Child offline on the same system | Child is offline on the same system. | <p>If Child fails over to the system on which parent in not running, parent continues running.</p> <p>If child fails over to system on which parent is running, parent switches to another system, if available.</p> <p>If no system is available for Child to fail over to, Parent continues running.</p> | <p>Parent fails over to system on which Child is not online.</p> <p>If no system is available, Child remains online</p>                                                                                             |

### Failover parent / Parallel child

With a failover parent and parallel child, no hard dependencies are supported.

| Link               | Failover Parent Depends on ...                          | Failover Parent is Online if ...                                         | If Parallel Child Faults on a system, then ...                                                                                                                                                                                 | If Failover Parent Faults, then ...                                                                                                   |
|--------------------|---------------------------------------------------------|--------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| online local soft  | Instance of parallel Child group on same system.        | Instance of Child is online on same system.                              | Parent fails over to other system and depends on Child instance there.                                                                                                                                                         | Parent fails over to other system and depends on Child instance there.<br><br>Child Instance remains online where the Parent faulted. |
| online local firm  | Instance of parallel Child group on same system.        | Instance of Child is online on same system.                              | Parent is taken offline. Parent fails over to other system and depends on Child instance there.                                                                                                                                | Parent fails over to other system and depends on Child instance there.<br><br>Child Instance remains online where Parent faulted.     |
| online global soft | All instances of parallel Child group remaining online. | One or more instances of Child group is online somewhere in the cluster. | Parent remains online if Child faults on any system.<br><br>If faulted Child fails over to another system, Parent is brought online on any system.<br><br>If Child cannot fail over to another system, Parent remains offline. | Parent fails over to another system, maintaining dependence on all Child instances.                                                   |

| Link               | Failover Parent Depends on ...                                                | Failover Parent is Online if ...                                  | If Parallel Child Faults on a system, then ...                                                                                                                                                                                   | If Failover Parent Faults, then ...                                                 |
|--------------------|-------------------------------------------------------------------------------|-------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| online global firm | All instances of parallel Child group remaining online.                       | All instances of Child group are online somewhere in the cluster. | Parent is taken offline. After Child fails over, Parent fails over to another system.<br><br>If Child cannot fail over, Parent remains offline.                                                                                  | Parent fails over to another system, maintaining dependence on all Child instances. |
| online remote soft | One or more instances parallel Child group remaining online on other systems. | One or more instances of Child group are online on other systems. | Parent remains online.<br><br>If Child fails over to the system on which Parent is online, Parent fails over to another system.                                                                                                  | Parent fails over to another system, maintaining dependence on the Child instances. |
| online remote firm | All instances parallel Child group remaining online on other systems.         | All instances of Child group are online on other systems.         | Parent is taken offline.<br><br>If Child fails over to the system on which Parent is online, Parent fails over to another system.<br><br>If Child fails over to another system, Parent is brought online on its original system. | Parent fails over to another system, maintaining dependence on all Child instances. |
| offline local      | Parallel Child offline on same system.                                        | No instance of Child is online on same system.                    | Parent remains online if Child fails over to another system.<br><br>If Child fails over to the system on which Parent is online, Parent fails over.                                                                              | Child remains online.                                                               |

Parallel parent / Failover child

| Link               | Parallel Parent Instances Depend on ...               | Parallel Parent Instances are Online if ...        | If Failover Child Faults on a system, then ...                                                                                                                                                                           | If Parallel Parent Faults, then ...                                                          |
|--------------------|-------------------------------------------------------|----------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------|
| online global soft | Failover Child group online somewhere in the cluster. | Failover Child is online somewhere in the cluster. | Parent remains online.                                                                                                                                                                                                   | Child remains online                                                                         |
| online global firm | Failover Child group somewhere in the cluster.        | Failover Child is online somewhere in the cluster. | All instances of Parent taken offline.<br>After Child fails over, Parent instances are brought failed over or restarted on the same systems.                                                                             | Child stays online.                                                                          |
| online remote soft | Failover Child group on another system.               | Failover Child is online on another system.        | If Child fails over to system on which Parent is online, Parent fails over to other systems.<br>If Child fails over to another system, Parent remains online.                                                            | Child remains online. Parent tries to fail over to another system where child is not online. |
| online remote firm | Failover Child group on another system.               | Failover Child is online on another system.        | All instances of Parent taken offline.<br>If Child fails over to system on which Parent was online, Parent fails over to other systems.<br>If Child fails over to another system, Parent brought online on same systems. | Child remains online. Parent tries to fail over to another system where child is not online. |

| Link          | Parallel Parent Instances Depend on ... | Parallel Parent Instances are Online if ...  | If Failover Child Faults on a system, then ...               | If Parallel Parent Faults, then ... |
|---------------|-----------------------------------------|----------------------------------------------|--------------------------------------------------------------|-------------------------------------|
| offline local | Failover Child offline on same system.  | Failover Child is not online on same system. | Parent remains online if Child fails over to another system. | Child remains online.               |

## Parallel parent / Parallel child

Global dependencies between parallel parent groups and parallel child groups are not supported.

| Link              | Parallel Parent Depends on ...                 | Parallel Parent is Online If ...                  | If Parallel Child Faults, then ...                                                                                                                                                                            | If Parallel Parent Faults, then ...                                                                                                                                  |
|-------------------|------------------------------------------------|---------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| online local soft | Parallel Child instance online on same system. | Parallel Child instance is online on same system. | <p>If Child fails over to another system, Parent migrates to the same system as the Child.</p> <p>If Child cannot fail over, Parent remains online.</p>                                                       | <p>Child instance stays online.</p> <p>Parent instance can fail over only to system where Child instance is running and other instance of Parent is not running.</p> |
| online local firm | Parallel Child instance online on same system. | Parallel Child instance is online on same system. | <p>Parent taken offline.</p> <p>If Child fails over to another system, VCS brings an instance of the Parent online on the same system as Child.</p> <p>If Child cannot fail over, Parent remains offline.</p> | <p>Child stays online.</p> <p>Parent instance can fail over only to system where Child instance is running and other instance of Parent is not running.</p>          |

| Link          | Parallel Parent Depends on ...         | Parallel Parent is Online If ...               | If Parallel Child Faults, then ...                           | If Parallel Parent Faults, then ... |
|---------------|----------------------------------------|------------------------------------------------|--------------------------------------------------------------|-------------------------------------|
| offline local | Parallel Child offline on same system. | No instance of Child is online on same system. | Parent remains online if Child fails over to another system. | Child remains online.               |

## Group Dependency FAQs

This section lists some commonly asked questions about group dependencies.

### Dependency Frequently asked questions

|               |                                                                                                                                                                                                                                                           |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Online local  | <b>Can child group be taken offline when parent group is online?</b><br>Soft=Yes Firm=No Hard = No.                                                                                                                                                       |
|               | <b>Can parent group be switched while child group is online?</b><br>Soft=No Firm=No Hard = No.                                                                                                                                                            |
|               | <b>Can child group be switched while parent group is online?</b><br>Soft=No Firm=No Hard = Yes. (Provided there are no other parents online for child group)                                                                                              |
| Online global | <b>Can child group be taken offline when parent group is online?</b><br>Soft=Yes Firm=No.                                                                                                                                                                 |
|               | <b>Can parent group be switched while child group is running?</b><br>Soft=Yes Firm=Yes Hard=Yes.                                                                                                                                                          |
|               | <b>Can child group be switched while parent group is running?</b><br>Soft=Yes Firm=No                                                                                                                                                                     |
| Online remote | <b>Can child group be taken offline when parent group is online?</b><br>Firm=No Soft=Yes.                                                                                                                                                                 |
|               | <b>Can parent group be switched while child group is running?</b><br>Firm=Yes, but not to system on which child is running.<br>Soft=Yes, but not to system on which child is running.                                                                     |
|               | <b>Can child group be switched while parent group is running?</b><br>Firm=No Soft=Yes, but not to system on which parent is running.                                                                                                                      |
| Offline local | <b>Can parent group be brought online when child group is offline?</b><br>Yes.                                                                                                                                                                            |
|               | <b>Can child group be taken offline when parent group is online?</b><br>Yes.                                                                                                                                                                              |
|               | <b>Can parent group be switched while the child group is running?</b><br>Yes, but not to system on which child is running.<br><b>Can child group be switched while the parent group is running?</b><br>Yes, but not to system on which parent is running. |

## Linking service groups

Note that a configuration may require that a certain service group be running before another service group can be brought online. For example, a group containing resources of a database service must be running before the database application is brought online.

See also “[Linking service groups](#)” on page 136

### To link service groups from the command line

- ◆ Type the following command

```
hagrp -link parent_group child_group gd_category
 gd_location gd_type
```

*parent\_group*    Name of the parent group

*child\_group*     Name of the child group

*gd\_category*     category of group dependency (online/offline).

*gd\_location*     the scope of dependency (local/global/remote).

*gd\_type*          type of group dependency (soft/firm/hard). Default is firm.

## VCS behavior with service group dependencies

VCS enables or restricts service group operations to honor service group dependencies. VCS rejects operations if the operation violates a group dependency.

### Online operations in group dependencies

Typically, bringing a child group online manually is never rejected, except under the following circumstances:

- For online local dependencies, if parent is online, a child group online is rejected for any system other than the system where parent is online.
- For online remote dependencies, if parent is online, a child group online is rejected for the system where parent is online.
- For offline local dependencies, if parent is online, a child group online is rejected for the system where parent is online.

The following examples describe situations where bringing a parallel child group online is accepted:



- For a parallel child group linked online local with failover/parallel parent, multiple instances of child group online are acceptable.
- For a parallel child group linked online remote with failover parent, multiple instances of child group online are acceptable, as long as child group does not go online on the system where parent is online.
- For a parallel child group linked offline local with failover/parallel parent, multiple instances of child group online are acceptable, as long as child group does not go online on the system where parent is online.

## Offline operations in group dependencies

VCS rejects offline operations if the procedure violates existing group dependencies. Typically, firm dependencies are more restrictive to taking child group offline while parent group is online. Rules for manual offline include:

- Parent group offline is never rejected.
- For all soft dependencies, child group can go offline regardless of the state of parent group.
- For all firm dependencies, if parent group is online, child group offline is rejected.
- For the online local hard dependency, if parent group is online, child group offline is rejected.

## Switch operations in group dependencies

Switching a service group implies manually taking a service group offline on one system, and manually bringing it back online on another system. VCS rejects manual switch if the group does not comply with the rules for offline or online operations.



# Administration-Beyond the basics

- [Chapter 13, “VCS event notification” on page 429](#)
- [Chapter 14, “VCS event triggers” on page 445](#)



# VCS event notification

- [About VCS event notification](#)
- [Components of VCS event notification](#)
- [VCS events and traps](#)
- [Monitoring aggregate events](#)
- [Configuring notification](#)

## About VCS event notification

VCS provides a method for notifying important events such as resource or system faults to administrators or designated recipients. VCS includes a notifier component, which consists of the notifier process and the hanotify utility.

VCS support SNMP consoles that can use an SNMP V2 MIB.

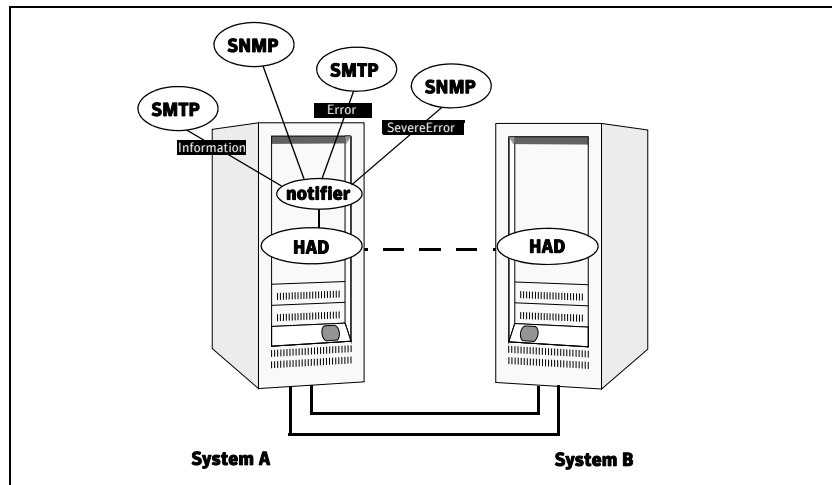
The notifier process performs the following tasks:

- Receives notifications from HAD
- Formats the notification
- Generates an SNMP (V2) trap or sends an email to the designated recipient, or does both.

If you have configured owners for resources, groups, or for the cluster, VCS also notifies owners of events that affect their resources. A resource owner is notified of resource-related events, a group owner of group-related events, and so on.

See “[About attributes and their definitions](#)” on page 644.

There are four severity levels: SevereError, Error, Warning, and Information. SevereError indicates the highest severity level, Information the lowest. Note that these severity levels are case-sensitive.



SNMP traps are forwarded to the SNMP console. Typically, traps are predefined for events such as service group or resource faults. You can use the hanotify utility to send additional traps.

## Event messages and severity levels

When the VCS engine starts up, it queues all messages as Information. However, when notifier connects, it communicates one of the following severity levels to HAD, depending on which is the lowest:

- lowest severity for SNMP options
- lowest severity for SMTP options

If notifier is started from the command line without specifying a severity level for the SNMP console or SMTP recipients, notifier communicates the default severity level Warning to HAD. If notifier is configured under VCS control, severity must be specified. See the description of the NotifierMngr agent in the *Veritas Cluster Server Bundled Agents Reference Guide*.

For example, if the following severities are specified for notifier:

- Warning for email recipient 1
- Error for email recipient 2
- SevereError for SNMP console

Notifier communicates the minimum severity, Warning, to HAD, which then queues all messages labelled severity level Warning and greater.

Notifier ensures recipients gets only the messages they are designated to receive (according to the specified severity level). However, until notifier communicates the specifications to HAD, HAD stores all messages, because it does not know the severity the user has specified. This behavior prevents messages from being lost between the time HAD stores them and notifier communicates the specifications to HAD.

## Persistent and replicated message queue

VCS includes a sophisticated mechanism for maintaining event messages, which ensures that messages are not lost. On each node, VCS queues messages to be sent to the notifier process. This queue is persistent as long as VCS is running and the contents of this queue remain the same on each node. If the notifier service group fails, notifier is failed over to another node in the cluster. Because the message queue is consistent across nodes, notifier can resume message delivery from where it left off even after failover.

## How HAD deletes messages

The VCS engine, HAD, stores messages to be sent to notifier. HAD deletes messages under the following conditions:

- The message has been in the queue for one hour and notifier is unable to deliver the message to the recipient. (This behavior means that until notifier connects to HAD, messages are stored permanently in the queue until one of the following conditions are met.)  
*or*
- The message queue is full and to make room for the latest message, the earliest message is deleted.  
*or*
- VCS receives a message acknowledgement from notifier when notifier has delivered the message to at least one designated recipient.  
Example: two SNMP consoles and two email recipients are designated. Notifier sends an acknowledgement to VCS, even if the message reached only one of the four recipients. Error messages are also printed to the log files when delivery errors occur.

## Components of VCS event notification

This section describes the notifier process and the hanotify utility.

### The notifier process

The notifier process configures how messages are received from VCS and how they are delivered to SNMP consoles and SMTP servers. Using notifier, you can specify notification based on the severity level of the events generating the messages. You can also specify the size of the VCS message queue, which is 30 by default. You can change this value by modifying the MessageQueue attribute. See the *Veritas Cluster Server Bundled Agents Reference Guide* for more information about this attribute.

When notifier is started from the command line, VCS does not control the notifier process. For best results, use the NotifierMngr agent that is bundled with VCS. Configure notifier as part of a highly available service group, which can then be monitored, brought online, and taken offline. For information about the agent, see the *Veritas Cluster Server Bundled Agents Reference Guide*.

Note that notifier must be configured in a failover group, not parallel, because only one instance of notifier runs in the entire cluster. Also note that notifier does not respond to SNMP `get` or `set` requests; notifier is a trap generator only.



Notifier enables you to specify configurations for the SNMP manager and SMTP server, including machine names, ports, community IDs, and recipients' email addresses. You can specify more than one manager or server, and the severity level of messages that are sent to each.

---

**Note:** If you start the notifier outside of VCS control, use the absolute path of the notifier in the command. VCS cannot monitor the notifier process if it is started outside of VCS control using a relative path.

---

## Example of notifier command

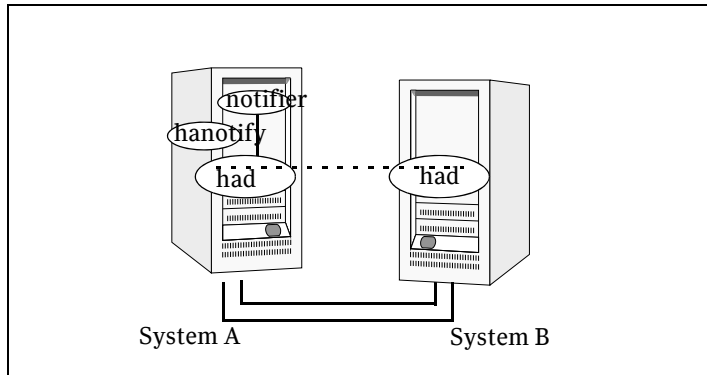
```
/opt/VRTSvcs/bin/notifier -s m=north -s
m=south,p=2000,l=Error,c=your_company
-t m=north,e="abc@your_company.com",l=SevereError
```

In this example, notifier:

- Sends all level SNMP traps to *north* at the default SNMP port and community value *public*.
- Sends Error and SevereError traps to *south* at *port 2000* and community value *your\_company*.
- Sends SevereError email messages to *north* as SMTP server at default port and to email recipient *abc@your\_company.com*.

## The hanotify utility

The hanotify utility enables you to construct user-defined messages. The utility forwards messages to HAD, which stores them in its internal message queue. Along with other messages, user-defined messages are also forwarded to the notifier process for delivery to email recipients, SNMP consoles, or both.



### Example of hanotify command

```
hanotify -i 1.3.6.1.4.1.1302.3.8.10.2.8.0.10 -l Warning -n
agentres -T 7 -t "custom agent" -o 4 -S sys1 -L mv -p
sys2 -P mv -c MyAgent -C 7 -O johndoe -m "Custom message"
```

In this example, the number 1.3.6.1.4.1.1302.3.8.10.2.8.0.10 is the OID for the message being sent. Because it is a user-defined message, VCS has no way of knowing the OID associated with the SNMP trap corresponding to this message. Users must provide the OID.

The message severity level is set to Warning. The affected systems are sys1 and sys2. Running this command sends a custom message for the resource agentres from the agent MyAgent.

## VCS events and traps

This section lists the events generate traps, email notification, or both. Note that SevereError indicates the highest severity level, Information the lowest. Traps specific to global clusters are ranked from Critical, the highest severity, to Normal, the lowest.

### Events and traps for clusters

| Event                                                          | Severity Level | Description                                                                                                             |
|----------------------------------------------------------------|----------------|-------------------------------------------------------------------------------------------------------------------------|
| Cluster has faulted.                                           | Error          | Self-explanatory.                                                                                                       |
| Heartbeat is down.<br>(Global Cluster Option)                  | Error          | The connector on the local cluster lost its heartbeat connection to the remote cluster.                                 |
| Remote cluster is in RUNNING state.<br>(Global Cluster Option) | Information    | Local cluster has complete snapshot of the remote cluster, indicating the remote cluster is in the RUNNING state.       |
| Heartbeat is “alive.”<br>(Global Cluster Option)               | Information    | Self-explanatory.                                                                                                       |
| User has logged on to VCS.                                     | Information    | A user log on has been recognized because a user logged on by Cluster Manager, or because a ha.xxx command was invoked. |

### Events and traps for agents

| Event               | Severity Level | Description                                       |
|---------------------|----------------|---------------------------------------------------|
| Agent is faulted.   | Warning        | The agent has faulted on one node in the cluster. |
| Agent is restarting | Information    | VCS is restarting the agent.                      |

## Events and traps for resources

| Event                                    | Severity Level                | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------------------------|-------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Resource state is unknown.               | Warning                       | VCS cannot identify the state of the resource.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Resource monitoring has timed out.       | Warning                       | Monitoring mechanism for the resource has timed out.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Resource is not going offline.           | Warning                       | VCS cannot take the resource offline.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| Health of cluster resource declined.     | Warning                       | Used by agents to give additional information on the state of a resource. Health of the resource declined while it was online.                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Resource went online by itself.          | Warning (not for first probe) | The resource was brought online on its own.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| Resource has faulted.                    | Error                         | Self-explanatory.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Resource is being restarted by agent.    | Information                   | The agent is restarting the resource.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| The health of cluster resource improved. | Information                   | Used by agents to give extra information about state of resource. Health of the resource improved while it was online.                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| Resource monitor time has changed.       | Warning                       | <p>This trap is generated when statistical analysis for the time taken by the monitor function of an agent is enabled for the agent.</p> <p>See “<a href="#">VCS agent statistics</a>” on page 579.</p> <p>This trap is generated when the agent framework detects a sudden change in the time taken to run the monitor function for a resource. The trap information contains details of:</p> <ul style="list-style-type: none"><li>■ The change in time required to run the monitor function</li><li>■ The actual times that were compared to deduce this change.</li></ul> |
| Resource is in ADMIN_WAIT state.         | Error                         | <p>The resource is in the admin_wait state.</p> <p>See “<a href="#">Controlling Clean behavior on resource faults</a>” on page 368.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                       |

## Events and traps for systems

| Event                                       | Severity Level | Description                                                                                                                                                                                                                         |
|---------------------------------------------|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| VCS is being restarted by hashadow.         | Warning        | Self-explanatory.                                                                                                                                                                                                                   |
| VCS is in jeopardy.                         | Warning        | One node running VCS is in jeopardy.                                                                                                                                                                                                |
| VCS is up on the first node in the cluster. | Information    | Self-explanatory.                                                                                                                                                                                                                   |
| VCS has faulted.                            | SevereError    | Self-explanatory.                                                                                                                                                                                                                   |
| A node running VCS has joined cluster.      | Information    | Self-explanatory.                                                                                                                                                                                                                   |
| VCS has exited manually.                    | Information    | VCS has exited gracefully from one node on which it was previously running.                                                                                                                                                         |
| CPU usage exceeded threshold on the system. | Warning        | <div>The system's CPU usage continuously exceeded the value that is set in the Notify threshold for a duration greater than the Notify time limit.</div> <div>See <a href="#">“When a resource comes online”</a> on page 568.</div> |

## Events and traps for service groups

| Event                                                                                       | Severity Level | Description                                                                                                                            |
|---------------------------------------------------------------------------------------------|----------------|----------------------------------------------------------------------------------------------------------------------------------------|
| Service group has faulted.                                                                  | Error          | Self-explanatory.                                                                                                                      |
| Service group concurrency violation.                                                        | SevereError    | A failover service group has become online on more than one node in the cluster.                                                       |
| Service group has faulted and cannot be failed over anywhere.                               | SevereError    | Specified service group faulted on all nodes where group could be brought online. There are no nodes to which the group can fail over. |
| Service group is online                                                                     | Information    | Self-explanatory.                                                                                                                      |
| Service group is offline.                                                                   | Information    | Self-explanatory.                                                                                                                      |
| Service group is autodisabled.                                                              | Information    | VCS has autodisabled the specified group because one node exited the cluster.                                                          |
| Service group is restarting.                                                                | Information    | Self-explanatory.                                                                                                                      |
| Service group is being switched.                                                            | Information    | VCS is taking the service group offline on one node and bringing it online on another.                                                 |
| Service group restarting in response to persistent resource going online.                   | Information    | Self-explanatory.                                                                                                                      |
| The global service group is online/partial on multiple clusters.<br>(Global Cluster Option) | SevereError    | A concurrency violation occurred for the global service group.                                                                         |
| Attributes for global service groups are mismatched.<br>(Global Cluster Option)             | Error          | The attributes ClusterList, AutoFailOver, and Parallel are mismatched for the same global service group on different clusters.         |

## SNMP-specific files

VCS includes two SNMP-specific files: `vcs.mib` and `vcs_trapd`, which are created in `/etc/VRTSvcs/snmp`. The file `vcs.mib` is the textual MIB for built-in traps that are supported by VCS. Load this MIB into your SNMP console to add it to the list of recognized traps.

The file `vcs_trapd` is specific to the HP OpenView Network Node Manager (NNM) SNMP console. The file includes sample events configured for the built-in SNMP traps supported by VCS. To merge these events with those configured for SNMP traps:

```
xnmevents -merge vcs_trapd
```

When you merge events, the SNMP traps sent by VCS by way of notifier are displayed in the HP OpenView NNM SNMP console.

---

**Note:** For more information on `xnmevents`, see the HP OpenView documentation.

---

## Trap variables in VCS MIB

Traps sent by VCS are reversible to SNMPv2 after an SNMPv2 -> SNMPv1 conversion.

For reversible translations between SNMPv1 and SNMPv2 trap PDUs, the second-last ID of the SNMP trap OID must be zero. This ensures that once you make a *forward* translation (SNMPv2 trap -> SNMPv1; RFC 2576 Section 3.2), the *reverse* translation (SNMPv1 trap --> SNMPv2 trap; RFC 2576 Section 3.1) is accurate.

The VCS notifier follows this guideline by using OIDs with second-last ID as zero, enabling reversible translations.

### severityId

This variable indicates the severity of the trap being sent. It can take the following values:

| Severity Level and Description                                          | Value in Trap PDU |
|-------------------------------------------------------------------------|-------------------|
| Information<br>Important events exhibiting normal behavior              | 0                 |
| Warning<br>Deviation from normal behavior                               | 1                 |
| Error<br>A fault                                                        | 2                 |
| Severe Error<br>Critical error that can lead to data loss or corruption | 3                 |



## entityType and entitySubType

These variables specify additional information about the entity.

| Entity Type | Entity Sub-type                                                                                      |
|-------------|------------------------------------------------------------------------------------------------------|
| Resource    | String. For example, disk.                                                                           |
| Group       | The type of the group: <ul style="list-style-type: none"><li>■ Failover</li><li>■ Parallel</li></ul> |
| System      | String. For example, Solaris 2.8.                                                                    |
| Heartbeat   | The type of the heartbeat.                                                                           |
| VCS         | String                                                                                               |
| GCO         | String                                                                                               |
| Agent name  | Agent name                                                                                           |

## entityState

This variable describes the state of the entity.

| Entity           | States                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| VCS states       | <ul style="list-style-type: none"><li>■ Cluster has hung (reserved for future use)</li><li>■ User has logged into VCS</li><li>■ Cluster has faulted</li><li>■ Cluster is in RUNNING state</li><li>■ Cluster has lost heartbeat with remote cluster</li><li>■ Heartbeat with remote cluster is alive</li></ul>                                                                                                                                      |
| Agent states     | <ul style="list-style-type: none"><li>■ Agent is restarting</li><li>■ Agent has faulted</li></ul>                                                                                                                                                                                                                                                                                                                                                  |
| Resources states | <ul style="list-style-type: none"><li>■ Resource state is unknown</li><li>■ Resource monitoring has timed out</li><li>■ Resource is not going offline</li><li>■ Resource is being restarted by agent</li><li>■ Resource went online by itself</li><li>■ Resource has faulted</li><li>■ Resource health declined</li><li>■ Resource health improved</li><li>■ Resource is in admin wait state</li><li>■ Resource monitor time has changed</li></ul> |

| Entity               | States                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Service group states | <div><div>■</div>Service group is online</div> <div><div>■</div>Service group is offline</div> <div><div>■</div>Service group is auto disabled</div> <div><div>■</div>Service group has faulted</div> <div><div>■</div>Service group has faulted and cannot be failed over anywhere</div> <div><div>■</div>Service group is restarting</div> <div><div>■</div>Service group is being switched</div> <div><div>■</div>Service group concurrency violation</div> <div><div>■</div>Service group is restarting in response to persistent resource going online</div> <div><div>■</div>Service group attribute value does not match corresponding remote group attribute value</div> <div><div>■</div>Global group concurrency violation</div> |
| System states        | <div><div>■</div>VCS is up on the first node in the Cluster</div> <div><div>■</div>VCS is being restarted by hashadow</div> <div><div>■</div>VCS is in jeopardy</div> <div><div>■</div>VCS has faulted</div> <div><div>■</div>A node running VCS has joined cluster</div> <div><div>■</div>VCS is up but not in cluster</div> <div><div>■</div>VCS has exited manually</div> <div><div>■</div>CPU Usage exceeded the threshold on the system</div> <div><div>■</div>SWAP Usage exceeded the threshold</div>                                                                                                                                                                                                                                |
| GCO heartbeat states | <div><div>■</div>Cluster has lost heartbeat with remote cluster</div> <div><div>■</div>Heartbeat with remote cluster is alive</div>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |

# Monitoring aggregate events

This section describes how you can detect aggregate events by monitoring individual notifications.

## How to detect service group failover

VCS does not send any explicit traps when a failover occurs in response to a service group fault. When a service group faults, VCS generates the following notifications if the AutoFailOver attribute for the service group is set to 1:

- Service Group Fault for the node on which the service group was online and faulted
- Service Group Offline for the node on which the service group faulted
- Service Group Online for the node to which the service group failed over.

## How to detect service group switch

When a service group is switched, VCS sends notification to indicate the following events:

- Service group is being switched
- Service Group Offline for the node from which the service group is switched
- Service Group Online for the node to which the service group was switched. This notification is sent after VCS completes the service group switch operation.

---

**Note:** You must configure appropriate severity for the notifier to receive these notifications. To receive VCS notifications, the minimum acceptable severity level is Information.

---

# Detecting complementary events

Table 13-1 lists some events that complement each other, or cancel each other out.

Table 13-1 Complementary events in VCS

| Event                                                  | Cancelling event                       |
|--------------------------------------------------------|----------------------------------------|
| Remote cluster has faulted.<br>(Global Cluster Option) | Remote cluster is in RUNNING state.    |
| Heartbeat is down.                                     | Heartbeat is alive.                    |
| Agent is faulted                                       | Agent is restarting                    |
| Resource state is unknown.                             | Resource went online by itself.        |
| Health of cluster resource declined.                   | Health of cluster resource improved.   |
| VCS has faulted.                                       | A node running VCS has joined cluster. |
| Service group has faulted.                             | Service group is online.               |
| Service group is offline.                              | Service group is online                |
| Service group is being switched.                       | Service group is online                |

# Configuring notification

Configuring notification involves creating a resource for the Notifier Manager (NotifierMgr) agent in the ClusterService group. See the *Veritas Cluster Server Bundled Agents Reference Guide* for more information about the agent.

VCS provides several methods for configuring notification:

- Manually editing the main.cf file.
- Using the Notifier wizard.  
See “[Setting up VCS event notification using the Notifier wizard](#)” on page 170.

# VCS event triggers

- [About VCS event triggers](#)
- [Using event triggers](#)
- [List of event triggers](#)

## About VCS event triggers

Triggers let you invoke user-defined scripts for specified events in a cluster. VCS determines if the event is enabled and invokes the `hatrigger` script. The script is located at:

`$VCS_HOME/bin/hatrigger`

VCS also passes the name of the event trigger and associated parameters. For example, when a service group comes online on a system, VCS invokes the following command:

```
hatrigger -postonline system service_group.
```

VCS does not wait for the trigger to complete execution. VCS calls the trigger and continues normal operation.

VCS invokes event triggers on the system where the event occurred, with the following exceptions:

- VCS invokes the `sysoffline` and `nofailover` event triggers on the lowest-numbered system in the `RUNNING` state.
- VCS invokes the `violation` event trigger on all systems on which the service group was brought partially or fully online.

## Using event triggers

VCS provides a sample Perl script for each event trigger at the following location:

`$VCS_HOME/bin/sample_triggers`

Customize the scripts according to your requirements: you may choose to write your own Perl scripts.

### To use an event trigger

- 1 Use the sample scripts to write your own custom actions for the trigger.
- 2 Move the modified trigger script to the following path on each node:  
`$VCS_HOME/bin/triggers`
- 3 Configure other attributes that may be required to enable the trigger. See the usage information for the trigger for more information.  
See [“List of event triggers”](#) on page 447.

# List of event triggers

The information in the following sections describes the various event triggers, including their usage, parameters, and location.

## cpuusage event trigger

|                        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Description            | <p>The cpuusage event trigger is invoked when the system's CPU usage exceeds the ActionThreshold value of the system's CPUUsageMonitoring attribute for a duration longer than the ActionTimeLimit value. The trigger is not invoked if it was invoked on the system within the last five minutes.</p> <p>See <a href="#">“When a resource comes online”</a> on page 568.</p> <p>This event trigger is configurable.</p>                                                                                                                                                                         |
| Usage                  | <p>– <code>cpuusage</code> <i>triggertype</i> <i>system</i> <i>cpu_usage</i></p> <ul style="list-style-type: none"><li>■ <i>triggertype</i>—represents whether trigger is custom (<i>triggertype</i>=0) or internal (<i>triggertype</i>=1).<br/>If 0, the trigger is invoked from:<br/>/opt/VRTSvcs/bin/triggers/cpuusage<br/>If 1, the system reboots by invoking the trigger from:<br/>/opt/VRTSvcs/bin/internal_triggers/cpuusage</li><li>■ <i>system</i>—represents the name of the system.</li><li>■ <i>cpu_usage</i>—represents the percentage of CPU utilization on the system.</li></ul> |
| To enable the trigger  | <p>Set following values in the system's CPUUsageMonitoring attribute:</p> <ul style="list-style-type: none"><li>■ Enabled = 1</li><li>■ ActionTimeLimit = Non-zero value representing time in seconds.</li><li>■ ActionThreshold = Non-zero value representing CPU percentage utilization.</li><li>■ Action = CUSTOM or REBOOT.<br/>CUSTOM—Invokes trigger from:<br/>/opt/VRTSvcs/bin/triggers/cpuusage<br/>REBOOT—invokes trigger from:<br/>/opt/VRTSvcs/bin/internal_triggers/cpuusage<br/>and the system reboots.</li></ul>                                                                   |
| To disable the trigger | <p>Set one of the following values in CPUUsageMonitoring system attribute to 0 for the system:</p> <ul style="list-style-type: none"><li>■ ActionTimeLimit = 0</li><li>■ ActionThreshold = 0</li></ul>                                                                                                                                                                                                                                                                                                                                                                                           |

## injeopardy event trigger

**Description** Invoked when a system is in jeopardy. Specifically, this trigger is invoked when a system has only one remaining link to the cluster, and that link is a network link (LLT). This event is a considered critical because if the system loses the remaining network link, VCS does not fail over the service groups that were online on the system. Use this trigger to notify the administrator of the critical event. The administrator can then take appropriate action to ensure that the system has at least two links to the cluster.

This event trigger is non-configurable.

**Usage** - `injeopardy triggertype system system_state`

- *triggertype*—represents whether trigger is custom (*triggertype*=0) or internal (*triggertype*=1).  
For this trigger, *triggertype*=0.
- *system*—represents the name of the system.
- *system\_state*—represents the value of the State attribute.



## loadwarning event trigger

**Description** Invoked when a system becomes overloaded because the load of the system's online groups exceeds the system's LoadWarningLevel attribute for an interval exceeding the LoadTimeThreshold attribute.

For example, say the Capacity is 150, the LoadWarningLevel is 80, and the LoadTimeThreshold is 300. Also, the sum of the Load attribute for all online groups on the system is 135. Because the LoadWarningLevel is 80, safe load is  $0.80 \times 150 = 120$ . Actual system load is 135. If system load stays above 120 for more than 300 seconds, the LoadWarningLevel trigger is invoked.

Use this trigger to notify the administrator of the critical event. The administrator can then switch some service groups to another system, ensuring that no one system is overloaded.

This event trigger is non-configurable.

**Usage** - loadwarning *triggertype* *system* *available\_capacity*

- *triggertype*—represents whether trigger is custom (*triggertype*=0) or internal (*triggertype*=1).  
For this trigger, *triggertype*=0.
- *system*—represents the name of the system.
- *available\_capacity*—represents the system's AvailableCapacity attribute. (AvailableCapacity=Capacity-sum of Load for system's online groups.)

## multinicb event trigger

**Description** Invoked when a network device that is configured as a MultiNICB resource changes its state. The trigger is also always called in the first monitor cycle. VCS provides a sample trigger script for your reference. You can customize the sample script according to your requirements.

**Usage** `-multinicb_postchange triggertype resource-name device-name previous-state current-state monitor_heartbeat`

- *triggertype*—represents whether trigger is custom (*triggertype=0*) or internal (*triggertype=1*).  
For this trigger, *triggertype=0*.
- *resource-name*—represents the MultiNICB resource that invoked this trigger.
- *device-name*—represents the network interface device for which the trigger is called.
- *previous-state*—represents the state of the device before the change. The value 1 indicates that the device is up; 0 indicates it is down.
- *current-state*—represents the state of the device after the change.
- *monitor-heartbeat*—an integer count, which is incremented in every monitor cycle. The value 0 indicates that the monitor routine is called for first time

## nfs\_postoffline event trigger

|                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Description</b> | Restarts the following daemons after the group goes offline: <ul style="list-style-type: none"><li>■ nfsd</li><li>■ mountd</li><li>■ lockd</li><li>■ statd</li><li>■ nfsrgyd</li></ul> When you set a value for the NFS agent's NFSv4Root attribute, restarts the nfsrgyd daemon.                                                                                                                                                                                                         |
|                    | <ul style="list-style-type: none"><li>■ gssd</li></ul> When you set a value of 1 for the NFS agent's NFSSecurity attribute, restarts the gssd daemon.                                                                                                                                                                                                                                                                                                                                     |
| <b>Usage</b>       | <p>- nfs_postoffline <i>triggertype system service_group</i></p> <ul style="list-style-type: none"><li>■ <i>triggertype</i>—represents whether trigger is custom (<i>triggertype=0</i>) or internal (<i>triggertype=1</i>).</li><li>■ For this trigger, <i>triggertype=0</i>.</li><li>■ <i>system</i>—represents the name of the last system on which an attempt was made to online the service group.</li><li>■ <i>service_group</i>—represents the name of the service group.</li></ul> |

## nfs\_preonline event trigger

|                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Description</b> | Terminates the following daemons when the value of the NFSRestart agent's NFLockFailover attribute is 1. <ul style="list-style-type: none"><li>■ statd</li><li>■ lockd</li></ul>                                                                                                                                                                                                                                                                                                        |
| <b>Usage</b>       | <p>- nfs_preonline <i>triggertype system service_group</i></p> <ul style="list-style-type: none"><li>■ <i>triggertype</i>—represents whether trigger is custom (<i>triggertype=0</i>) or internal (<i>triggertype=1</i>).</li><li>■ For this trigger, <i>triggertype=0</i>.</li><li>■ <i>system</i>—represents the name of the last system on which an attempt was made to online the service group.</li><li>■ <i>service_group</i>—represents the name of the service group.</li></ul> |

## nofailover event trigger

- Description**
- Called from the lowest-numbered system in RUNNING state when a service group cannot fail over.  
This event trigger is non-configurable.
- Usage**
- `-nofailover`

`triggertype system service_group`

- *triggertype*—represents whether trigger is custom (*triggertype*=0) or internal (*triggertype*=1).  
For this trigger, *triggertype*=0.
  - *system*—represents the name of the last system on which an attempt was made to online the service group.
  - *service\_group*—represents the name of the service group.

## postoffline event trigger

- Description**
- This event trigger is invoked on the system where the group went offline from a partial or fully online state. This trigger is invoked when the group faults, or is taken offline manually.  
This event trigger is non-configurable.
- Usage**
- `-postoffline`

`triggertype system service_group`

- *triggertype*—represents whether trigger is custom (*triggertype*=0) or internal (*triggertype*=1).  
For this trigger, *triggertype*=0.
  - *system*—represents the name of the system.
  - *service\_group*—represents the name of the service group that went offline.

## postonline event trigger

- Description**
- This event trigger is invoked on the system where the group went online from an offline state.  
This event trigger is non-configurable.

**Usage**

-postonline *triggertype system service\_group*

- *triggertype*—represents whether trigger is custom (*triggertype*=0) or internal (*triggertype*=1).  
For this trigger, *triggertype*=0.
- *system*—represents the name of the system.
- *service\_group*—represents the name of the service group that went online.

## preonline event trigger

|                        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Description            | <p>Indicates that when the HAD should call a user-defined script before bringing a service group online in response to the <code>hagrp -online</code> command or a fault.</p> <p>If the trigger does not exist, VCS continues to bring the group online. If the script returns 0 without an exit code, VCS runs the <code>hagrp -online -nopre</code> command, with the <code>-checkpartial</code> option if appropriate.</p> <p>If you do want to bring the group online, define the trigger to take no action. This event trigger is configurable.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| Usage                  | <p><code>-preonline <i>triggertype system service_group whyonlining [system_where_group_faulted]</i></code></p> <ul style="list-style-type: none"><li>■ <i>triggertype</i>—represents whether trigger is custom (<i>triggertype</i>=0) or internal (<i>triggertype</i>=1).<br/>For this trigger, <i>triggertype</i>=0.</li><li>■ <i>system</i>—represents the name of the system.</li><li>■ <i>service_group</i>—represents the name of the service group on which the <code>hagrp</code> command was issued or the fault occurred.</li><li>■ <i>whyonlining</i>—represents two values:<br/>FAULT: Indicates that the group was brought online in response to a group failover or switch.<br/>MANUAL: Indicates that group was brought online manually on the system that is represented by the variable <i>system</i>.</li><li>■ <i>system_where_group_faulted</i>—represents the name of the system on which the group has faulted or switched. This variable is optional and set when the engine invokes the trigger during a failover or switch.</li></ul> |
| To enable the trigger  | <p>Set the PreOnline attribute in the service group definition to 1.</p> <p>You can set a local (per-system) value for the attribute to control behavior on each node in the cluster.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| To disable the trigger | <p>Set the PreOnline attribute in the service group definition to 0.</p> <p>You can set a local (per-system) value for the attribute to control behavior on each node in the cluster.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

## preonline\_ipc event trigger

|                               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Description</b>            | <p>Performs the following actions before a service containing IP resources is brought online:</p> <ul style="list-style-type: none"><li>■ Verifies that IP addresses configured for IP resources in the service group are not online anywhere else.<br/>If a configured IP address is online somewhere else, the trigger waits for 5 seconds. If the configured IP is online after 5 seconds, the trigger aborts the online operation for the IP resource.</li><li>■ Verifies that the node is not isolated from other nodes in the cluster by pinging the broadcast address and checking the heartbeat links. If the trigger does not find any heartbeat links and the broadcast ping fails, the trigger aborts the online operation to prevent possible split brain condition.</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>Usage</b>                  | <p><code>-preonline <i>triggertype</i> <i>system</i> <i>service_group</i> <i>whyonlining</i> [<i>system_where_group_faulted</i>]</code></p> <ul style="list-style-type: none"><li>■ <i>triggertype</i>—represents whether trigger is custom (<i>triggertype</i>=0) or internal (<i>triggertype</i>=1).<br/>For this trigger, <i>triggertype</i>=0.</li><li>■ <i>system</i>—represents the name of the system.</li><li>■ <i>service_group</i>—represents the name of the service group on which the <code>hagrp</code> command was issued or the fault occurred.</li><li>■ <i>whyonlining</i>—represents two values:<br/>FAULT: Indicates that the group was brought online in response to a group failover or switch.<br/>MANUAL: Indicates that group was brought online manually on the system that is represented by the variable <i>system</i>.</li><li>■ <i>system_where_group_faulted</i>—represents the name of the system on which the group has faulted or switched. This variable is optional and set when the engine invokes the trigger during a failover or switch.</li></ul> <p><b>Note:</b> To use this trigger, copy the <code>preonline_ipc</code> script from <code>\$VCS_HOME/bin/sample_triggers</code> to <code>\$VCS_HOME/bin/triggers/</code> and rename the script to <code>preonline</code>. If you use the <code>preonline</code> trigger for other purposes, merge the contents of the <code>preonline_ipc</code> script with those of the <code>preonline</code> script.</p> |
| <b>To enable the trigger</b>  | <p>Set the PreOnline attribute in the service group definition to 1.</p> <p>You can set a local (per-system) value for the attribute to control behavior on each node in the cluster.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>To disable the trigger</b> | <p>Set the PreOnline attribute in the service group definition to 0.</p> <p>You can set a local (per-system) value for the attribute to control behavior on each node in the cluster.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |

## resadminwait event trigger

**Description** Invoked when a resource enters ADMIN\_WAIT state.

When VCS sets a resource in the ADMIN\_WAIT state, it invokes the resadminwait trigger according to the reason the resource entered the state.

See “[Clearing resources in the ADMIN\\_WAIT state](#)” on page 369.

This event trigger is non-configurable.

**Usage** - `resadminwait system resource adminwait_reason`

- *system*—represents the name of the system.
- *resource*—represents the name of the faulted resource.
- *adminwait\_reason*—represents the reason the resource entered the ADMIN\_WAIT state. Values range from 0-5:
  - 0 = The offline function did not complete within the expected time.
  - 1 = The offline function was ineffective.
  - 2 = The online function did not complete within the expected time.
  - 3 = The online function was ineffective.
  - 4 = The resource was taken offline unexpectedly.
  - 5 = The monitor function consistently failed to complete within the expected time.



## resfault event trigger

|                       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Description           | <p>Invoked on the system where a resource has faulted. Note that when a resource is faulted, resources within the upward path of the faulted resource are also brought down.</p> <p>This event trigger is configurable.</p> <p>To configure this trigger, you must define the following:</p> <p>TriggerResFault: Set the attribute to 1 to invoke the trigger when a resource faults.</p>                                                                                                                             |
| Usage                 | <p>- resfault <i>triggertype</i> <i>system</i> <i>resource</i> <i>previous_state</i></p> <ul style="list-style-type: none"><li>■ <i>triggertype</i>—represents whether trigger is custom (<i>triggertype</i>=0) or internal (<i>triggertype</i>=1).<br/>For this trigger, <i>triggertype</i>=0.</li><li>■ <i>system</i>—represents the name of the system.</li><li>■ <i>resource</i>—represents the name of the faulted resource.</li><li>■ <i>previous_state</i>—represents the resource's previous state.</li></ul> |
| To enable the trigger | <p>To invoke the trigger when a resource faults, set the TriggerResFault attribute to 1.</p>                                                                                                                                                                                                                                                                                                                                                                                                                          |

## resnotoff event trigger

|             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Description | <p>Invoked on the system if a resource in a service group does not go offline even after issuing the offline command to the resource.</p> <p>This event trigger is configurable.</p> <p>To configure this trigger, you must define the following:</p> <p><b>Resource Name</b> Define resources for which to invoke this trigger by entering their names in the following line in the script: @resources = ( "resource1", "resource2" ) ;</p> <p>If any of these resources do not go offline, the trigger is invoked with that resource name and system name as arguments to the script.</p> |
|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

### Usage

- resnotoff *triggertype system resource*

- *triggertype*—represents whether trigger is custom (*triggertype*=0) or internal (*triggertype*=1).  
For this trigger, *triggertype*=0.
- *system*—represents the system on which the resource is not going offline.
- *resource*—represents the name of the resource.

## resstatechange event trigger

**Description** This trigger is invoked under the following conditions:

- Resource goes from OFFLINE to ONLINE.
- Resource goes from ONLINE to OFFLINE.
- Resource goes from ONLINE to FAULTED.
- Resource goes from FAULTED to OFFLINE. (When fault is cleared on non-persistent resource.)
- Resource goes from FAULTED to ONLINE. (When faulted persistent resource goes online or faulted non-persistent resource is brought online outside VCS control.)
- Resource is restarted by an agent because resource faulted and RestartLimit was greater than 0.

This event trigger is configurable.

**Usage**      `- resstatechange triggertype system resource previous_state new_state`

- *triggertype*—represents whether trigger is custom (*triggertype*=0) or internal (*triggertype*=1).  
For this trigger, *triggertype*=0.
- *system*—represents the name of the system.
- *resource*—represents the name of the resource.
- *previous\_state*—represents the resource's previous state.
- *new\_state*—represents the resource's new state.

**To enable the trigger**      This event trigger is not enabled by default. You must enable resstatechange by setting the attribute TriggerResStateChange to 1 in the main.cf file, or by issuing the command:

```
hagrp -modify service_group TriggerResStateChange 1
```

**Note:** Use the resstatechange trigger carefully. For example, enabling this trigger for a service group with 100 resources means 100 hattrigger processes and 100 resstatechange processes are fired each time the group is brought online or taken offline. Also, this is not a “wait-mode trigger. Specifically, VCS invokes the trigger and does not wait for trigger to return to continue operation

## sysoffline event trigger

**Description** Called from the lowest-numbered system in RUNNING state when a system leaves the cluster.

This event trigger is non-configurable.

**Usage**

- `sysoffline system system_state`
  - *system*—represents the name of the system.
  - *system\_state*—represents the value of the State attribute.

See “[System states](#)” on page 638.

## sysup event trigger

**Description** Called when the first system comes up in the cluster

This event trigger is configurable.

**Usage**

- `sysup triggertype system system_state`
  - *triggertype*—represents whether trigger is custom (*triggertype*=0) or internal (*triggertype*=1).  
For this trigger, *triggertype*=0.
  - *system*—represents the name of the system.
  - *system\_state*—state of the system (RUNNING in this case)

See “[System states](#)” on page 638.

## sysjoin event trigger

**Description** Called when a new system joins the cluster. The trigger is invoked on all nodes in the cluster.

This event trigger is configurable.

**Usage**

- `sysjoin triggertype system system_state`
  - *triggertype*—represents whether trigger is custom (*triggertype*=0) or internal (*triggertype*=1).
  - For this trigger, *triggertype*=0.*system*—represents the name of the system.
  - *system\_state*—represents the value of the State attribute.

See “[System states](#)” on page 638.

## unable\_to\_restart\_agent event trigger

**Description** This trigger is invoked when an agent faults more than a predetermined number of times with in an hour. When this occurs, VCS gives up trying to restart the agent. VCS invokes this trigger on the node where the agent faults.

You can use this trigger to notify the administrators that an agent has faulted, and that VCS is unable to restart the agent. The administrator can then take corrective action.

**Usage**

- `unable_to_restart_agent system resource_type`
  - *system*—represents the name of the system.
  - *resource\_type*—represents the resource type associated with the agent.

**To disable the trigger** Remove the files associated with the trigger from the `$VCS_HOME/bin/triggers` directory.

## unable\_to\_restart\_had event trigger

**Description** This event trigger is invoked by hashadow when hashadow cannot restart HAD on a system. If HAD fails to restart after six attempts, hashadow invokes the trigger on the system.

The default behavior of the trigger is to reboot the system. However, service groups previously running on the system are autodisabled when hashadow fails to restart HAD. Before these service groups can be brought online elsewhere in the cluster, you must autoenable them on the system. To do so, customize the `unable_to_restart_had` trigger to remotely execute the following command from any node in the cluster where VCS is running:

```
hagrp -autoenable service_group -sys system
```

For example, if hashadow fails to restart HAD on *system1*, and if *group1* and *group2* were online on that system, a trigger customized in this manner would autoenable *group1* and *group2* on *system1* before rebooting. Autoenabling *group1* and *group2* on *system1* enables these two service groups to come online on another system when the trigger reboots *system1*.

This event trigger is non-configurable.

**Usage** `-unable_to_restart_had`

This trigger has no arguments.

## violation event trigger

**Description** This trigger is invoked only on the system that caused the concurrency violation. Specifically, it takes the service group offline on the system where the trigger was invoked. Note that this trigger applies to failover groups only. The default trigger takes the service group offline on the system that caused the concurrency violation.

This event trigger is non-configurable.

**Usage** `-violation system service_group`

- *system*—represents the name of the system.
- *service\_group*—represents the name of the service group that was fully or partially online.



## Cluster configurations for disaster recovery

- [Chapter 15, “Connecting clusters–Creating global clusters” on page 465](#)
- [Chapter 16, “Administering global clusters from Cluster Manager \(Java console\)” on page 507](#)
- [Chapter 17, “Administering global clusters from the command line” on page 525](#)
- [Chapter 18, “Setting up replicated data clusters” on page 539](#)
- [Chapter 19, “Setting up campus clusters” on page 547](#)





# Connecting clusters– Creating global clusters

- [How VCS global clusters work](#)
- [VCS global clusters: The building blocks](#)
- [Prerequisites for global clusters](#)
- [About setting up global clusters](#)
- [Setting up a global cluster](#)
- [When a cluster faults](#)
- [Setting up a fire drill](#)
- [Multi-tiered application support using the RemoteGroup agent in a global environment](#)
- [Test scenario for a multi-tiered environment](#)

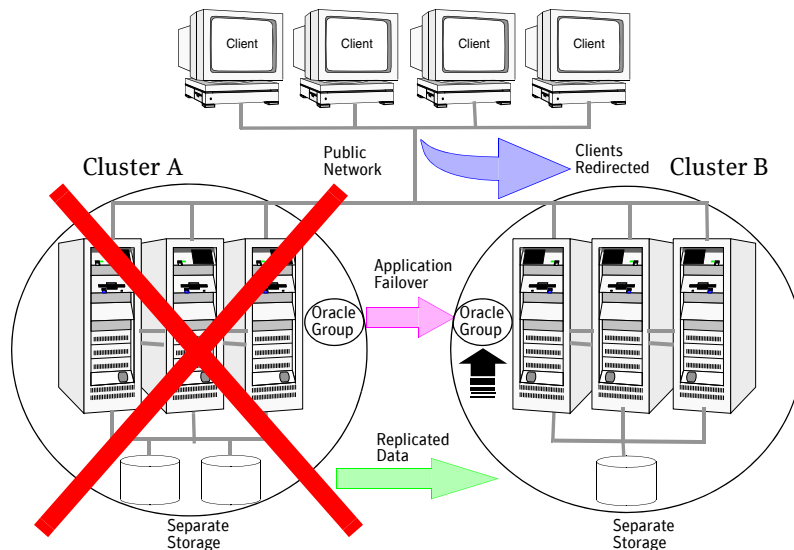
## How VCS global clusters work

Local clustering provides local failover for each site or building. But, these configurations do not provide protection against large-scale disasters such as major floods, hurricanes, and earthquakes that cause outages for an entire city or region. The entire cluster could be affected by an outage.

In such situations, VCS global clusters ensure data availability by migrating applications to remote clusters located considerable distances apart.

Let us take the example of an Oracle database configured in a VCS global cluster. Oracle is installed and configured in both clusters. Oracle data is located on shared disks within each cluster and is replicated across clusters to ensure data concurrency. The Oracle service group is online on a system in cluster A and is configured to fail over globally, on clusters A and B.

**Figure 15-1** Sample global cluster setup



VCS continuously monitors and communicates events between clusters. Inter-cluster communication ensures that the global cluster is aware of the state of the service groups that are configured in the global cluster at all times.

In the event of a system or application failure, VCS fails over the Oracle service group to another system in the same cluster. If the entire cluster fails, VCS fails over the service group to the remote cluster, which is part of the global cluster. VCS also redirects clients once the application is online on the new location.

# VCS global clusters: The building blocks

VCS extends clustering concepts to wide-area high availability and disaster recovery with the following:

- [Visualization of remote cluster objects](#)
- [Global service groups](#)
- [Global cluster management](#)
- [Serialization—The Authority attribute](#)
- [Resiliency and “Right of way”](#)
- [VCS agents to manage wide-area failover](#)
- [The Steward process: Split-brain in two-cluster global clusters](#)

## Visualization of remote cluster objects

VCS enables you to visualize remote cluster objects using the VCS command-line, the Java Console, and VCS Management Console.

You can define remote clusters in your configuration file, `main.cf`. The Remote Cluster Configuration wizard provides an easy interface to do so. The wizard updates the `main.cf` files of all connected clusters with the required configuration changes.

See [“Adding a remote cluster”](#) on page 509.

## Global service groups

A *global* service group is a regular VCS group with additional properties to enable wide-area failover. The global service group attribute `ClusterList` defines the list of clusters to which the group can fail over. The service group must be configured on all participating clusters and must have the same name on each cluster. The Global Group Configuration wizard provides an easy interface to configure global groups.

See [“Administering global service groups”](#) on page 516.

VCS agents manage the replication during cross-cluster failover.

See [“VCS agents to manage wide-area failover”](#) on page 471.

## Global cluster management

VCS enables you to perform operations (online, offline, switch) on global service groups from any system in any cluster. You must log on with adequate privileges for cluster operations.

See “[User privileges in global clusters](#)” on page 75.

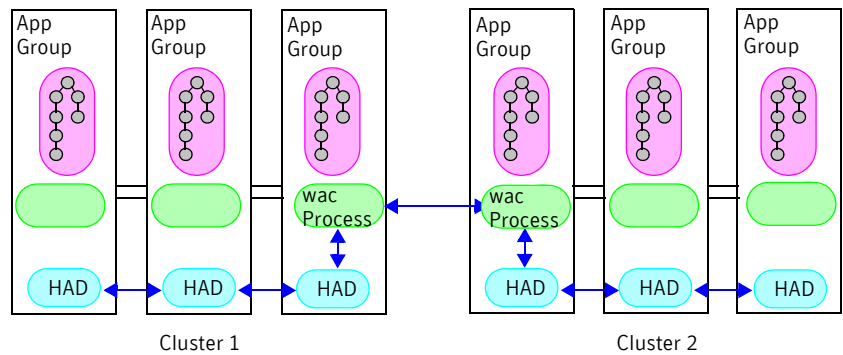
You can bring service groups online or switch them to any system in any cluster. If you do not specify a target system, VCS uses the FailOverPolicy to determine the system.

See “[Defining failover policies](#)” on page 366.

Management of remote cluster objects is aided by inter-cluster communication enabled by the wide-area connector (wac) process.

### Wide-area connector process

The wide-area connector (wac) is a failover Application resource that ensures communication between clusters.



The wac process runs on one system in each cluster and connects with peers in remote clusters. It receives and transmits information about the status of the cluster, service groups, and systems. This communication enables VCS to create a consolidated view of the status of all the clusters configured as part of the global cluster. The process also manages wide-area heartbeating to determine the health of remote clusters. The process also transmits commands between clusters and returns the result to the originating cluster.

VCS provides the option of securing the communication between the wide-area connectors.

See “[Secure communication in global clusters](#)” on page 473.

## Wide-area heartbeats

The wide-area Heartbeat agent manages the inter-cluster heartbeat. Heartbeats are used to monitor the health of remote clusters.

See [“Heartbeat attributes \(for global clusters\)” on page 697](#).

You can change the default values of the heartbeat agents using the command **hahb -modify**

### Sample configuration

```
Heartbeat Icmp (
 ClusterList = {C1, C2}
 AYAInterval@C1 = 20
 AYAInterval@C1 = 30
 Arguments@c1 = {"192.168.10.10"}
 Arguments@c2 = {"64.203.10.12"}
)
```

## Serialization—The Authority attribute

VCS ensures that multi-cluster service group operations are conducted serially to avoid timing problems and to ensure smooth performance. The *Authority* attribute prevents a service group from coming online in multiple clusters at the same time. Authority is a persistent service group attribute and it designates which cluster has the right to bring a global service group online. The attribute cannot be modified at runtime.

If two administrators simultaneously try to bring a service group online in a two-cluster global group, one command is honored, and the other is rejected based on the value of the Authority attribute.

The attribute prevents bringing a service group online in a cluster that does not have the authority to do so. If the cluster holding authority is down, you can enforce a takeover by using the command `hagrp -online -force service_group`. This command enables you to fail over an application to another cluster when a disaster occurs.

---

**Note:** A cluster assuming authority for a group does not guarantee the group will be brought online on the cluster. The attribute merely specifies the right to attempt bringing the service group online in the cluster. The presence of Authority does not override group settings like frozen, autodisabled, non-probed, and so on, that prevent service groups from going online.

---

You must seed authority if it is not held on any cluster.

Offline operations on global groups can originate from any cluster and do not require a change of authority to do so, because taking a group offline does not necessarily indicate an intention to perform a cross-cluster failover.

## Authority and AutoStart

The attributes Authority and AutoStart work together to avoid potential concurrency violations in multi-cluster configurations.

If the AutoStartList attribute is set, and if a group's Authority attribute is set to 1, the VCS engine waits for the wac process to connect to the peer. If the connection fails, it means the peer is down and the AutoStart process proceeds. If the connection succeeds, HAD waits for the remote snapshot. If the peer is holding the authority for the group and the remote group is online (because of takeover), the local cluster does not bring the group online and relinquishes authority.

If the Authority attribute is set to 0, AutoStart is not invoked.

## Resiliency and “Right of way”

VCS global clusters maintain resiliency using the wide-area connector process and the ClusterService group. The wide-area connector process runs as long as there is at least one surviving node in a cluster.

The wide-area connector, its alias, and notifier are components of the ClusterService group.

## VCS agents to manage wide-area failover

VCS agents now manage external objects that are part of wide-area failover. These objects include replication, DNS updates, and so on. These agents provide a robust framework for specifying attributes and restarts, and can be brought online upon fail over.

### DNS agent

The DNS agent updates the canonical name-mapping in the domain name server after a wide-area failover.

See the *Veritas Cluster Server Bundled Agents Reference Guide* for more information.

### VCS agents for VVR

You can use the following VCS agents for VVR in a VCS global cluster setup:

- **RVG agent**  
The RVG agent manages the Replicated Volume Group (RVG). Specifically, it brings the RVG online, monitors read-write access to the RVG, and takes the RVG offline. Use this agent when using VVR for replication.
- **RVGPrimary agent**  
The RVGPrimary agent attempts to migrate or take over a Secondary to a Primary following an application failover. The agent has no actions associated with the offline and monitor routines.
- **RVGSnapshot agent**  
The RVGSnapshot agent, used in fire drill service groups, takes space-optimized snapshots so that applications can be mounted at secondary sites during a fire drill operation.

See the *Veritas Cluster Server Agents for Veritas Volume Replicator Configuration Guide* for more information.

### VCS agents for third-party replication technologies

VCS provides agents for other third-party array-based or application-based replication solutions. These agents are available in the Veritas High Availability Agent Pack software.

See the *Veritas High Availability Agent Pack Getting Started Guide* for a list of replication technologies that VCS supports.

See the *Veritas Cluster Server Agent for <Replication Solution> Installation and Configuration Guide* for more information.

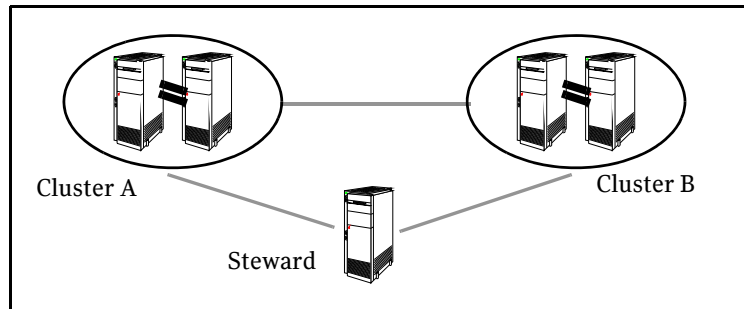
## The Steward process: Split-brain in two-cluster global clusters

Failure of all heartbeats between any two clusters in a global cluster indicates one of the following:

- The remote cluster is faulted.
- All communication links between the two clusters are broken.

In global clusters with more than three clusters, VCS queries the connected clusters to confirm that the remote cluster is truly down. This mechanism is called *inquiry*.

In a two-cluster setup, VCS uses the *Steward* process to minimize chances of a wide-area split-brain. The process runs as a standalone binary on a system outside of the global cluster configuration.



When all communication links between any two clusters are lost, each cluster contacts the Steward with an inquiry message. The Steward sends an ICMP ping to the cluster in question and responds with a negative inquiry if the cluster is running or with positive inquiry if the cluster is down. The Steward can also be used in configurations with more than two clusters.

VCS provides the option of securing communication between the Steward process and the wide-area connectors.

See [“Secure communication in global clusters”](#) on page 473.

A Steward is effective only if there are independent paths from each cluster to the host that runs the Steward. If there is only one path between the two clusters, you must prevent split-brain by confirming manually via telephone or some messaging system with administrators at the remote site if a failure has occurred. By default, VCS global clusters fail over an application across cluster boundaries with administrator confirmation. You can configure automatic failover by setting the ClusterFailOverPolicy attribute to Auto.

See [“Administering the cluster from the Cluster Management Console”](#) on page 77.



The default port for the steward is 14156.

## Secure communication in global clusters

In global clusters, VCS provides the option of making the following communications secure:

- Communication between the wide-area connectors.
- Communication between the wide-area connectors and the Steward process.

For secure authentication, the wide-area connector process gets a security context as an account in the local authentication broker on each cluster node.

The WAC account belongs to the same domain as HAD and Command Server and is specified as:

```
name = _WAC_GCO_(systemname)
domain = HA_SERVICES@(fully_qualified_system_name)
```

You must configure the wide-area connector process in all clusters to run in secure mode. If the wide-area connector process runs in secure mode, you must run the Steward in secure mode.

See [“Configuring the Steward process \(optional\)”](#) on page 483.

See [“Prerequisites for clusters running in secure mode”](#) on page 476.

## Prerequisites for global clusters

This section describes the prerequisites for configuring global clusters.

### Cluster setup

You must have at least two clusters to set up a global cluster. Every cluster must have the required licenses. A cluster can be part of one global cluster. VCS supports a maximum of four clusters participating in a global cluster.

Clusters must be running on the same platform; the operating system versions can be different. Clusters must be using the same VCS version.

Cluster names must be unique within each global cluster; system and resource names need not be unique across clusters. Service group names need not be unique across clusters; however, global service groups must have identical names.

Every cluster must have a valid virtual IP address, which is tied to the cluster. Define this IP address in the cluster's ClusterAddress attribute. This address is normally configured as part of the initial VCS installation. The IP address must have a DNS entry.

The global cluster operations require that the port for Wide-Area Connector (WAC) process (default is 14155) are open across firewalls.

All clusters in a global cluster must use either IPv4 or IPv6 addresses. VCS does not support configuring clusters that use different Internet Protocol versions in a global cluster.

For remote cluster operations, you must configure a VCS user with the same name and privileges in each cluster.

See [“User privileges in global clusters”](#) on page 75.

### Application setup

Applications to be configured as global groups must be configured to represent each other in their respective clusters. The multiple application groups of a global group must have the same name in each cluster. The individual resources of the groups can be different. For example, one group might have a MultiNIC resource or more Mount-type resources. Client systems redirected to the remote cluster in case of a wide-area failover must be presented with the same application they saw in the primary cluster.

However, the resources that make up a global group must represent the same application from the point of the client as its peer global group in the other cluster. Clients redirected to a remote cluster should not be aware that a cross-

cluster failover occurred, except for some downtime while the administrator initiates or confirms the failover.

## Wide-area heartbeats

There must be at least one wide-area heartbeat going from each cluster to every other cluster. VCS starts communicating with a cluster only after the heartbeat reports that the cluster is *alive*. VCS uses the ICMP ping by default, the infrastructure for which is bundled with the product. VCS configures the ICMP heartbeat if you use Cluster Manager (Java Console) to set up your global cluster. Other heartbeats must be configured manually.

## ClusterService group

The ClusterService group must be configured with the Application (for the wide-area connector), NIC, and IP resources. The service group may contain additional groups or resources for VCS Management Console, Authentication Service, or notification, if these components are configured. The ClusterService group is configured automatically when VCS is installed or upgraded.

If you entered a license that includes VCS global cluster support during the VCS install or upgrade, the installer provides you an option to automatically configure a resource `wac` of type Application in the ClusterService group. The installer also configures the wide-area connector process.

You can run the GCO Configuration wizard later to configure the WAC process and to update the ClusterService group with an Application resource for WAC.

See [“Configuring clusters for global cluster setup”](#) on page 479.

## Replication setup

VCS global clusters are used for disaster recovery, so you must set up real-time data replication between clusters. You can set up replication using Veritas Volume Replicator or one of the third-party replication solutions that VCS supports. If you use a third-party replication solution, you must also install and configure VCS agent for the replication solution to manage the replication.

These agents are available in the Veritas High Availability Agent Pack software.

See the *Veritas High Availability Agent Pack Getting Started Guide* for a list of replication technologies that VCS supports.

## Prerequisites for clusters running in secure mode

If you plan to configure secure communication among clusters in the global clusters, then you must meet the following prerequisites:

- For both clusters to communicate in secure mode, both clusters must share a root broker or the root brokers must share a trust relationship.
- Both clusters must run in secure mode.
- If you plan to secure the communication between the wide-area connector processes, you must configure the processes in both clusters to run in secure mode.

When you configure the wac processes to run in secure mode, an AT account for the wac process is created automatically.

- If the wide-area connector process runs in secure mode, you must run the Steward in secure mode.

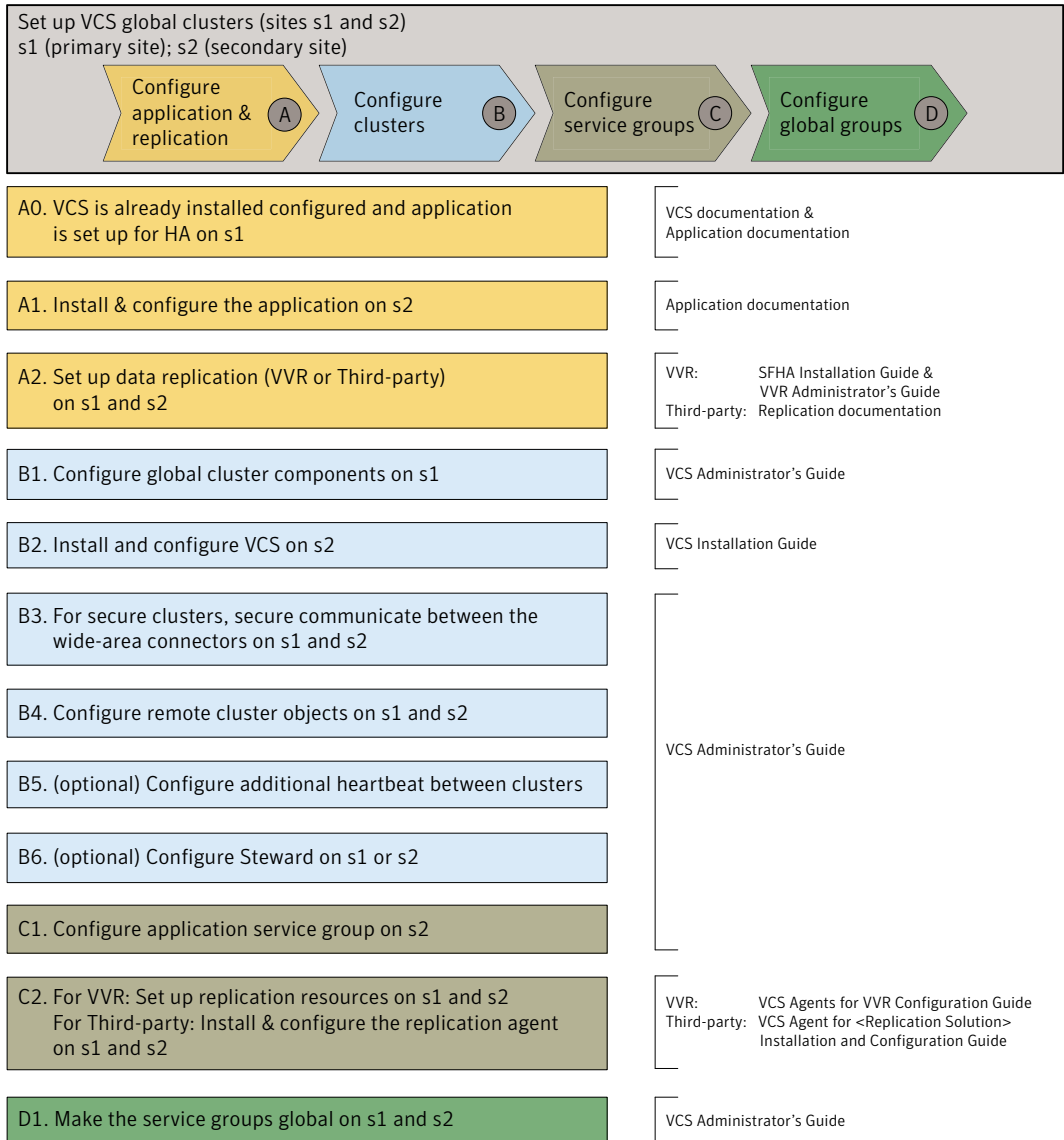
## About setting up global clusters

Before you set up the global cluster, make sure you completed the following:

- Review the concepts.  
See [“VCS global clusters: The building blocks”](#) on page 467.
- Plan the configuration and verify that you have the required physical infrastructure.  
See [“Prerequisites for global clusters”](#) on page 474.
- Verify that you have the required software to install VCS and a supported replication technology.  
If you use a third-party replication technology, then verify that you have the appropriate replication agent software.

[Figure 15-2](#) depicts the workflow to set up a global cluster where VCS cluster is already configured at site s1, and has application set up for high availability.

**Figure 15-2** High-level workflow to set up VCS global clusters



# Setting up a global cluster

This procedure assumes that you have configured a VCS cluster at the primary site and have set up application for high availability.

In this example, a single-instance Oracle database is configured as a VCS service group (appgroup) on a two-node cluster.

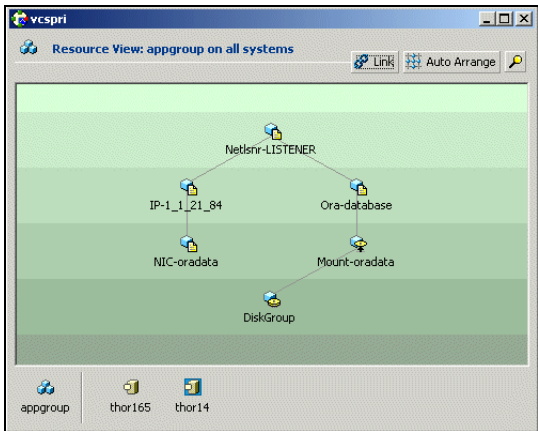


Table 15-2 lists the high-level tasks to set up VCS global clusters.

Table 15-2 Task map to set up VCS global clusters

| Task   | Reference                                                                           |
|--------|-------------------------------------------------------------------------------------|
| Task A | See “Configuring application and replication for global cluster setup” on page 478. |
| Task B | See “Configuring clusters for global cluster setup” on page 479.                    |
| Task C | See “Configuring service groups for global cluster setup” on page 485.              |
| Task D | See “Configuring global service groups for global cluster setup” on page 488.       |

## Configuring application and replication for global cluster setup

Perform the following steps to configure application at the secondary site. This procedure assumes that you have already set up application for high availability at the primary site.

### To configure application and replication

- 1 At the secondary site, install and configure the application that you want to make highly available.  
See the corresponding application documentation for instructions.
- 2 At each site, set up data replication using a replication technology that VCS supports:
  - Veritas Volume Replicator  
See the *Veritas Storage Foundation High Availability Installation Guide* and *Veritas Volume Replicator Administrator's Guide* for instructions.
  - A supported third-party replication technology  
See the corresponding replication documentation for instructions.  
See the *Veritas High Availability Agent Pack Getting Started Guide* for a list of third-party replication technologies that VCS supports.

## Configuring clusters for global cluster setup

Perform the following steps to configure the clusters for disaster recovery:

- See [“Configuring global cluster components at the primary site”](#) on page 479.
- See [“Installing and configuring VCS at the secondary site”](#) on page 480.
- See [“Securing communication between the wide-area connectors”](#) on page 481.
- See [“Configuring remote cluster objects”](#) on page 482.
- See [“Configuring additional heartbeat links \(optional\)”](#) on page 482.
- See [“Configuring the Steward process \(optional\)”](#) on page 483.

### Configuring global cluster components at the primary site

Perform the following steps to configure global cluster components at the primary site. If you have already completed these steps during the VCS cluster configuration at the primary site, then proceed to the next task to set up a VCS cluster at the secondary site.

See [“Installing and configuring VCS at the secondary site”](#) on page 480.

Run the GCO Configuration wizard to create or update the ClusterService group. The wizard verifies your configuration and validates it for a global cluster setup. You must have installed the required licenses on all nodes in the cluster.

See [“Installing a VCS license”](#) on page 185.

### To configure global cluster components at the primary site

- 1 Start the GCO Configuration wizard.  
`# gcoconfig`
- 2 The wizard discovers the NIC devices on the local system and prompts you to enter the device to be used for the global cluster. Specify the name of the device and press Enter.
- 3 If you do not have NIC resources in your configuration, the wizard asks you whether the specified NIC will be the public NIC used by all systems. Enter **y** if it is the public NIC; otherwise enter **n**. If you entered **n**, the wizard prompts you to enter the names of NICs on all systems.
- 4 Enter the virtual IP to be used for the global cluster.  
 You must use either IPv4 or IPv6 address. VCS does not support configuring clusters that use different Internet Protocol versions in a global cluster.
- 5 If you do not have IP resources in your configuration, the wizard does the following:
  - For IPv4 address:  
 The wizard prompts you for the netmask associated with the virtual IP. The wizard detects the netmask; you can accept the suggested value or enter another value.
  - For IPv6 address:  
 The wizard prompts you for the prefix associated with the virtual IP.
- 6 The wizard prompts for the values for the network hosts. Enter the values.
- 7 The wizard starts running commands to create or update the ClusterService group. Various messages indicate the status of these commands. After running these commands, the wizard brings the ClusterService group online.
- 8 Verify that the gcoip resource that monitors the virtual IP address for inter-cluster communication is online.  
`# hares -state gcoip`

### Installing and configuring VCS at the secondary site

Perform the following steps to set up a VCS cluster at the secondary site.

#### To install and configure VCS at the secondary site

- 1 At the secondary site, install and configure VCS cluster.  
 Note the following points for this task:
  - During VCS installation, enable licenses for global clusters.



- During VCS configuration, answer the prompts to configure global cluster. This step configures the virtual IP for inter-cluster communication.

See the *Veritas Cluster Server Installation Guide* for instructions.

- 2 Verify that the gcoip resource that monitors the virtual IP address for inter-cluster communication is online.

```
hares -state gcoip
```

## Securing communication between the wide-area connectors

Perform the following steps to configure secure communication between the wide-area connectors.

### To secure communication between the wide-area connectors

- 1 Verify that the Symantec Product Authentication Service (AT) is running in both the clusters.
- 2 If the clusters use different root brokers, establish trust between the clusters.

For example in a VCS global cluster environment with two clusters, perform the following steps to establish trust between the clusters:

- On each node of the first cluster, enter the following command:

```
/opt/VRTSat/bin/vssat setuptrust \
-b IP_address_of_any_node_from_the_second_cluster:2821 \
-s medium
```

The command obtains and displays the security certificate and other details of the root broker of the second cluster.

If the details are correct, enter y at the command prompt to establish trust. For example:

```
The hash of above credential is
b36a2607bf48296063068e3fc49188596aa079bb
Do you want to trust the above?(y/n) y
```

- On each node of the second cluster, enter the following command:

```
/opt/VRTSat/bin/vssat setuptrust \
-b IP_address_of_any_node_from_the_first_cluster:2821 \
-s medium
```

The command obtains and displays the security certificate and other details of the root broker of the first cluster.

If the details are correct, enter y at the command prompt to establish trust.

- 3 On each cluster, take the wac resource offline on the node where the wac resource is online. For each cluster, run the following command:

```
hares -offline wac -sys node_where_wac_is_online
```

- 4 Update the values of the StartProgram and MonitorProcesses attributes of the wac resource:

```
hares -modify wac StartProgram \
"/opt/VRTSvc/bin/wacstart -secure"
```

```
hares -modify wac MonitorProcesses \
"/opt/VRTSvc/bin/wac -secure"
```

- 5 On each cluster, bring the wac resource online. For each cluster, run the following command on any node:

```
hares -online wac -sys systemname
```

## Configuring remote cluster objects

After you set up the VCS and replication infrastructure at both sites, you must link the two clusters. You must configure remote cluster objects at each site to link the two clusters. The Remote Cluster Configuration wizard provides an easy interface to link clusters.

### To configure remote cluster objects

- ◆ Start the Remote Cluster Configuration wizard.  
 From Cluster Explorer, click **Edit > Add/Delete Remote Cluster**.  
 You must use the same IP address as the one assigned to the IP resource in the ClusterService group. Global clusters use this IP address to communicate and exchange ICMP heartbeats between the clusters.  
 See [“Adding a remote cluster”](#) on page 509.

## Configuring additional heartbeat links (optional)

You can configure additional heartbeat links to exchange ICMP heartbeats between the clusters.

### To configure an additional heartbeat between the clusters (optional)

- 1 On Cluster Explorer’s **Edit** menu, click **Configure Heartbeats**.
- 2 In the Heartbeat configuration dialog box, enter the name of the heartbeat and select the check box next to the name of the cluster.
- 3 Click the icon in the **Configure** column to open the Heartbeat Settings dialog box.
- 4 Specify the value of the Arguments attribute and various timeout and interval fields. Click **+** to add an argument value; click **-** to delete it.  
 If you specify IP addresses in the Arguments attribute, make sure the IP addresses have DNS entries.
- 5 Click **OK**.

- 6 Click **OK** in the Heartbeat configuration dialog box.  
Now, you can monitor the state of both clusters from the Java Console.

## Configuring the Steward process (optional)

In case of a two-cluster global cluster setup, you can configure a Steward to prevent potential split-brain conditions, provided the proper network infrastructure exists.

See [“The Steward process: Split-brain in two-cluster global clusters”](#) on page 472.

### To configure the Steward process for clusters not running in secure mode

- 1 Identify a system that will host the Steward process.  
Make sure both clusters can connect to the system through a ping command.
- 2 Copy the file `steward` from a node in the cluster to the Steward system. The file resides at the following path:  
`/opt/VRTSvcs/bin/`
- 3 In both clusters, set the Stewards attribute to the IP address of the system running the Steward process. For example:
 

```
cluster cluster1938 (
 UserNames = { admin = gNOgNInKOjOOmWOiNL }
 ClusterAddress = "10.182.147.19"
 Administrators = { admin }
 CredRenewFrequency = 0
 CounterInterval = 5
 Stewards = {"10.212.100.165"}
)
```
- 4 On the system designated to host the Steward, start the Steward process:
 

```
steward -start
```

### To configure the Steward process for clusters running in secure mode

- 1 Verify the prerequisites for securing Steward communication are met.  
See [“Prerequisites for clusters running in secure mode”](#) on page 476.  
To verify that the `wac` process runs in secure mode, do the following:
  - Check the value of the `wac` resource attributes:
 

```
hares -value wac StartProgram
```

 The value must be `"/opt/VRTSvcs/bin/wacstart -secure."`

```
hares -value wac MonitorProcesses
```

 The value must be `"/opt/VRTSvcs/bin/wac -secure."`
  - List the `wac` process:
 

```
ps -ef | grep wac
```

The wac process must run as “/opt/VRTSvcs/bin/wac -secure.”

- 2 Identify a system that will host the Steward process.  
 Make sure both clusters can connect to the system through a ping command.
- 3 Copy the steward file from a node in the cluster to the Steward system. The file resides at the following path:  
 /opt/VRTSvcs/bin/
- 4 Install the Symantec Product Authentication Services client on the system that is designated to run the Steward process.  
 See the Symantec Product Authentication Service documentation for instructions.
- 5 Create an account for the Steward in any authentication broker of the clusters that are part of the global cluster. All cluster nodes serve as authentication brokers when the cluster runs in secure mode.

```
vssat addprpl --pdrtype local --domain HA_SERVICES --
prplname Steward_GCO_systemname --password password --
prpltype service
```

When creating the account, make sure the following conditions are met:

- The domain name must be of the form: HA\_SERVICES
- The account name must be of the form: Steward\_GCO\_systemname
- The account type must be service and the domain type must be VX.

- 6 Note the password used to create the account.
- 7 Retrieve the broker hash for the account.

```
vssat showbrokerhash
```

- 8 Create a credential package (steward.cred) for this account. Note that the credential package will be bound to a system.

```
vssat createpkg --prplname Steward_GCO_systemname --domain
vx:HA_SERVICES@<fully_qualified_name_of_cluster_node_on_which_this_command_is_being_run> --broker systemname:2821 --
password password --hash <brokerhash_obtained_in_above_step>
--out steward.cred --host_ctx
systemname_on_which_steward_will_run
```

- 9 Copy the file steward.cred to the system designated to run the Steward process.  
 Copy the file to the directory where the steward is installed.
- 10 Execute the credential package on the system designated to run the Steward process.

```
vssat execpkg --in <path_to_credential>\steward.cred --ob --
host_ctx
```

The variable *<path\_to\_credential>* represents the directory to which you copied the steward credentials.

- 11 On the Steward system, create a file called `Steward.conf` and populate it with the following information:

```
broker=system_name
accountname=accountname
domain=HA_SERVICES@FQDN_of_system_that_issued_the_certificate
```

- 12 In both clusters, set the Stewards attribute to the IP address of the system that runs the Steward process. For example:

```
cluster cluster1938 (
 UserNames = { admin = gNOgNInKOjOOmWOiNL }
 ClusterAddress = "10.182.147.19"
 Administrators = { admin }
 CredRenewFrequency = 0
 CounterInterval = 5
 Stewards = {"10.212.100.165"}
)
```

- 13 On the system designated to run the Steward, start the Steward process:

```
steward -start -secure
```

#### To stop the Steward process

- ◆ To stop the Steward process that is not configured in secure mode, open a new command window and run the following command:  
**steward -stop**
- ◆ To stop the Steward process running in secure mode, open a new command window and run the following command:  
**steward -stop -secure**

## Configuring service groups for global cluster setup

Perform the following steps to configure the service groups for disaster recovery.

#### To configure service groups

- 1 At the secondary site, set up the application for high availability. Configure VCS service groups for the application. Create a configuration that is similar to the one in the first cluster.
  - You can do this by either using Cluster Manager (Java Console) to copy and paste resources from the primary cluster, or by copying the configuration from the `main.cf` file in the primary cluster to the secondary cluster.

- Make appropriate changes to the configuration. For example, you must modify the SystemList attribute to reflect the systems in the secondary cluster.
  - Make sure that the name of the service group (appgroup) is identical in both clusters.
- 2 To assign remote administration privileges to users for remote cluster operations, configure users with the same name and privileges on both clusters.  
See [“User privileges in global clusters”](#) on page 75.
  - 3 If your setup uses BIND DNS, add a resource of type DNS to the application service group at each site.  
Refer to the *Veritas Cluster Server Bundled Agent’s Reference Guide* for more details.
  - 4 At each site, perform the following depending on the replication technology you have set up:
    - Veritas Volume Replicator  
Configure VCS to manage and monitor VVR Replicated Volume Groups (RVGs).  
See [“Configuring VCS service group for VVR-based replication”](#) on page 486.
    - A supported third-party replication technology  
Install and configure the corresponding VCS agent for replication.  
See the Installation and Configuration Guide for the corresponding VCS replication agent for instructions.

## Configuring VCS service group for VVR-based replication

Perform the following steps to configure VCS to monitor Veritas Volume Replicator (VVR). Then, set an online local hard group dependency from application service group (appgroup) to replication service group (appgroup\_rep) to ensure that the service groups fail over and switch together.

### To create the RVG resources in VCS

- 1 Create a new service group, say appgroup\_rep.
- 2 Copy the DiskGroup resource from the appgroup to the new group.
- 3 Configure new resources of type IP and NIC in the appgroup\_rep service group. The IP resource monitors the virtual IP that VVR uses for replication.
- 4 Configure a new resource of type RVG in the new (appgroup\_rep) service group.

5 Configure the RVG resource.

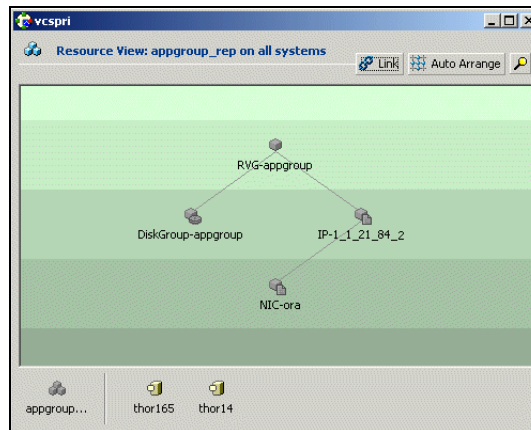
See the *Veritas Cluster Server Agents for Veritas Volume Replicator Configuration Guide* for more information about the resource.

Note that the RVG resource starts, stops, and monitors the RVG in its current state and does not promote or demote VVR when you want to change the direction of replication. That task is managed by the RVGPrimary agent.

6 Set dependencies as per the following information:

- RVG resource depends on the IP resource.
- RVG resource depends on the DiskGroup resource.
- IP resource depends on the NIC resource.

The service group now looks like:



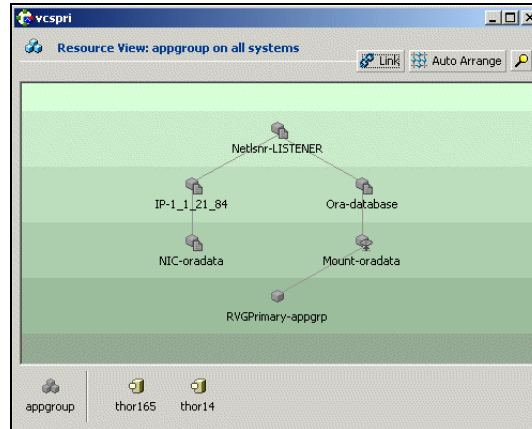
7 Delete the DiskGroup resource from the appgroup service group.

8 In the application service group (appgroup), add a resource of type RVGPrimary and configure its attributes.

See the *Veritas Cluster Server Agents for Veritas Volume Replicator Configuration Guide* for more information about the resource.

9 Set resource dependencies such that the Mount resource depends on the RVGPrimary resource.

The appgroup now looks like:



#### To link the application and replication service groups

- 1 In the Cluster Explorer configuration tree, click the cluster name.
- 2 In the view panel, click the **Service Groups** tab. This opens the service group dependency graph.
- 3 Click **Link**.
- 4 Click the parent group, appgroup, and move the mouse toward the child group, appgroup\_rep.
- 5 Click the child group appgroup\_rep.
- 6 In the Link Service Groups dialog box, click the online local relationship and the hard dependency type and click **OK**.

## Configuring global service groups for global cluster setup

Run the Global Group Configuration wizard to configure the application service group (appgroup) as a global group.

#### To create the global service group

- 1 In the service group tree of Cluster Explorer, right-click the application service group (appgroup)
- 2 Select **Configure As Global** from the menu.
- 3 Enter the details of the service group to modify (appgroup).



- 4 From the **Available Clusters** box, click the clusters on which the group can come online. The local cluster is not listed as it is implicitly defined to be part of the ClusterList. Click the right arrow to move the cluster name to the **ClusterList** box.
- 5 Select the policy for cluster failover:
  - **Manual** prevents a group from automatically failing over to another cluster.
  - **Auto** enables a group to automatically fail over to another cluster if it is unable to fail over within the cluster, or if the entire cluster faults.
  - **Connected** enables a group to automatically fail over to another cluster if it is unable to fail over within the cluster.
- 6 Click **Next**.
- 7 Enter or review the connection details for each cluster. Click the **Configure** icon to review the remote cluster information for each cluster.
- 8 Enter the IP address of the remote cluster, the IP address of a cluster system, or the host name of a cluster system.
- 9 Enter the user name and the password for the remote cluster and click **OK**.
- 10 Click **Next**.
- 11 Click **Finish**.
- 12 Save the configuration.

The appgroup service group is now a global group and can be failed over between clusters.

## When a cluster faults

In the global cluster setup, consider a case where the primary cluster suffers a failure. The Oracle service group cannot fail over in the local cluster and must fail over globally, to a node in another cluster.

In this situation, VCS sends an alert indicating that the cluster is down.

An administrator can bring the group online in the remote cluster.

The RVGPrimary agent ensures that VVR volumes are made writable and the DNS agent ensures that name services are resolved to the remote site. The application can be started at the remote site.

## Declaring the type of failure

If a disaster disables all processing power in your primary data center, heartbeats from the failover site to the primary data center fail. VCS sends an

alert signalling cluster failure. If you choose to take action on this failure, VCS prompts you to declare the type of failure.

You can choose one of the following options to declare the failure:

- *Disaster*, implying permanent loss of the primary data center
- *Outage*, implying the primary may return to its current form in some time
- *Disconnect*, implying a split-brain condition; both clusters are up, but the link between them is broken
- *Replica*, implying that data on the takeover target has been made consistent from a backup source and that the RVGPrimary can initiate a takeover when the service group is brought online. This option applies to VVR environments only.

You can select the groups to be failed over to the local cluster, in which case VCS brings the selected groups online on a node based on the group's FailOverPolicy attribute. It also marks the groups as being OFFLINE in the other cluster. If you do not select any service groups to fail over, VCS takes no action except implicitly marking the service groups as offline in the failed cluster.

## Switching the service group back to the primary

You can switch the service group back to the primary after resolving the fault at the primary site. Before switching the application to the primary site, you must resynchronize any changed data from the active Secondary site since the failover. This can be done manually through VVR or by running a VCS action from the RVGPrimary resource.

**To switch the service group when the primary site has failed and the secondary did a takeover**

- 1 In the **Service Groups** tab of the configuration tree, right-click the resource.
- 2 Click **Actions**.
- 3 Specify the details of the action:
  - From the **Action** list, choose fbsync.
  - Click the system on which to execute the action.
  - Click **OK**.

This begins a fast-failback of the replicated data set. You can monitor the value of the ResourceInfo attribute for the RVG resource to determine when the resynchronization has completed.

- 4 Once the resynchronization completes, switch the service group to the primary cluster.

- In the **Service Groups** tab of the Cluster Explorer configuration tree, right-click the service group.
- Click **Switch To**, and click **Remote switch**.
- In the Switch global group dialog box, click the cluster to switch the group. Click the specific system, or click **Any System**, and click **OK**.

## Setting up a fire drill

The Disaster Recovery Fire Drill procedure tests the fault-readiness of a configuration by mimicking a failover from the primary site to the secondary site. This procedure is done without stopping the application at the primary site and disrupting user access, interrupting the flow of replicated data, or causing the secondary to need resynchronization.

The initial steps to create a fire drill service group on the secondary site that closely follows the configuration of the original application service group and contains a point-in-time copy of the production data in the Replicated Volume Group (RVG). Bringing the fire drill service group online on the secondary site demonstrates the ability of the application service group to fail over and come online at the secondary site, should the need arise. Fire drill service groups do not interact with outside clients or with other instances of resources, so they can safely come online even when the application service group is online.

You must conduct a fire drill only at the Secondary site; do not bring the fire drill service group online on the node hosting the original application.

Set an offline local dependency between the fire drill service group and the application service group to make sure a fire drill does not block an application failover in case a disaster strikes the primary site.

VCS also supports HA fire drills to verify a resource can fail over to another node in the cluster.

See [“Testing resource failover using HA fire drills”](#) on page 263.

---

**Note:** You can conduct fire drills only on regular VxVM volumes; volume sets (vset) are not supported.

---

VCS provides hardware replication agents for array-based solutions, such as Hitachi Truecopy, EMC SRDF, and so on. If you are using hardware replication agents to monitor the replicated data clusters, refer to the VCS replication agent documentation for details on setting up and configuring fire drill.

## Creating and configuring the fire drill service group manually

You can create the fire drill service group using the command line or Cluster Manager (Java Console.) The fire drill service group uses the duplicated copy of the application data.

Creating and configuring the fire drill service group involves the following tasks:

- [“Creating the fire drill service group”](#) on page 493
- [“Linking the fire drill and replication service groups”](#) on page 493

- [“Adding resources to the fire drill service group”](#) on page 495
- [“Configuring the fire drill service group”](#) on page 495
- [“Enabling the FireDrill attribute”](#) on page 495

## Creating the fire drill service group

This section describes how to use the Cluster Manager (Java Console) to create the fire drill service group and change the failover attribute to false so that the fire drill service group does not failover to another node during a test.

### To create the fire drill service group

- 1 Open the Veritas Cluster Manager (Java Console). (**Start>All Programs>Symantec>Veritas Cluster Manager - Java Console**)
- 2 Log on to the cluster and click **OK**.
- 3 Click the **Service Group** tab in the left pane and click the **Resources** tab in the right pane.
- 4 Right-click the cluster in the left pane and click **Add Service Group**.
- 5 In the Add Service Group dialog box, provide information about the new service group.
  - In Service Group name, enter a name for the fire drill service group
  - Select systems from the Available Systems box and click the arrows to add them to the Systems for Service Group box.
  - Click **OK**.

### To disable the AutoFailOver attribute

- 1 Click the **Service Group** tab in the left pane and select the fire drill service group.
- 2 Click the **Properties** tab in the right pane.
- 3 Click the **Show all attributes** button.
- 4 Double-click the **AutoFailOver** attribute.
- 5 In the Edit Attribute dialog box, clear the **AutoFailOver** check box.
- 6 Click **OK** to close the Edit Attribute dialog box.
- 7 Click the **Save and Close Configuration** icon in the tool bar.

## Linking the fire drill and replication service groups

Create an online local firm dependency link between the fire drill service group and the replication service group.

**To link the service groups**

- 1** In Cluster Explorer, click the System tab in the left pane and click the **Service Groups** tab in the right pane.
- 2** Click **Link**.
- 3** Click the fire drill service group, drag the link and click the replication service group.
- 4** Define the dependency. Choose the **online local** and **firm** options and click **OK**.

## Adding resources to the fire drill service group

Add resources to the new fire drill service group to recreate key aspects of the application service group.

### To add resources to the service group

- 1 In Cluster Explorer, click the **Service Group** tab in the left pane, click the application service group and click the **Resources** tab in the right pane.
- 2 Right-click the resource at the top of the tree, select **Copy** and click **Self and Child Nodes**.
- 3 In the left pane, click the fire drill service group.
- 4 Right-click the right pane, and click **Paste**.
- 5 In the Name Clashes dialog box, specify a way for the resource names to be modified, for example, insert an **FD\_** prefix. Click **Apply**.
- 6 Click **OK**.

## Configuring the fire drill service group

After copying resources to the fire drill service group, edit the resources so they will work properly with the duplicated data. The attributes must be modified to reflect the configuration at the remote site. Bringing the service group online without modifying resource attributes is likely to result in a cluster fault and interruption in service.

### To configure the service group

- 1 In Cluster Explorer, click the **Service Group** tab in the left pane, click the fire drill service group in the left pane and click the **Resources** tab in the right pane.
- 2 Right-click the RVGPrimary resource and click **Delete**.
- 3 Right-click the resource to be edited and click **View>Properties View**. If a resource to be edited does not appear in the pane, click **Show All Attributes**.
- 4 Edit attributes to reflect the configuration at the remote site. For example, change the MountV resources so that they point to the volumes used in the fire drill service group. Similarly, reconfigure the Lanman and IP resources.

## Enabling the FireDrill attribute

You must edit certain resource types so they are FireDrill-enabled. Making a resource type FireDrill-enabled changes the way that VCS checks for concurrency violations. Typically, when FireDrill is not enabled, resources can not come online on more than one node in a cluster at a time. This behavior

prevents multiple nodes from using a single resource or from answering client requests. Fire drill service groups do not interact with outside clients or with other instances of resources, so they can safely come online even when the application service group is online.

Typically, you would enable the FireDrill attribute for the resource type used the configure the agent. For example, in a service group monitoring SQL Server, enable the FireDrill attribute for the SQLServer2000 and the MSSearch resource types.

#### To enable the FireDrill attribute

- 1 In Cluster Explorer, click the **Types** tab in the left pane, right-click the type to be edited, and click **View > Properties View**.
- 2 Click **Show All Attributes**.
- 3 Double click **FireDrill**.
- 4 In the Edit Attribute dialog box, enable **FireDrill** as required, and click **OK**. Repeat the process of enabling the FireDrill attribute for all required resource types.

## Configuring the fire drill service group using the wizard

Use the Fire Drill Setup Wizard to set up the fire drill configuration.

The wizard performs the following specific tasks:

- Prepares all data volumes with FMR 4.0 technology, which enables space-optimized snapshots.
- Creates a Cache object to store changed blocks during the fire drill, which minimizes disk space and disk spindles required to perform the fire drill.
- Configures a VCS service group that resembles the real application group.
- Schedules the fire drill and the notification of results.

The wizard works only with application groups that contain one disk group. The wizard sets up the first RVG in an application. If the application has more than one RVG, you must create space-optimized snapshots and configure VCS manually, using the first RVG as reference.

## Running the fire drill setup wizard

#### To run the wizard

- 1 Start the RVG Secondary Fire Drill wizard on the VVR secondary site, where the application service group is offline and the replication group is online as a secondary:



```
/opt/VRTSvcs/bin/fdsetup
```

- 2 Read the information on the Welcome screen and press the Enter key.
- 3 The wizard identifies the global service groups. Enter the name of the service group for the fire drill.
- 4 Review the list of volumes in disk group that could be used for a space-optimized snapshot. Enter the volumes to be selected for the snapshot. Typically, all volumes used by the application, whether replicated or not, should be prepared, otherwise a snapshot might not succeed. Press the Enter key when prompted.
- 5 Enter the cache size to store writes when the snapshot exists. The size of the cache must be large enough to store the expected number of changed blocks during the fire drill. However, the cache is configured to grow automatically if it fills up. Enter disks on which to create the cache. Press the Enter key when prompted.
- 6 The wizard starts running commands to create the fire drill setup. Press the Enter key when prompted.  
The wizard creates the application group with its associated resources. It also creates a fire drill group with resources for the application (Oracle, for example), the Mount, and the RVGSnapshot types.  
The application resources in both service groups define the same application, the same database in this example. The wizard sets the FireDrill attribute for the application resource to 1 to prevent the agent from reporting a concurrency violation when the actual application instance and the fire drill service group are online at the same time.

## Configuring local attributes in the fire drill service group

The fire drill setup wizard does not recognize localized attribute values for resources. If the application service group has resources with local (per-system) attribute values, you must manually set these attributes after running the wizard.

## Verifying a successful fire drill

Set an offline local dependency between the fire drill service group and the application service group to make sure a fire drill does not block an application failover in case a disaster strikes the primary site.

Bring the fire drill service group online on a node that does not have the application running. Verify that the fire drill service group comes online.

This action validates that your disaster recovery solution is configured correctly and the production service group will fail over to the secondary site in the event of an actual failure (disaster) at the primary site.

If the fire drill service group does not come online, review the VCS engine log to troubleshoot the issues so that corrective action can be taken as necessary in the production service group.

You can also view the fire drill log, located at `/tmp/fd-servicegroup.pid`

Remember to take the fire drill offline once its functioning has been validated. Failing to take the fire drill offline could cause failures in your environment. For example, if the application service group were to fail over to the node hosting the fire drill service group, there would be resource conflicts, resulting in both service groups faulting.

## Scheduling a fire drill

You can schedule the fire drill for the service group using the `fdsched` script. The `fdsched` script is designed to run only on the lowest numbered node that is currently running in the cluster. The scheduler runs the command `hagrp -online firedrill_group -any` at periodic intervals.

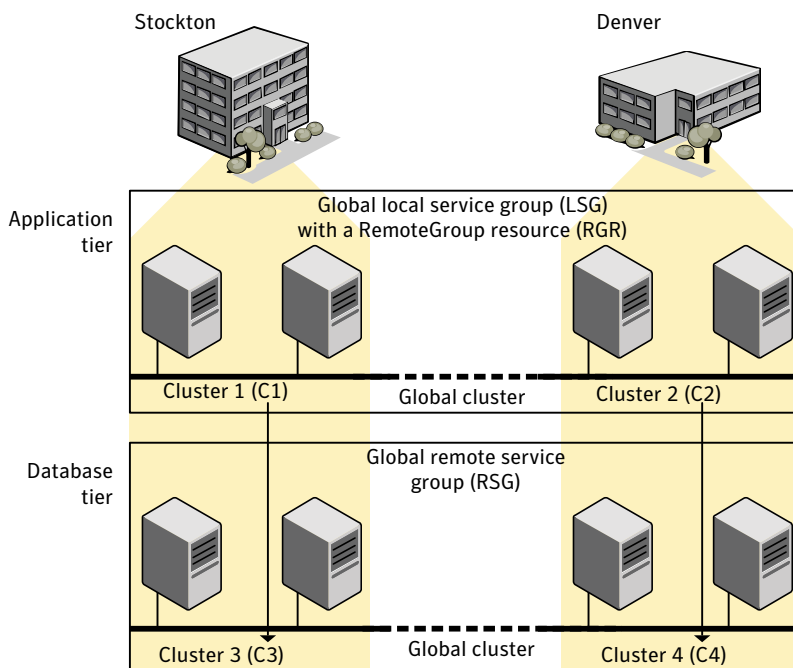
### To schedule a fire drill

- 1 Add the file `/opt/VRTSvcs/bin/fdsched` to your crontab.
- 2 To make fire drills highly available, add the `fdsched` file to each node in the cluster.

## Multi-tiered application support using the RemoteGroup agent in a global environment

Figure 15-3 represents a two-site, two-tier environment. The application cluster, which is globally clustered between two sites, has cluster dependencies up and down the tiers. Cluster 1 (C1), depends on the remote service group for cluster 3 (C3). At the same time, cluster 2 (C2) also depends on the remote service group for cluster 4 (C4).

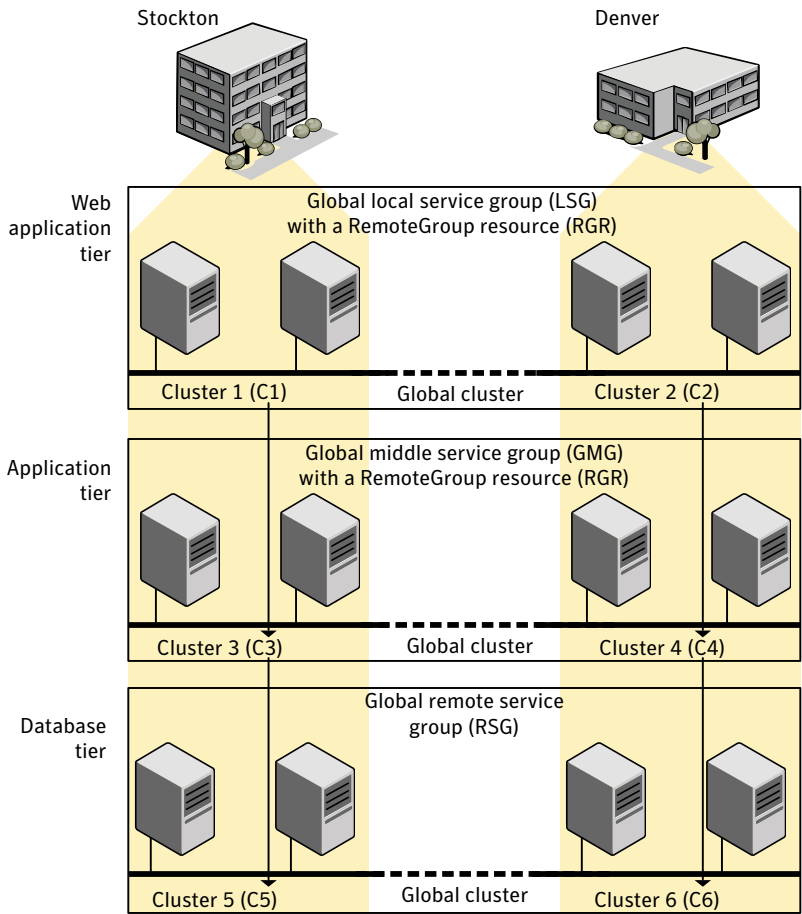
**Figure 15-3** A VCS two-tiered globally clustered application and database



Just as a two-tier, two-site environment is possible, you can also tie a three-tier environment together.

Figure 15-4 represents a two-site, three-tier environment. The application cluster is globally clustered between two sites. The cluster has cluster dependencies up and down the tiers. Cluster 1 (C1), depends on the RemoteGroup resource on the DB tier for cluster 3 (C3), and then on the remote service group for cluster 5 (C5). The stack for C2, C4, and C6 functions the same.

Figure 15-4      A three-tiered globally clustered application, database, and storage



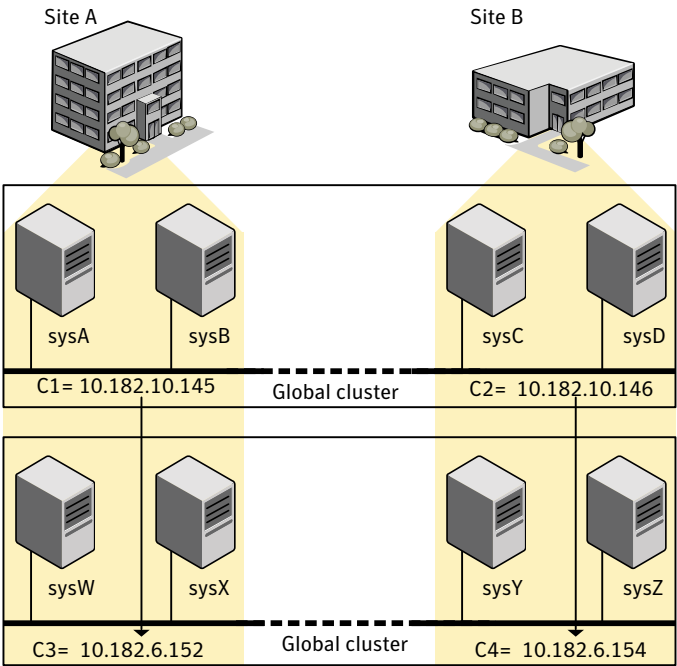
# Test scenario for a multi-tiered environment

In the following scenario, eight systems reside in four clusters. Each tier contains a global cluster. The global local service group in the top tier depends on the global remote service group in the bottom tier.

The following main.cf files show this multi-tiered environment. The FileOnOff resource is used to test the dependencies between layers. Note that some attributes have been edited for clarity, and that these clusters are not running in secure mode.

Figure 15-5 shows the scenario for testing.

Figure 15-5 A VCS two-tiered globally clustered scenario



## The main.cf file for cluster 1

The contents of the main.cf file for cluster 1 (C1) in the top tier, containing the sysA and sysB nodes.

```
include "types.cf"

cluster C1 (
 ClusterAddress = "10.182.10.145"
)

remotecluster C2 (
 ClusterAddress = "10.182.10.146"
)

heartbeat Icmp (
 ClusterList = { C2 }
 AYATimeout = 30
 Arguments @C2 = { "10.182.10.146" }
)

system sysA (
)

system sysB (
)

group LSG (
 SystemList = { sysA = 0, sysB = 1 }
 ClusterList = { C2 = 0, C1 = 1 }
 AutoStartList = { sysA, sysB }
 ClusterFailOverPolicy = Auto
)

FileOnOff filec1 (
 PathName = "/tmp/c1"
)

RemoteGroup RGR (
 IPAddress = "10.182.6.152"
 // The above IPAddress is the highly available address of C3—
 // the same address that the wac uses
 Username = root
 Password = xxxyyy
 GroupName = RSG
 VCSSysName = ANY
 ControlMode = OnOff
)
```

## The main.cf file for cluster 2

The contents of the main.cf file for cluster 2 (C2) in the top tier, containing the sysC and sysD nodes.

```
include "types.cf"

cluster C2 (
 ClusterAddress = "10.182.10.146"
)

remoteclass C1 (
 ClusterAddress = "10.182.10.145"
)

heartbeat Icmp (
 ClusterList = { C1 }
 AYATimeout = 30
 Arguments @C1 = { "10.182.10.145" }
)

system sysC (
)

system sysD (
)

group LSG (
 SystemList = { sysC = 0, sysD = 1 }
 ClusterList = { C2 = 0, C1 = 1 }
 Authority = 1
 AutoStartList = { sysC, sysD }
 ClusterFailOverPolicy = Auto
)

FileOnOff filec2 (
 PathName = filec2
)

RemoteGroup RGR (
 IPAddress = "10.182.6.154"
 // The above IPAddress is the highly available address of C4—
 // the same address that the wac uses
 Username = root
 Password = vvvyyy
 GroupName = RSG
 VCSSysName = ANY
 ControlMode = OnOff
)
```

## The main.cf file for cluster 3

The contents of the main.cf file for cluster 3 (C3) in the bottom tier, containing the sysW and sysX nodes.

```
include "types.cf"

cluster C3 (
 ClusterAddress = "10.182.6.152"
)

remoteclass C4 (
 ClusterAddress = "10.182.6.154"
)

heartbeat Icmp (
 ClusterList = { C4 }
 AYATimeout = 30
 Arguments @C4 = { "10.182.6.154" }
)

system sysW (
)

system sysX (
)

group RSG (
 SystemList = { sysW = 0, sysX = 1 }
 ClusterList = { C3 = 1, C4 = 0 }
 AutoStartList = { sysW, sysX }
 ClusterFailOverPolicy = Auto
)

FileOnOff filec3 (
 PathName = "/tmp/filec3"
)
```

## The main.cf file for cluster 4

The contents of the main.cf file for cluster 4 (C4) in the bottom tier, containing the sysY and sysZ nodes.

```
include "types.cf"

cluster C4 (
 ClusterAddress = "10.182.6.154"
)

remoteclass C3 (
 ClusterAddress = "10.182.6.152"
)
```



```
heartbeat Icmp (
 ClusterList = { C3 }
 AYATimeout = 30
 Arguments @C3 = { "10.182.6.152" }
)

system sysY (
)

system sysZ (
)

group RSG (
 SystemList = { sysY = 0, sysZ = 1 }
 ClusterList = { C3 = 1, C4 = 0 }
 Authority = 1
 AutoStartList = { sysY, sysZ }
 ClusterFailOverPolicy = Auto
)

FileOnOff filec4 (
 PathName = "/tmp/filec4"
)
```



# Administering global clusters from Cluster Manager (Java console)

- [About global clusters](#)
- [Adding a remote cluster](#)
- [Deleting a remote cluster](#)
- [Administering global service groups](#)
- [Administering global heartbeats](#)

## About global clusters

The process of creating a global cluster environment involves creating a common service group for specified clusters, making sure all the service groups are capable of being brought online in the specified clusters, connecting the standalone clusters, and converting the service group that is common to all the clusters to a global service group. Use the console to add and delete remote clusters, create global service groups, and manage cluster heartbeats.

Creating a global cluster environment requires the following conditions:

- All service groups are properly configured and able to come online.
- The service group that will serve as the global group has the same unique name across all applicable clusters.
- The clusters must use the same version of VCS.
- The clusters must use the same operating system.
- The clusters are standalone and do not already belong to a global cluster environment.

Through the Java Console, you can simulate the process of generating and clearing global cluster faults in an OFFLINE state. Use VCS Simulator to complete these operations.

See “[Predicting VCS behavior using VCS Simulator](#)” on page 265.

For remote cluster operations, you must configure a VCS user with the same name and privileges in each cluster.

See “[User privileges in global clusters](#)” on page 75.

## Adding a remote cluster

Cluster Explorer provides a wizard to create global clusters by linking standalone clusters. Command Center only enables you to perform remote cluster operations on the local cluster.

- If you are creating a global cluster environment for the first time with two standalone clusters, run the wizard from either of the clusters.
- If you are adding a standalone cluster to an existing global cluster environment, run the wizard from a cluster already in the global cluster environment.

The following information is required for the Remote Cluster Configuration Wizard in Cluster Explorer:

- The active host name or IP address of each cluster in the global configuration and of the cluster being added to the configuration.
- The user name and password of the administrator for each cluster in the configuration.
- The user name and password of the administrator for the cluster being added to the configuration.

---

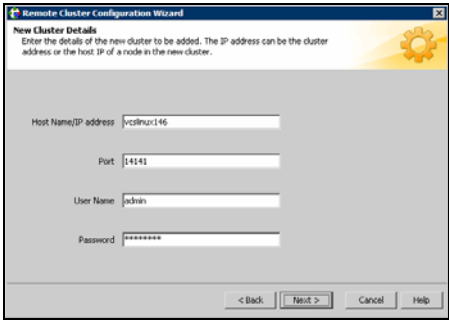
**Note:** Symantec does not support adding a cluster that is already part of a global cluster environment. To merge the clusters of one global cluster environment (for example, cluster A and cluster B) with the clusters of another global environment (for example, cluster C and cluster D), separate cluster C and cluster D into standalone clusters and add them one by one to the environment containing cluster A and cluster B.

---

### To add a remote cluster to a global cluster environment in Cluster Explorer

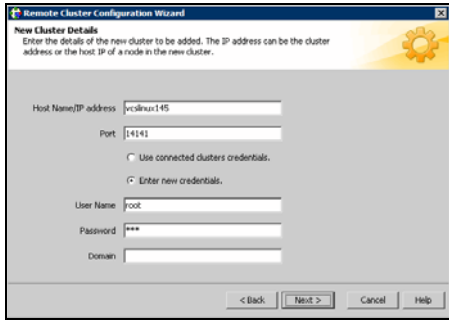
- 1 From Cluster Explorer, click **Add/Delete Remote Cluster** on the **Edit** menu.  
*or*  
From the Cluster Explorer configuration tree, right-click the cluster name, and click **Add/Delete Remote Clusters**.
- 2 Review the required information for the **Remote Cluster Configuration Wizard** and click **Next**.
- 3 In the Wizard Options dialog box:
  - Click **Add Cluster**.
  - Click **Next**.
- 4 Enter the details of the new cluster:

If the cluster is not running in secure mode:



- Enter the host name of a cluster system, an IP address of a cluster system, or the IP address of the cluster that will join the global environment.
- Verify the port number.
- Enter the user name and the password.
- Click **Next**.

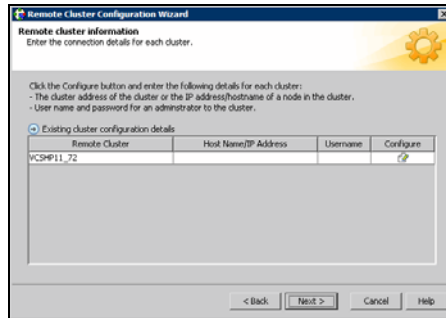
If the cluster is running in secure mode:



- Enter the host name of a cluster system, an IP address of a cluster system, or the IP address of the cluster that will join the global environment.
- Verify the port number.
- Choose to connect to the remote cluster with the credentials used for the current cluster connection or enter new credentials, including the user name, password, and the domain.  
If you have connected to the remote cluster using the wizard earlier, you can use the credentials from the previous connection.

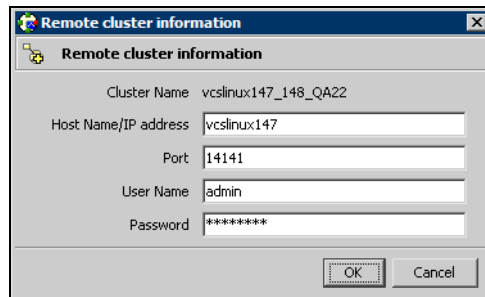
Click **Next**.

- 5 Enter the details of the existing remote clusters; this information on administrator rights enables the wizard to connect to all the clusters and make changes to the configuration:



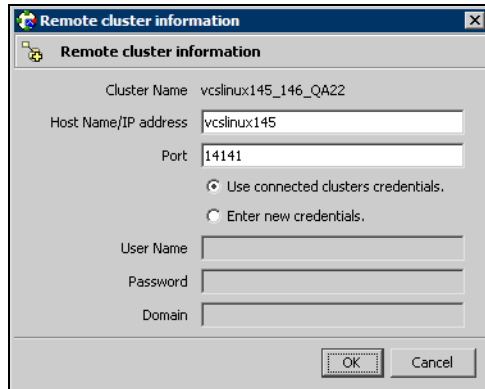
- 6 Click the **Configure** icon. The Remote cluster information dialog box is displayed.

**If the cluster is not running in secure mode:**



- Enter the host name of a cluster system, an IP address of a cluster system, or the IP address of the cluster that will join the global environment.
- Verify the port number.
- Enter the user name.
- Enter the password.
- Click **OK**.
- Repeat these steps for each cluster in the global environment.

**If the cluster is running in secure mode:**

A screenshot of a Java dialog box titled "Remote cluster information". The dialog has a standard Windows-style title bar with a close button. Inside, there's a tabbed interface with the first tab selected. The form contains several fields: "Cluster Name" with the value "vcslinux145\_146\_QA22", "Host Name/IP address" with "vcslinux145", "Port" with "14141", and three radio buttons for authentication: "Use connected clusters credentials." (selected), "Enter new credentials.", and "Enter new credentials." (disabled). Below these are fields for "User Name", "Password", and "Domain". At the bottom right are "OK" and "Cancel" buttons.

- Enter the host name of a cluster system, an IP address of a cluster system, or the IP address of the cluster that will join the global environment.
- Verify the port number.
- Choose to connect to the remote cluster with the credentials used for the current cluster connection or enter new credentials, including the user name, password, and the domain.
- Click **OK**.

7 Click **Next**.

8 Click **Finish**. After running the wizard, the configurations on all the relevant clusters are opened and changed; the wizard does not close the configurations.

**To add a remote cluster to a global cluster environment in Command Center**

---

**Note:** Command Center enables you to perform operations on the local cluster; this does not affect the overall global cluster configuration.

---

- 1 Click **Commands>Configuration>Cluster Objects>Add Remote Cluster**.
- 2 Enter the name of the cluster.
- 3 Enter the IP address of the cluster.
- 4 Click **Apply**.



## Deleting a remote cluster

The Remote Cluster Configuration Wizard enables you to delete a remote cluster. This operation involves the following tasks:

- Taking the `ApplicationProcess` resource configured to monitor the `wac` resource offline on the cluster that will be removed from the global environment. For example, to delete cluster C2 from a global environment containing C1 and C2, log on to C2 and take the `wac` resource offline.
- Removing the name of the specified cluster (C2) from the cluster lists of the other global groups using the Global Group Configuration Wizard. Note that the Remote Cluster Configuration Wizard in Cluster Explorer updates the cluster lists for heartbeats. Log on to the local cluster (C1) to complete this task before using the Global Group Configuration Wizard.
- Deleting the cluster (C2) from the local cluster (C1) through the Remote Cluster Configuration Wizard.

---

**Note:** You cannot delete a remote cluster if the cluster is part of a cluster list for global service groups or global heartbeats, or if the cluster is in the `RUNNING`, `BUILD`, `INQUIRY`, `EXITING`, or `TRANSITIONING` states.

---

### To take the `wac` resource offline

- 1 From Cluster Monitor, log on to the cluster that will be deleted from the global cluster environment.
- 2 In the **Service Groups** tab of the Cluster Explorer configuration tree, right-click the **wac** resource under the **Application** type in the **ClusterService** group.  
*or*  
Click the **ClusterService** group in the configuration tree, click the **Resources** tab, and right-click the resource in the view panel.
- 3 Click **Offline**, and click the appropriate system from the menu.

### To remove a cluster from a cluster list for a global group

- 1 From Cluster Explorer, click **Configure Global Groups** on the **Edit** menu.
- 2 Click **Next**.
- 3 Enter the details of the service group to modify:
  - Click the name of the service group.
  - For global to local cluster conversion, click the left arrow to move the cluster name from the cluster list back to the **Available Clusters** box.

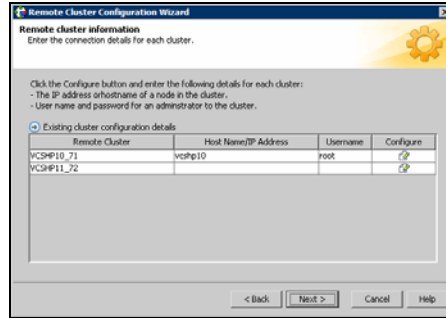
**Deleting a remote cluster**

- Click **Next**.
- 4 Enter or review the connection details for each cluster. Click the **Configure** icon to review the remote cluster information for each cluster.  
**If the cluster is not running in secure mode:**
  - Enter the IP address of the remote cluster, the IP address of a cluster system, or the host name of a cluster system.
  - Verify the port number.
  - Enter the user name.
  - Enter the password.
  - Click **OK**.**If the cluster is running in secure mode:**
  - Enter the IP address of the remote cluster, the IP address of a cluster system, or the host name of a cluster system.
  - Verify the port number.
  - Choose to connect to the remote cluster using the connected cluster's credentials or enter new credentials, including the user name, password, and the domain.
  - Click **OK**.
- 5 Click **Next**.
- 6 Click **Finish**.

**To delete a remote cluster from the local cluster**

- 1 From Cluster Explorer, click **Add/Delete Remote Cluster** on the **Edit** menu.  
*or*  
 From the Cluster Explorer configuration tree, right-click the cluster name, and click **Add/Delete Remote Clusters**.
- 2 Review the required information for the **Remote Cluster Configuration Wizard** and click **Next**.
- 3 In the Wizard Options dialog box:
  - Click **Delete Cluster**.
  - Click **Next**.
- 4 In the Delete Cluster dialog box:
  - Click the name of the remote cluster to delete.
  - Click **Next**.

- 5 Review the connection details for each cluster. Click the **Configure** icon to review the remote cluster information for each cluster.



#### If the cluster is not running in secure mode:

- Enter the IP address of the remote cluster, the IP address of a cluster system, or the host name of a cluster system.
- Verify the port number.
- Enter the user name.
- Enter the password.
- Click **OK**.

#### If the cluster is running in secure mode:

- Enter the IP address of the remote cluster, the IP address of a cluster system, or the host name of a cluster system.
- Verify the port number.
- Choose to connect to the remote cluster with the credentials used for the current cluster connection or enter new credentials, including the user name, password, and the domain.  
If you have connected to the remote cluster using the wizard earlier, you can use the credentials from the previous connection.
- Click **OK**.

- 6 Click **Finish**.

# Administering global service groups

After connecting clusters in a global cluster environment, use the Global Group Configuration Wizard to convert a local service group that is common to the global clusters to a global group. This wizard also enables you to convert global groups into local groups.

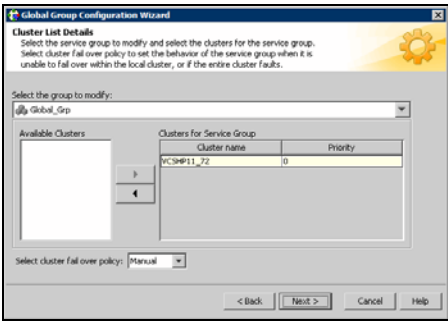
Administering global groups requires the following conditions:

- A group that will serve as the global group must have the same name across all applicable clusters.
- You must know the user name and password for the administrator for each cluster in the configuration.

Use Cluster Explorer to bring a global group online and take a global group offline on a remote cluster.

## Converting local and global groups

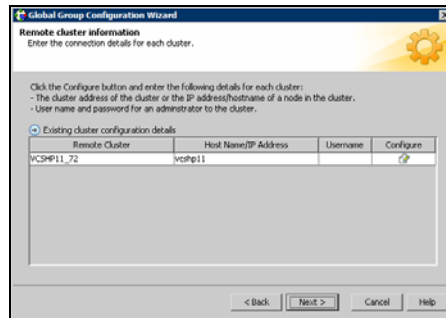
- 1 From Cluster Explorer, click **Configure Global Groups...** on the **Edit** menu.  
*or*  
From the Cluster Explorer configuration tree, right-click the service group, click **Configure As Global...** or **Make Local...** and proceed to step 3b.
- 2 Review the information required for the Global Group Configuration Wizard and click **Next**.
- 3 Enter the details of the service group to modify:



- Click the name of the service group that will be converted from a local group to a global group, or vice versa.
- From the **Available Clusters** box, click the clusters on which the group can come online. Click the right arrow to move the cluster name to the **Clusters for Service Group** box; for global to local cluster conversion,

click the left arrow to move the cluster name back to the **Available Clusters** box. A priority number (starting with 0) indicates the cluster in which the group will attempt to come online. If necessary, double-click the entry in the **Priority** column to enter a new value.

- Select the policy for cluster failover:
    - **Manual** prevents a group from automatically failing over to another cluster.
    - **Auto** enables a group to automatically fail over to another cluster if it is unable to fail over within the cluster, or if the entire cluster faults.
    - **Connected** enables a group to automatically fail over to another cluster if it is unable to fail over within the cluster.
  - Click **Next**.
- 4 Enter or review the connection details for each cluster:



Click the **Configure** icon to review the remote cluster information for each cluster.

**If the cluster is not running in secure mode:**

- Enter the IP address of the remote cluster, the IP address of a cluster system, or the host name of a cluster system.
- Verify the port number.
- Enter the user name and password.
- Click **OK**.

Repeat these steps for each cluster in the global environment.

**If the cluster is running in secure mode:**

- Enter the IP address of the remote cluster, the IP address of a cluster system, or the host name of a cluster system.
- Verify the port number.

**Administering global service groups**

- Choose to connect to the remote cluster with the credentials used for the current cluster connection, or enter new credentials, including the user name, password, and the domain.  
If you have connected to the remote cluster using the wizard earlier, you can use the credentials from the previous connection.
- Click **OK**.

Repeat these steps for each cluster in the global environment.

- 5 In the Remote cluster information dialog box, click **Next**.
- 6 Click **Finish**.

## Bringing a service group online in a remote cluster

- 1 In the **Service Groups** tab of the Cluster Explorer configuration tree of a local cluster, right-click the service group.  
*or*  
Click a local cluster in the configuration tree, click the **Service Groups** tab, and right-click the service group icon in the view panel.
- 2 Click **Online**, and click **Remote online...**
- 3 In the Online global group dialog box:
  - Click the remote cluster to bring the group online.
  - Click the specific system, or click **Any System**, to bring the group online.
  - Click **OK**.
- 4 In the Question dialog box, click **Yes**.

## Taking a service group offline in a remote cluster

- 1 In the **Service Groups** tab of the Cluster Explorer configuration tree of a local cluster, right-click the service group.  
*or*  
Click a local cluster in the configuration tree, click the **Service Groups** tab, and right-click the service group icon in the view panel.
- 2 Click **Offline**, and click **Remote offline...**
- 3 In the Offline global group dialog box:
  - Click the remote cluster to take the group offline.
  - Click the specific system, or click **All Systems**, to take the group offline.
  - Click **OK**.
- 4 In the Question dialog box, click **Yes**.

## Switching a service group to a remote cluster

- 1 In the **Service Groups** tab of the Cluster Explorer configuration tree of a local cluster, right-click the service group.  
*or*  
Click a local cluster in the configuration tree, click the **Service Groups** tab, and right-click the service group icon in the view panel.
- 2 Click **Switch To**, and click **Remote switch...**
- 3 In the Switch global group dialog box:
  - Click the cluster to switch the group.
  - Click the specific system, or click **Any System**, to switch the group.  
If you specify a system to switch the group and if the PreSwitch attribute value is set to 1, the VCS engine invokes the PreSwitch actions for the resources that support the action. If you want to skip these actions, you must temporarily set the PreSwitch attribute value to 0. See [“Service group attributes”](#) on page 663.
  - Click **OK**.
- 4 In the Question dialog box, click **Yes**.

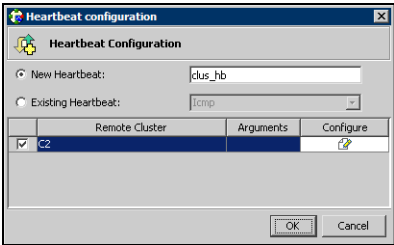
# Administering global heartbeats

Use Cluster Explorer to add, modify, and delete heartbeats in a global cluster environment. *Icmp* heartbeats send Icmp packets simultaneously to all IP addresses; *IcmpS* heartbeats send individual Icmp packets to IP addresses in serial order. Global clustering requires a minimum of one heartbeat between clusters; the Icmp heartbeat is added when the cluster is added to the environment. You can add additional heartbeats as a precautionary measure.

## Adding a global heartbeat

To add a cluster heartbeat from Cluster Explorer

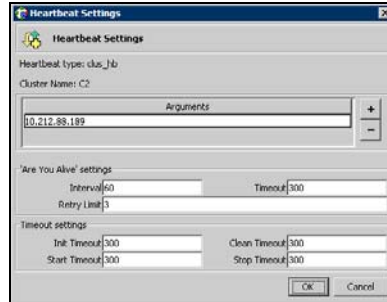
- 1 Click **Configure Heartbeats** on the **Edit** menu.
- 2 In the Heartbeat Configuration dialog box:



- Enter the name of the heartbeat.
- Select the check box next to the name of the cluster to add it to the cluster list for the heartbeat.
- Click the icon in the **Configure** column to open the Heartbeat Settings dialog box.



- Specify the value of the Arguments attribute and various timeout and interval fields. Click + to add an argument value; click - to delete it.



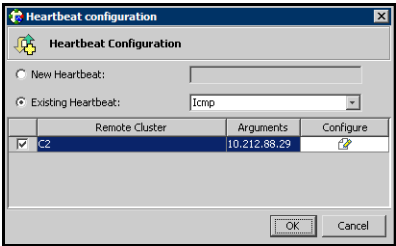
- Click **OK**.
- Click **OK** on the Heartbeat configuration dialog box.

#### To add a cluster heartbeat from Command Center

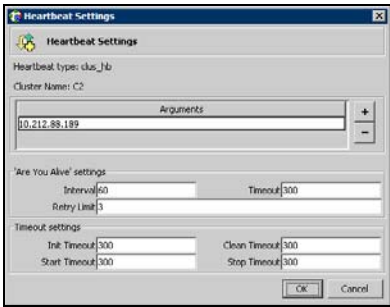
- 1 Click **Commands>Configuration>Cluster Objects>Add Heartbeat**.
- 2 Enter the name of the heartbeat.
- 3 Click **Apply**.

## Modifying a global heartbeat

- 1 From Cluster Explorer, click **Configure Heartbeats** on the **Edit** menu.
- 2 In the Heartbeat Configuration dialog box:



- Click **Existing Heartbeat**.
- Click the name of the existing heartbeat from the menu.
- Select or clear the check box next to the name of a cluster to add or remove it from the cluster list for the heartbeat.
- If necessary, click the icon in the **Configure** column to open the Heartbeat Settings dialog box. Otherwise, proceed to the last step.
- Change the values of the Arguments attribute and various timeout and interval fields. Click + to add an argument value; click - to delete it.



- Click **OK**.
- Click **OK** on the Heartbeat Configuration dialog box.

## Deleting a global heartbeat

You cannot delete the last heartbeat between global clusters.

**To delete a cluster heartbeat from Command Center**

- 1 Click **Commands>Configuration>Cluster Objects>Delete Heartbeat**.
- 2 Click the heartbeat to delete.
- 3 Click **Apply**.



# Administering global clusters from the command line

- [About administering global clusters from the command line](#)
- [Global querying](#)
- [Administering global service groups](#)
- [Administering resources](#)
- [Administering clusters in global clusters](#)
- [Administering heartbeats](#)

# About administering global clusters from the command line

For remote cluster operations, you must configure a VCS user with the same name and privileges in each cluster.

See “[User privileges in global clusters](#)” on page 75.

## Global querying

VCS enables you to query global cluster objects, including service groups, resources, systems, resource types, agents, and clusters. You may enter query commands from any system in the cluster. Commands to display information on the global cluster configuration or system states can be executed by all users; you do not need root privileges. Only global service groups may be queried.

### Querying global cluster service groups

To display service group attribute values across clusters

```
hagrp -value service_group attribute [system] [-clus cluster |
-localclus]
```

The option `-clus` displays the attribute value on the cluster designated by the variable *cluster*; the option `-localclus` specifies the local cluster.

If the attribute has local scope, you must specify the system name, except when querying the attribute on the system from which you run the command.

To display the state of a service group across clusters

```
hagrp -state [service_groups -sys systems] [-clus cluster |
-localclus]
```

The option `-clus` displays the state of all service groups on a cluster designated by the variable *cluster*; the option `-localclus` specifies the local cluster.

To display service group information across clusters

```
hagrp -display [service_groups] [-attribute attributes]
[-sys systems] [-clus cluster | -localclus]
```

The option `-clus` applies to global groups only. If the group is local, the cluster name must be the local cluster name, otherwise no information is displayed.

**To display service groups in a cluster**

```
hagrp -list [conditionals] [-clus cluster | -localclus]
```

The option **-clus** lists all service groups on the cluster designated by the variable *cluster*; the option **-localclus** specifies the local cluster.

**To display usage for the service group command**

```
hagrp [-help [-modify|-link|-list]]
```

**To display the resources for a service group in a cluster**

- ◆ Type the following command:

```
hagrp -resources service_group [-clus cluster_name | -localclus]
```

The option **-clus** displays information for the cluster designated by the variable *cluster\_name*; the option **-localclus** specifies the local cluster.

## Querying resources

### To display resource attribute values across clusters

```
hares -value resource attribute [system] [-clus cluster |
-localclus]
```

The option `-clus` displays the attribute value on the cluster designated by the variable *cluster*; the option `-localclus` specifies the local cluster.

If the attribute has local scope, you must specify the system name, except when querying the attribute on the system from which you run the command.

### To display the state of a resource across clusters

```
hares -state [resource -sys system] [-clus cluster |
-localclus]
```

The option `-clus` displays the state of all resources on the specified cluster; the option `-localclus` specifies the local cluster. Specifying a system displays resource state on a particular system.

### To display resource information across clusters

```
hares -display [resources] [-attribute attributes] [-group
service_groups] [-type types] [-sys systems] [-clus cluster |
-localclus]
```

The option `-clus` lists all service groups on the cluster designated by the variable *cluster*; the option `-localclus` specifies the local cluster.

### For a list of resources across clusters

```
hares -list [conditionals] [-clus cluster | -localclus]
```

The option `-clus` lists all resources that meet the specified conditions in global service groups on a cluster as designated by the variable *cluster*.

### To display usage for the resource command

```
hares -help [-modify | -list]
```



## Querying systems

### To display system attribute values across clusters

```
hasys -value system attribute [-clus cluster | -localclus]
```

The option `-clus` displays the values of a system attribute in the cluster as designated by the variable *cluster*; the option `-localclus` specifies the local cluster.

### To display the state of a system across clusters

```
hasys -state [system] [-clus cluster | -localclus]
```

Displays the current state of the specified system. The option `-clus` displays the state in a cluster designated by the variable *cluster*; the option `-localclus` specifies the local cluster. If you do not specify a system, the command displays the states of all systems.

### For information about each system across clusters

```
hasys -display [systems] [-attribute attributes] [-clus cluster | -localclus]
```

The option `-clus` displays the attribute values on systems (if specified) in a cluster designated by the variable *cluster*; the option `-localclus` specifies the local cluster.

### For a list of systems across clusters

```
hasys -list [conditionals] [-clus cluster | -localclus]
```

Displays a list of systems whose values match the given conditional statements. The option `-clus` displays the systems in a cluster designated by the variable *cluster*; the option `-localclus` specifies the local cluster.

## Querying clusters

### For the value of a specific cluster attribute on a specific cluster

```
haclus -value attribute [cluster] [-localclus]
```

The attribute must be specified in this command. If you do not specify the cluster name, the command displays the attribute value on the local cluster.

### To display the state of a local or remote cluster

```
haclus -state [cluster] [-localclus]
```

The variable *cluster* represents the cluster. If a cluster is not specified, the state of the local cluster and the state of all remote cluster objects as seen by the local cluster are displayed.

### For information on the state of a local or remote cluster

```
haclus -display [cluster] [-localclus]
```

If a cluster is not specified, information on the local cluster is displayed.

### For a list of local and remote clusters

```
haclus -list [conditionals]
```

Lists the clusters that meet the specified conditions, beginning with the local cluster.

### To display usage for the cluster command

```
haclus [-help [-modify]]
```

### To display the status of a faulted cluster

```
haclus -status cluster
```

Displays the status on the specified faulted cluster. If no cluster is specified, the command displays the status on all faulted clusters. It lists the service groups that were not in the OFFLINE or the FAULTED state before the fault occurred. It also suggests corrective action for the listed clusters and service groups.

## Querying status

### For the status of local and remote clusters

```
hastatus
```

## Querying heartbeats

The `hahb` command is used to manage WAN heartbeats that emanate from the local cluster. Administrators can monitor the “health of the remote cluster via heartbeat commands and mechanisms such as Internet, satellites, or storage replication technologies. Heartbeat commands are applicable only on the cluster from which they are issued.

---

**Note:** You must have Cluster Administrator privileges to add, delete, and modify heartbeats.

---

The following commands are issued from the command line.

### For a list of heartbeats configured on the local cluster

```
hahb -list [conditionals]
```

The variable *conditionals* represents the conditions that must be met for the heartbeat to be listed.

### To display information on heartbeats configured in the local cluster

```
hahb -display [heartbeat ...]
```

If *heartbeat* is not specified, information regarding all heartbeats configured on the local cluster is displayed.

### To display the state of the heartbeats in remote clusters

```
hahb -state [heartbeat] [-clus cluster]
```

For example, to get the state of heartbeat ICMP from the local cluster to the remote cluster phoenix:

```
hahb -state ICMP -clus phoenix
```

**To display an attribute value of a configured heartbeat**

```
hahb -value heartbeat attribute [-clus cluster]
```

The `-value` option provides the value of a single attribute for a specific heartbeat. The cluster name must be specified for cluster-specific attribute values, but not for global.

For example, to display the value of the ClusterList attribute for heartbeat ICMP:

```
hahb -value Icmp ClusterList
```

Note that ClusterList is a global attribute.

**To display usage for the command `hahb`**

```
hahb [-help [-modify]]
```

If the `-modify` option is specified, the usage for the `hahb -modify` option is displayed.

# Administering global service groups

Operations for the VCS global clusters option are enabled or restricted depending on the permissions with which you log on. The privileges associated with each user role are enforced for cross-cluster, service group operations.

See “[User privileges in global clusters](#)” on page 75.

## To bring a service group online across clusters for the first time

```
hagrp -online -force
```

## To bring a service group online across clusters

```
hagrp -online service_group -sys system [-clus cluster |
-localclus]
```

The option `-clus` brings the service group online on the system designated in the cluster. If a system is not specified, the service group is brought online on any node within the cluster. The option `-localclus` brings the service group online in the local cluster.

## To bring a service group online on any node

```
hagrp -online [-force] service_group -any [-clus cluster |
-localclus]
```

The option `-any` specifies that HAD brings a failover group online on the optimal system, based on the requirements of service group workload management and existing group dependencies. If bringing a parallel group online, HAD brings the group online on each system designated in the SystemList attribute.

## To take a service group offline across clusters

```
hagrp -offline [-force] [-ifprobed] service_group -sys system
[-clus cluster | -localclus]
```

The option `-clus` takes offline the service group on the system designated in the cluster.

## To take a service group offline anywhere

```
hagrp -offline [-ifprobed] service_group -any
[-clus cluster | -localclus]
```

The option `-any` specifies that HAD takes a failover group offline on the system on which it is online. For a parallel group, HAD takes the group offline on each system on which the group is online. HAD adheres to the existing group dependencies when taking groups offline.

#### To switch a service group across clusters

```
hagrp -switch service_group -to system [-clus cluster |
-localclus [-nopre]]
```

The option `-clus` identifies the cluster to which the service group will be switched. The service group is brought online on the system specified by the `-to system` argument. If a system is not specified, the service group may be switched to any node within the specified cluster.

The option `-nopre` indicates that the VCS engine must switch the service group regardless of the value of the PreSwitch service group attribute.

See [“Service group attributes”](#) on page 663.

#### To switch a service group anywhere

```
hagrp -switch service_group -any [-clus cluster | -localclus]
```

The `-any` option specifies that the VCS engine switches a service group to the best possible system on which it is currently not online, based on the value of the group's FailOverPolicy attribute. The VCS engine switches a global service group from a system to another system in the local cluster or a remote cluster.

If you do not specify the `-clus` option, the VCS engine by default assumes `-localclus` option and selects an available system within the local cluster.

The option `-clus` identifies the remote cluster to which the service group will be switched. The VCS engine then selects the target system on which to switch the service group.

#### To switch a parallel global service group across clusters

```
hagrp -switch service_group -any -clus remote_cluster
```

VCS brings the parallel service group online on all possible nodes in the remote cluster.

# Administering resources

## To take action on a resource across clusters

```
hares -action resource token [-actionargs arg1 ...]
[-sys system] [-clus cluster | -localclus]
```

The option `-clus` implies resources on the cluster. If the designated system is not part of the local cluster, an error is displayed. If the `-sys` option is not used, it implies resources on the local node.

## To invoke the Info function across clusters

```
hares -refreshinfo resource [-sys system] [-clus cluster |
-localclus]
```

Causes the Info function to update the value of the ResourceInfo resource level attribute for the specified resource if the resource is online. If no system or remote cluster is specified, the Info function runs on local system(s) where the resource is online.

## To display usage for the resource command

To display usage for the command `hares` and its various options:

```
hares [-help [-modify | -list]]
```

## Administering clusters in global clusters

### To add a remote cluster object

```
haclus -add cluster ip
```

The variable *cluster* represents the cluster. This command does not apply to the local cluster.

### To delete a remote cluster object

```
haclus -delete cluster
```

The variable *cluster* represents the cluster.

### To modify an attribute of a local or remote cluster object

```
haclus -modify attribute value [-clus cluster]...
```

The variable *cluster* represents the cluster.

### To declare the state of a cluster after a disaster

```
haclus -declare disconnnet/outage/disaster/replica -clus cluster
[-failover]
```

The variable *cluster* represents the remote cluster.

### To manage cluster alerts

- ◆ Run the `haalert` command to manage cluster alerts.

|                        |                                                                                     |
|------------------------|-------------------------------------------------------------------------------------|
| <b>haalert -testfd</b> | Generates a simulated "cluster fault" alert that is sent to the VCS engine and GUI. |
|------------------------|-------------------------------------------------------------------------------------|

|                         |                                                                 |
|-------------------------|-----------------------------------------------------------------|
| <b>haalert -display</b> | For each alert, the command displays the following information: |
|-------------------------|-----------------------------------------------------------------|

- alert ID
- time when alert occurred
- cluster on which alert occurred
- object name for which alert occurred
- (cluster name, group name, and so on).
- informative message about alert

|                      |                                                                 |
|----------------------|-----------------------------------------------------------------|
| <b>haalert -list</b> | For each alert, the command displays the following information: |
|----------------------|-----------------------------------------------------------------|

- time when alert occurred
- alert ID



|                        |                                                                                        |
|------------------------|----------------------------------------------------------------------------------------|
| <b>haalert -delete</b> | Delete a specific alert by. You must enter a text message                              |
| <b>alert_id -notes</b> | within quotes describing the reason for deleting the alert.                            |
| <b>"description"</b>   | The comment is written to the engine log as well as sent to any connected GUI clients. |
| <b>haalert -help</b>   | Displays the usage text                                                                |

## Changing the cluster name

This section describes how to change the ClusterName in a global cluster configuration. The instructions describe how to rename VCSPriCluster to VCSPriCluster2 in a two-cluster configuration, comprising clusters VCSPriCluster and VCSSecCluster configured with the global group AppGroup.

Before changing the cluster name, make sure the cluster is not part of any ClusterList, in the wide-area Heartbeat agent and in global service groups.

### To change the name of a cluster

- 1 Run the following commands from cluster VCSPriCluster:

```
hagrp -offline ClusterService -any
hagrp -modify AppGroup ClusterList -delete VCSPriCluster
haclus -modify ClusterName VCSPriCluster2
hagrp -modify AppGroup ClusterList -add VCSPriCluster2 0
```
- 2 Run the following commands from cluster VCSSecCluster:

```
hagrp -offline ClusterService -any
hagrp -modify appgrp ClusterList -delete VCSPriCluster
hahb -modify Icmp ClusterList -delete VCSPriCluster
haclus -delete VCSPriCluster
haclus -add VCSPriCluster2 your_ip_address
hahb -modify Icmp ClusterList -add VCSPriCluster2
hahb -modify Icmp Arguments your_ip_address -clus VCSPriCluster2
hagrp -modify AppGroup ClusterList -add VCSPriCluster2 0
hagrp -online ClusterService -any
```
- 3 Run the following command from the cluster renamed to VCSPriCluster2:

```
hagrp -online ClusterService -any
```

## Administering heartbeats

### To create a heartbeat

```
hahb -add heartbeat
```

For example, type the following command to add a new IcmpS heartbeat. This represents a heartbeat sent from the local cluster and immediately forks off the specified agent process on the local cluster.

```
hahb -add IcmpS
```

### To modify a heartbeat

```
hahb -modify heartbeat attribute value ... [-clus cluster]
```

If the attribute is local, that is, it has a separate value for each remote cluster in the ClusterList attribute, the option `-clus cluster` must be specified. Use `-delete -keys` to clear the value of any list attributes. For example, type the following command to modify the ClusterList attribute and specify targets “phoenix” and “houston” for the newly created heartbeat:

```
hahb -modify ICMP ClusterList phoenix houston
```

To modify the Arguments attribute for target phoenix:

```
hahb -modify ICMP Arguments phoenix.example.com -clus phoenix
```

### To delete a heartbeat

```
hahb -delete heartbeat
```

### To change the scope of an attribute to cluster-specific

```
hahb -local heartbeat attribute
```

For example, type the following command to change the scope of the attribute AYAInterval from global to cluster-specific:

```
hahb -local ICMP AYAInterval
```

### To change the scope of an attribute to global

```
hahb -global heartbeat attribute value ...
| key ... | key value ...
```

For example, type the following command to change the scope of the attribute AYAInterval from cluster-specific to cluster-generic:

```
hahb -global ICMP AYAInterval 60
```

# Setting up replicated data clusters

- [About replicated data clusters](#)
- [How VCS replicated data clusters work](#)
- [Setting up a replicated data cluster configuration](#)
- [Migrating a service group](#)
- [Setting up a fire drill](#)

## About replicated data clusters

The Replicated Data Cluster (RDC) configuration provides both local high availability and disaster recovery functionality in a single VCS cluster.

You can set up RDC in a VCS environment using Veritas Volume Replicator (VVR.)

A Replicated Data Cluster (RDC) uses data replication to assure data access to nodes. An RDC exists within a single VCS cluster. In an RDC configuration, if an application or a system fails, the application is failed over to another system within the current primary site. If the entire primary site fails, the application is migrated to a system in the remote secondary site (which then becomes the new primary).

For VVR replication to occur, the disk groups containing the Replicated Volume Group (RVG) must be imported at the primary and secondary sites. The replication service group must be online at both sites simultaneously, and must be configured as a hybrid VCS service group.

The application service group is configured as a failover service group. The application service group must be configured with an *online local hard* dependency on the replication service group.

---

**Note:** VVR supports multiple replication secondary targets for any given primary. However, RDC for VCS supports only one replication secondary for a primary.

---

An RDC configuration is appropriate in situations where dual dedicated LLT links are available between the primary site and the disaster recovery secondary site but lacks shared storage or SAN interconnect between the primary and secondary data centers. In an RDC, data replication technology is employed to provide node access to data in a remote site.

---

**Note:** You must use dual dedicated LLT links between the replicated nodes.

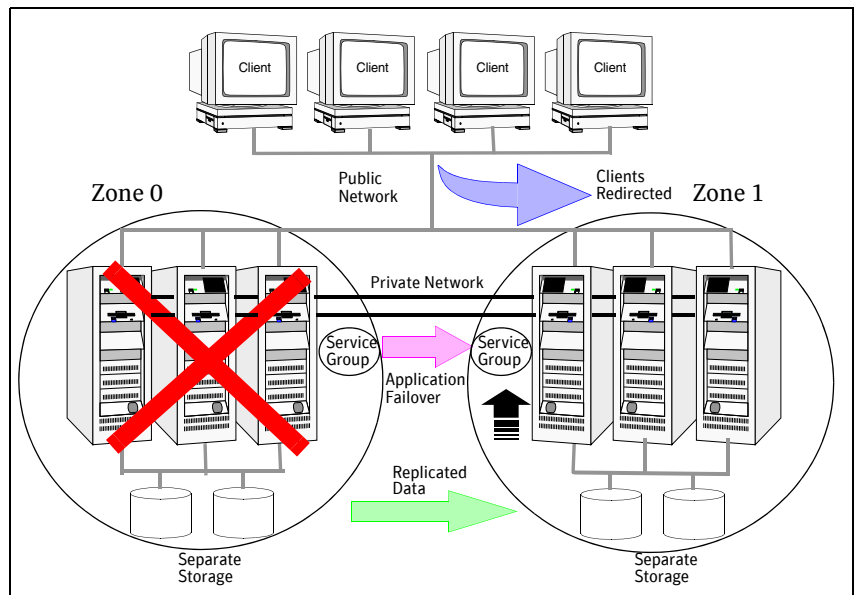
---

## How VCS replicated data clusters work

To understand how a replicated data cluster configuration works, let us take the example of an application configured in a VCS replicated data cluster. The configuration has two system zones:

- Primary zone (zone 0) comprising nodes located at the primary site and attached to the primary storage
- Secondary zone (zone 1) comprising nodes located at the secondary site and attached to the secondary storage

The application is installed and configured on all nodes in the cluster. Application data is located on shared disks within each RDC zone and is replicated across RDC zones to ensure data concurrency. The application service group is online on a system in the current primary zone and is configured to fail over in the cluster.



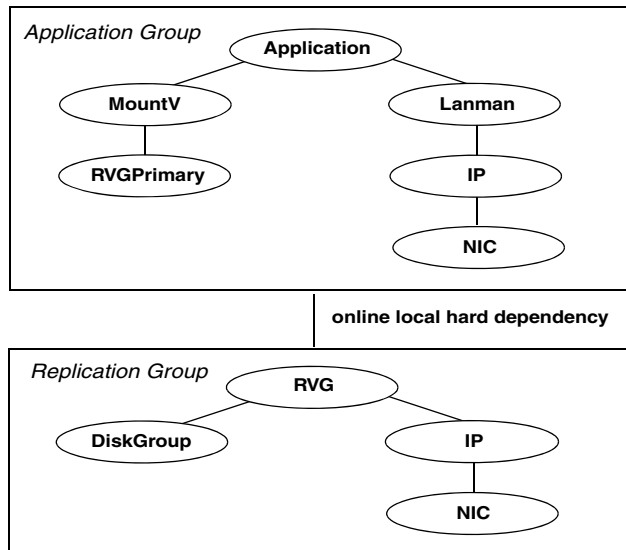
In the event of a system or application failure, VCS attempts to fail over the application service group to another system within the same RDC zone. However, in the event that VCS fails to find a failover target node within the primary RDC zone, VCS switches the service group to a node in the current secondary RDC zone (zone 1). VCS also redirects clients once the application is online on the new location.

## Setting up a replicated data cluster configuration

This section describes the steps for planning, configuring, testing, and using the VCS RDC configuration to provide a robust and easy-to-manage disaster recovery protection for your applications. It describes an example of converting a single instance Oracle database configured for local high availability in a VCS cluster to a disaster-protected RDC infrastructure. The solution uses Veritas Volume Replicator to replicate changed data.

### Typical replicated data cluster configuration

The following illustration depicts a typical RDC configuration:



In this example, a single-instance Oracle database is configured as a VCS service group (oragroup) on a four-node cluster, with two nodes in the primary RDC system zone and two in the secondary RDC system zone. In the event of a failure on the primary node, VCS fails over Oracle to the second node in the primary zone.

The process involves the following steps:

- [Setting Up Replication](#)
- [Configuring the Service Groups](#)
- [Configuring the Service Group Dependencies](#)

## Setting up replication

Veritas Volume Replicator (VVR) technology is a license-enabled feature of Veritas Volume Manager (VxVM), so you can convert VxVM-managed volumes into replicated volumes managed using VVR. In this example, the process involves grouping the Oracle data volumes into a Replicated Volume Group (RVG), and creating the VVR Secondary on hosts in another VCS cluster, located in your DR site.

When setting up VVR, it is a best practice to use the same DiskGroup and RVG name on both sites. If the volume names are the same on both zones, the Mount resources will mount the same block devices, and the same Oracle instance will start on the secondary in case of a failover.

## Configuring the service groups

### To configure the replication group

- 1 Create a hybrid service group (oragrp\_rep) for replication. You can use the VvrRvgGroup template to create the service group.  
See “[Types of service groups](#)” on page 26.
- 2 Copy the DiskGroup resource from the application to the new group.  
Configure the resource to point to the disk group that contains the RVG.
- 3 Configure new resources of type IP and NIC.
- 4 Configure a new resource of type RVG in the service group.
- 5 Configure the RVG resource. See the VVR documentation for more information.

Note that the RVG resource starts, stops, and monitors the RVG in its current state and does not promote or demote VVR when you want to change the direction of replication. The RVGPrimary agent manages that task.

- 6 Set resource dependencies as per the following information:
  - RVG resource depends on the IP resource
  - RVG resource depends on the DiskGroup resource
  - IP resource depends on the NIC resource
- 7 Set the SystemZones attribute of the child group, oragrp\_rep, such that all nodes in the primary RDC zone are in system zone 0 and all nodes in the secondary RDC zone are in system zone 1.

**To configure the application service group**

- 1 In the original Oracle service group (oragroup), delete the DiskGroup resource.
- 2 Add an RVGPrimary resource and configure its attributes.  
Set the value of the RvgResourceName attribute to the name of the RVG type resource that will be promoted and demoted by the RVGPrimary agent.  
Set the AutoTakeover and AutoResync attributes from their defaults as desired.  
See [“VCS agents to manage wide-area failover”](#) on page 471..
- 3 Set resource dependencies such that all Mount resources depend on the RVGPrimary resource. If there are a lot of Mount resources, you can set the TypeDependencies attribute for the group to denote that the Mount resource type depends on the RVGPrimary resource type.
- 4 Set the SystemZones attribute of the Oracle service group such that all nodes in the primary RDC zone are in system zone 0 and all nodes in the secondary RDC zone are in zone 1. The SystemZones attribute of both the parent and the child group must be identical.
- 5 If your setup uses BIND DNS, add a resource of type DNS to the oragroup service group. Set the Hostname attribute to the canonical name of the host or virtual IP address that the application uses on that cluster. This ensures DNS updates to the site when the group is brought online. A DNS resource would be necessary only if the nodes in the primary and the secondary RDC zones are in different IP subnets.



## Configuring the service group dependencies

Set an *online local hard* group dependency from application service group to the replication service group to ensure that the service groups fail over and switch together.

- 1 In the Cluster Explorer configuration tree, select the cluster name.
- 2 In the view panel, click the **Service Groups** tab. This opens the service group dependency graph.
- 3 Click **Link**.
- 4 Click the parent group oragroup and move the mouse toward the child group, oragroup\_rep.
- 5 Click the child group oragroup\_rep.
- 6 On the Link Service Groups dialog box, click the online local relationship and the hard dependency type and click **OK**.

## Migrating a service group

In the RDC set up for the Oracle database, consider a case where the primary RDC zone suffers a total failure of the shared storage. In this situation, none of the nodes in the primary zone see any device.

The Oracle service group cannot fail over locally within the primary RDC zone, because the shared volumes cannot be mounted on any node. So, the service group must fail over, to a node in the current secondary RDC zone.

The RVGPrimary agent ensures that VVR volumes are made writable and the DNS agent ensures that name services are resolved to the DR site. The application can be started at the DR site and run there until the problem with the local storage is corrected.

If the storage problem is corrected, you can switch the application to the primary site using VCS.

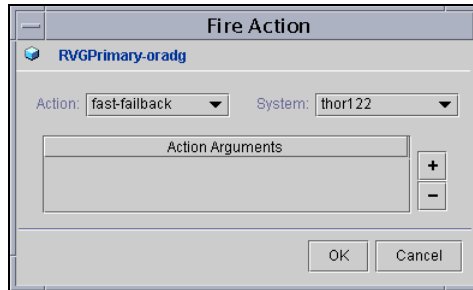
## Switching the service group

Before switching the application back to the original primary RDC zone, you must resynchronize any changed data from the active DR site since the failover. This can be done manually through VVR or by running a VCS action from the RVGPrimary resource.

### To switch the service group

- 1 In the **Service Groups** tab of the configuration tree, right-click the resource.

- 2 Click **Actions**.
- 3 Specify the details of the action:



- From the **Action** list, choose fast-failback.
- Click the system on which to execute the action.
- Click **OK**.

This begins a fast-failback of the replicated data set. You can monitor the value of the ResourceInfo attribute for the RVG resource to determine when the resynchronization has completed.

- 4 Once the resynchronization completes, switch the service group to the primary cluster. In the **Service Groups** tab of the Cluster Explorer configuration tree, right-click the service group.
- 5 Click **Switch To** and select the system in the primary RDC zone to switch to and click OK.

## Setting up a fire drill

You can use fire drills to test the configuration's fault readiness by mimicking a failover without stopping the application in the primary data center.

See “[Setting up a fire drill](#)” on page 492.

# Setting up campus clusters

- [About campus cluster configuration](#)
- [VCS campus cluster requirements](#)
- [Typical VCS campus cluster setup](#)
- [How VCS campus clusters work](#)
- [Setting up a campus cluster configuration](#)
- [About fire drill in campus clusters](#)
- [About the DiskGroupSnap agent](#)
- [Running a fire drill in a campus cluster](#)

## About campus cluster configuration

The campus cluster configuration provides local high availability and disaster recovery functionality in a single VCS cluster. This configuration uses data mirroring to duplicate data at different sites. There is no host or array replication involved.

VCS supports campus clusters that employ disk groups mirrored with Veritas Volume Manager.

## VCS campus cluster requirements

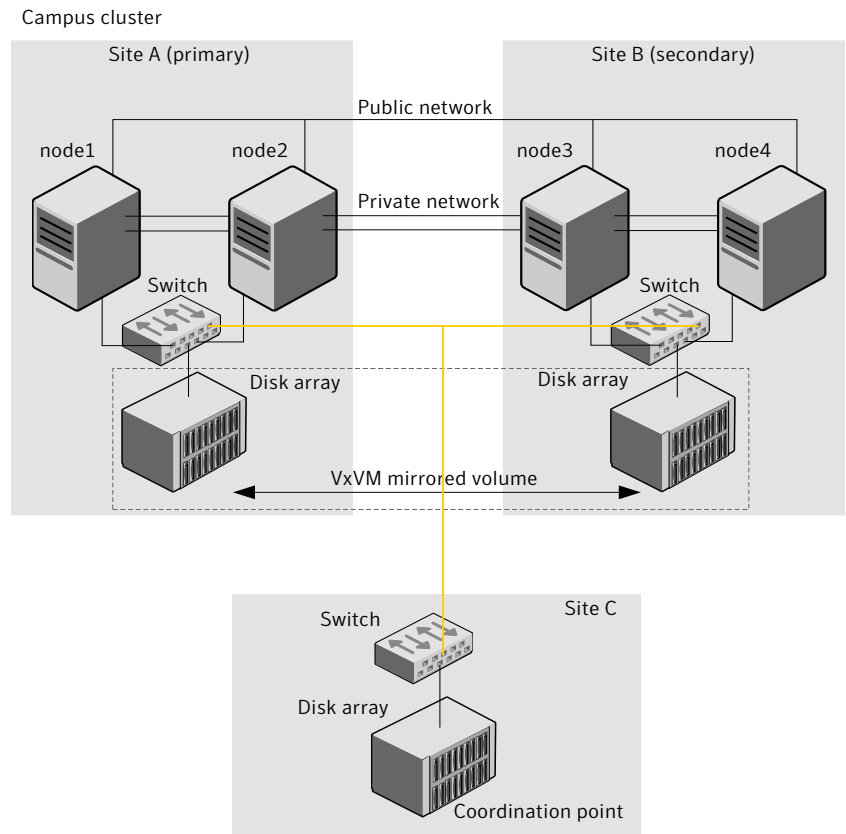
Review the following requirements for VCS campus clusters:

- You must install VCS.  
You must enable the HA/DR license if you want to manually control a service group failover across sites or system zones.
- You must have a single VCS cluster with at least one node in each of the two sites, where the sites are separated by a physical distance of no more than 80 kilometers.
- The storage and networks must have redundant-loop access between each node and each storage array.  
Symantec recommends the following in a campus cluster setup:
  - A common cross-site physical infrastructure for storage and LLT private networks.
  - Technologies such as Dense Wavelength Division Multiplexing (DWDM) for network and I/O traffic across sites to minimize the impact of network failure.
- Symantec recommends that you configure I/O fencing to prevent data corruption in the event of link failures.
- You must install Veritas Volume Manager 5.0 MP3 or later with the FMR license and the Site awareness license.
- You must configure storage to meet site-based allocation and site-consistency requirements for VxVM.
  - All the disks must be tagged with the appropriate VxVM site names.
  - The VxVM site names of both the sites in the campus cluster must be added to the diskgroups.
  - The allsites attribute for each volume in the diskgroup must be set to on. (By default, the value is set to on.)
  - The siteconsistent attribute for the diskgroups must be set to on.

# Typical VCS campus cluster setup

Figure 19-1 depicts a typical VCS campus cluster setup.

Figure 19-1 Typical VCS campus cluster setup



VCS campus cluster typically has the following characteristics:

- Single VCS cluster spans multiple sites.  
In the sample figure, VCS is configured on four nodes: node 1 and node 2 are located at site A and node 3 and node 4 at site B.
- I/O fencing is configured with one coordination point from each site of the campus cluster and another coordination point from a third site.

Figure 19-1 illustrates a typical setup with disk-based I/O fencing. You can also configure server-based I/O fencing.

See [“About I/O fencing in campus clusters”](#) on page 554.

- The shared data is located on mirrored volumes on a disk group configured using Veritas Volume Manager.
- The volumes that are required for the application have mirrors on both the sites.
- All nodes in the cluster are tagged with the VxVM site name. All disks that belong to a site are tagged with the corresponding VxVM site name.
- The disk group is configured in VCS as a resource of type DiskGroup and is mounted using the Mount resource type.

## How VCS campus clusters work

This section describes how VCS works with VxVM to provide high availability in a campus cluster environment.

In a campus cluster setup, VxVM automatically mirrors volumes across sites. To enhance read performance, VxVM reads from the plexes at the local site where the application is running. VxVM writes to plexes at both the sites.

In the event of a storage failure at a site, VxVM detaches all the disks at the failed site from the diskgroup to maintain data consistency. When the failed storage comes back online, VxVM automatically reattaches the site to the diskgroup and recovers the plexes.

See *Veritas Volume Manager Administrator's Guide* for more information.

When service group or system faults occur, VCS fails over the service groups or the nodes based on the values you set for the service group attributes SystemZones and AutoFailOver.

See [“Service group attributes”](#) on page 663.

For campus cluster setup, you must define the SystemZones attribute in such a way that the nodes at each site are grouped together. Depending on the value of the AutoFailOver attribute, VCS failover behavior is as follows:

- 0 VCS does not fail over the service group or the node.
- 1 VCS fails over the service group to another suitable node. VCS chooses to fail over the service group within the same site before choosing a node in the other site.

By default, the AutoFailOver attribute value is set to 1.

- 2
- VCS fails over the service group if another suitable node exists in the same site. Otherwise, VCS waits for administrator intervention to initiate the service group failover to a suitable node in the other site.
- This configuration requires the HA/DR license enabled.
- Symantec recommends that you set the value of AutoFailOver attribute to 2.

Sample definition for these service group attributes in the VCS main.cf is as follows:

```
group oragroup1 (
 SystemList = { node1=0, node2=1, node3=2, node4=3 }
 SystemZones = { node1=0, node2=0, node3=1, node4=1 }
 AutoFailOver = 2
 ...
)
```

Table 19-1 lists the possible failure scenarios and how VCS campus cluster recovers from these failures.

Table 19-1 Failure scenarios in campus cluster

| Failure             | Description and recovery                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Node failure        | <div><div>■ A node in a site fails.<br/>If the value of the AutoFailOver attribute is set to 1, VCS fails over the Oracle service group to another system within the same site that is defined in the SystemZones attribute.</div><div>■ All nodes in a site fail.<br/>If the value of the AutoFailOver attribute is set to 1, VCS fails over the Oracle service group to a system in the other site that is defined in the SystemZones attribute.<br/>If the value of the AutoFailOver attribute is set to 2, VCS requires administrator intervention to initiate the Oracle service group failover to a system in the other site.</div><div>If the value of the AutoFailOver attribute is set to 0, VCS requires administrator intervention to initiate a fail over in both the cases of node failure.</div></div> |
| Application failure | The behavior is similar to the node failure.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |

Table 19-1 Failure scenarios in campus cluster

| Failure                                             | Description and recovery                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Storage failure - one or more disks at a site fails | <p>VCS does not fail over the service group when such a storage failure occurs.</p> <p>VxVM detaches the site from the diskgroup if any volume in that diskgroup does not have at least one valid plex at the site where the disks failed.</p> <p>VxVM does not detach the site from the diskgroup in the following cases:</p> <ul style="list-style-type: none"><li>■ None of the plexes are configured on the failed disks.</li><li>■ Some of the plexes are configured on the failed disks, and at least one plex for a volume survives at each site.</li></ul> <p>If only some of the disks that failed come online and if the vxrelocd daemon is running, VxVM relocates the remaining failed disks to any available disks. Then, VxVM automatically reattaches the site to the diskgroup and resynchronizes the plexes to recover the volumes.</p> <p>If all the disks that failed come online, VxVM automatically reattaches the site to the diskgroup and resynchronizes the plexes to recover the volumes.</p> |
| Storage failure - all disks at both sites fail      | <p>VCS acts based on the DiskGroup agent's PanicSystemOnDGLoss attribute value.</p> <p>See <i>Veritas Bundled Agents Reference Guide</i> for more information.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| Site failure                                        | <p>All nodes and storage at a site fail.</p> <p>Depending on the value of the AutoFailOver attribute, VCS fails over the Oracle service group as follows:</p> <ul style="list-style-type: none"><li>■ If the value is set to 1, VCS fails over the Oracle service group to a system in the other site that is defined in the SystemZones attribute.</li><li>■ If the value is set to 2, VCS requires administrator intervention to initiate the Oracle service group failover to a system in the other site.</li></ul> <p>Because the storage at the failed site is inaccessible, VCS imports the disk group in the application service group with all devices at the failed site marked as NODEVICE.</p> <p>When the storage at the failed site comes online, VxVM automatically reattaches the site to the diskgroup and resynchronizes the plexes to recover the volumes.</p>                                                                                                                                        |



**Table 19-1** Failure scenarios in campus cluster

| Failure                                                | Description and recovery                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|--------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Network failure (LLT interconnect failure)             | <p>Nodes at each site lose connectivity to the nodes at the other site</p> <p>The failure of private interconnects between the nodes can result in split brain scenario and cause data corruption.</p> <p>Review the details on other possible causes of split brain and how I/O fencing protects shared data from corruption.</p> <p>See <a href="#">“About data protection”</a> on page 307.</p> <p>Symantec recommends that you configure I/O fencing to prevent data corruption in campus clusters.</p> <p>See <a href="#">“About I/O fencing in campus clusters”</a> on page 554.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Network failure (LLT and storage interconnect failure) | <p>Nodes at each site lose connectivity to the storage and the nodes at the other site</p> <p>Symantec recommends that you configure I/O fencing to prevent split brain and serial split brain conditions.</p> <ul style="list-style-type: none"> <li>■ If I/O fencing is configured:<br/> The site that loses the race commits suicide.<br/> See <a href="#">“About I/O fencing in campus clusters”</a> on page 554.<br/> When you restore the network connectivity, VxVM detects the storage at the failed site, reattaches the site to the diskgroup, and resynchronizes the plexes to recover the volumes.</li> <li>■ If I/O fencing is not configured:<br/> If the application service group was online at site A during such failure, the application service group remains online at the same site. Because the storage is inaccessible, VxVM detaches the disks at the failed site from the diskgroup. At site B where the application service group is offline, VCS brings the application service group online and imports the disk group with all devices at site A marked as NODEVICE. So, the application service group is online at both the sites and each site uses the local storage. This causes inconsistent data copies and leads to a site-wide split brain.<br/> When you restore the network connectivity between sites, a serial split brain may exist.<br/> See <i>Veritas Volume Manager Administrator’s Guide</i> for details to recover from a serial split brain condition.</li> </ul> |

## About I/O fencing in campus clusters

You must configure I/O fencing to prevent corruption of shared data in the event of a network partition.

See [“About membership arbitration”](#) on page 289.

See [“About data protection”](#) on page 307.

You can use coordinator disks, coordination point server (CP server), or a mix of both as coordination points to configure I/O fencing.

See the *Veritas Cluster Server Installation Guide* for more details to configure I/O fencing.

In a campus cluster setup, you can configure I/O fencing as follows:

- Two coordination points at one site and one coordination point at the other site  
In this case, the site that has two coordination points has a higher probability to win the race. The disadvantage with this configuration is that if the site that has two coordination points encounters a site failure, then the other site also commits suicide. With this configuration, I/O fencing cannot distinguish between an inaccessible coordination point and a failed preempt operation.
- One coordination point in each of the two sites and a third coordination point at a third site  
This configuration ensures that fencing works even if one of the sites becomes unavailable. A coordinator disk in a third site allows at least a sub-cluster to continue operations in the event of a site failure in a campus cluster. The site that can access the coordinator disk in the third site in addition to its local coordinator disk wins the race. However, if both the sites of the campus cluster are unable to access the coordination point at the third site, each site gains one vote and the nodes at both the sites commit suicide.

## Setting up a campus cluster configuration

You must perform the following tasks to set up a campus cluster:

- [Preparing to set up a campus cluster configuration](#)
- [Configuring I/O fencing to prevent data corruption](#)
- [Configuring VxVM diskgroups for campus cluster configuration](#)
- [Configuring VCS service group for campus clusters](#)

## Preparing to set up a campus cluster configuration

Before you set up the configuration, review the VCS campus cluster requirements.

See [“VCS campus cluster requirements”](#) on page 548.

### To prepare to set up a campus cluster configuration

- 1 Set up the physical infrastructure.
  - Set up access to the local storage arrays and to remote storage arrays on each node.
  - Set up private heartbeat network.
 See [“Typical VCS campus cluster setup”](#) on page 549.
- 2 Install VCS on each node to form a cluster with at least one node in each of the two sites.  
 See *Veritas Cluster Server Installation Guide* for instructions.
- 3 Install VxVM on each node with the required licenses.  
 See *Veritas Storage Foundation Installation Guide* for instructions.

## Configuring I/O fencing to prevent data corruption

Perform the following tasks to configure I/O fencing to prevent data corruption in the event of a communication failure.

See [“About I/O fencing in campus clusters”](#) on page 554.

See *Veritas Cluster Server Installation Guide* for more details.

### To configure I/O fencing to prevent data corruption

- 1 Set up the storage at a third site.  
 You can extend the DWDM to the third site to have FC SAN connectivity to the storage at the third site. You can also use iSCSI targets as the coordinator disks at the third site.
- 2 Set up I/O fencing.

## Configuring VxVM diskgroups for campus cluster configuration

Follow the procedure to configure VxVM diskgroups for remote mirroring.

See *Veritas Volume Manager Administrator’s Guide* for more information on the VxVM commands.

### To configure VxVM diskgroups for campus cluster configuration

- 1 Set the site name for each host:

```
vxctl set site=sitename
```

The site name is stored in the `/etc/vx/volboot` file. Use the following command to display the site names:

```
vxctl list | grep siteid
```

- 2 Set the site name for all the disks in an enclosure:

```
vxdisk settag site=sitename encl:enclosure
```

To tag specific disks, use the following command:

```
vxdisk settag site=sitename disk
```

- 3 Verify that the disks are registered to a site.

```
vxdisk listtag
```

- 4 Create a diskgroup with disks from both the sites.

```
vxvg init diskgroup siteA_disk1 siteB_disk2
```

- 5 Configure site-based allocation on the diskgroup that you created for each site that is registered to the disk group.

```
vxvg -g diskgroup addsite sitename
```

- 6 Configure site consistency on the diskgroup.

```
vxvg -g diskgroup set siteconsistent=on
```

- 7 Create one or more mirrored volumes in the disk group.

```
vxassist -g diskgroup make volume size
```

With the Site Awareness license installed on all hosts, the volume that you create has the following characteristics by default:

- The `allsites` attribute is set to `on`; the volumes have at least one plex at each site.
- The volumes are automatically mirrored across sites.
- The read policy `rdpol` is set to `siteread`.
- The volumes inherit the site consistency value that is set on the diskgroup.

## Configuring VCS service group for campus clusters

Follow the procedure to configure the diskgroups under VCS control and set up the VCS attributes to define failover in campus clusters.

### To configure VCS service groups for campus clusters

- 1 Create a VCS service group (`app_sg`) for the application that runs in the campus cluster.

```
hagrp -add app_sg
```

```
hagrp -modify app_sg SystemList node1 0 node2 1 node3 2 node4 3
```

- 2 Set up the system zones. Configure the `SystemZones` attribute for the service group.

- ```
hagrp -modify app_sg SystemZones node1 0 node2 0 node3 1 node4 1
```
- 3 Set up the group fail over policy. Set the value of the AutoFailOver attribute for the service group.

```
hagrp -modify app_sg AutoFailOver 2
```
 - 4 For the disk group you created for campus clusters, add a DiskGroup resource to the VCS service group app_sg.

```
hares -add dg_res1 DiskGroup app_sg
hares -modify dg_res1 DiskGroup diskgroup_name
hares -modify dg_res1 Enabled 1
```
 - 5 Configure the application and other related resources to the app_sg service group.
 - 6 Bring the service group online.

About fire drill in campus clusters

Fire drill tests the disaster-readiness of a configuration by mimicking a failover without stopping the application and disrupting user access.

The process involves creating a fire drill service group, which is similar to the original application service group. Bringing the fire drill service group online on the remote node demonstrates the ability of the application service group to fail over and come online at the site, should the need arise.

Fire drill service groups do not interact with outside clients or with other instances of resources, so they can safely come online even when the application service group is online. Conduct a fire drill only at the remote site; do not bring the fire drill service group online on the node hosting the original application.

About the DiskGroupSnap agent

The DiskGroupSnap agent verifies the VxVM diskgroups and volumes for site awareness and disaster readiness in a campus cluster environment. To perform a fire drill in campus clusters, you must configure a resource of type DiskGroupSnap in the fire drill service group.

Note: To perform fire drill, the application service group must be online at the primary site.

During fire drill, the DiskGroupSnap agent does the following:

- For each node in a site, the agent correlates the value of the SystemZones attribute for the application service group to the VxVM site names for that node.

- For the diskgroup in the application service group, the agent verifies that the VxVM site tags are defined for the diskgroup.
- For the diskgroup in the application service group, the agent verifies that the disks at the secondary site are not tagged with the same VxVM site name as the disks at the primary site.
- The agent verifies that all volumes in the diskgroup have a plex at each site.

See *Veritas Cluster Server Bundled Agents Reference Guide* for more information on the agent.

Running a fire drill in a campus cluster

Do the following tasks to perform fire drill:

- [Configuring the fire drill service group](#)
- [Running a successful fire drill in a campus cluster](#)

Configuring the fire drill service group

Perform the following steps to configure the fire drill service group.

To configure the fire drill service group

- 1 Configure a fire drill service group similar to the application service group with the following exceptions:
 - The AutoFailOver attribute must be set to 0.
 - Network-related resources must not be configured.
 - The diskgroup names for the DiskGroup and the Mount resources in the fire drill service group must be appended with “_fd”.
For example, if the value of the DiskGroup attribute in the application service group is ccdg, then the corresponding value in the fire drill service group must be ccdg_fd.
If the value of the BlockDevice attribute for the Mount resource in the application service group is /dev/vx/dsk/ccdg/ccvol, then the corresponding value in the fire drill service group must be /dev/vx/dsk/ccdg_fd/ccvol.
- 2 Add a resource of type DiskGroupSnap. Define the TargetResName and the FDSiteName attributes for the DiskGroupSnap resource.
See *Veritas Cluster Server Bundled Agent Reference Guide* for attribute descriptions.

- 3 Create a dependency such that the DiskGroup resource depends on the DiskGroupSnap resource.
- 4 Create a group dependency such that the fire drill service group has an offline local dependency on the application service group.

Running a successful fire drill in a campus cluster

Bring the fire drill service group online on a node within the system zone that does not have the application running. Verify that the fire drill service group comes online. This action validates that your solution is configured correctly and the production service group will fail over to the remote site in the event of an actual failure (disaster) at the local site.

You must take the fire drill service group offline before you shut down the node or stop VCS locally on the node where the fire drill service group is online or where the diskgroup is online. Otherwise, after the node restarts you must manually reattach the fire drill site to the diskgroup that is imported at the primary site.

Note: For the applications for which you want to perform fire drill, you must set the value of the FireDrill attribute for those application resource types to 1. After you complete fire drill, reset the value to 0.

To run a successful fire drill

- 1 Set the FireDrill attribute for the application resource type to 1 to prevent the agent from reporting a concurrency violation when the application service group and the fire drill service group are online at the same time.
- 2 Bring the fire drill service group online.
If the fire drill service group does not come online, review the VCS engine log to troubleshoot the issues so that corrective action can be taken as necessary in the production service group.

Warning: You must take the fire drill service group offline after you complete the fire drill so that the failover behavior of the application service group is not impacted. Otherwise, when a disaster strikes at the primary site, the application service group cannot fail over to the secondary site due to resource conflicts.

- 3 After you complete the fire drill, take the fire drill service group offline.
- 4 Reset the FireDrill attribute for the application resource type to 0.

Troubleshooting and performance

- [Chapter 20, “VCS performance considerations” on page 563](#)
- [Chapter 21, “Troubleshooting and recovery for VCS” on page 585](#)

VCS performance considerations

- [How cluster components affect performance](#)
- [How cluster operations affect performance](#)
- [Scheduling class and priority configuration](#)
- [Monitoring CPU usage](#)
- [VCS agent statistics](#)
- [About VXFEN tunable parameters](#)

How cluster components affect performance

VCS and its agents run on the same systems as the applications. Therefore, VCS attempts to minimize its impact on overall system performance. The three main components of clustering that have an impact on performance include the kernel; specifically, GAB and LLT, the VCS engine (HAD), and the VCS agents. For details on attributes or commands mentioned in the following sections, see the chapter on administering VCS from the command line and the appendix on VCS attributes.

Kernel components (GAB and LLT)

Typically, overhead of VCS kernel components is minimal. Kernel components provide heartbeat and atomic information exchange among cluster systems. By default, each system in the cluster sends two small heartbeat packets per second to other systems in the cluster. Heartbeat packets are sent over all network links configured in the following LLT configuration file:

`/etc/llttab`

System-to-system communication is load-balanced across all private network links. If a link fails, VCS continues to use all remaining links. Typically, network links are private and do not increase traffic on the public network or LAN. You can configure a public network (LAN) link as low-priority, which by default generates a small (approximately 64-byte) unicast packet per second from each system, and which will carry data only when all private network links have failed.

The VCS engine (HAD)

The VCS engine, HAD, runs as a daemon process. By default it runs as a high-priority process, which ensures it sends heartbeats to kernel components and responds quickly to failures. HAD runs logging activities in a separate thread to reduce the performance impact on the engine due to logging.

VCS waits in a loop waiting for messages from agents, ha commands, the graphical user interfaces, and the other systems. Under normal conditions, the number of messages processed by HAD is few. They mainly include heartbeat messages from agents and update messages from the global counter. VCS may exchange additional messages when an event occurs, but typically overhead is nominal even during events. Note that this depends on the type of event; for example, a resource fault may involve taking the group offline on one system and bringing it online on another system. A system fault invokes failing over all online service groups on the faulted system.

To continuously monitor VCS status, use the VCS graphical user interfaces or the command `hastatus`. Both methods maintain connection to VCS and register for events, and are more efficient compared to running commands like `hastatus -summary` or `hasys` in a loop.

The number of clients connected to VCS can affect performance if several events occur simultaneously. For example, if five GUI processes are connected to VCS, VCS sends state updates to all five. Maintaining fewer client connections to VCS reduces this overhead.

How agents impact performance

The VCS agent processes have the most impact on system performance. Each agent process has two components: the agent framework and the agent functions. The agent framework provides common functionality, such as communication with the HAD, multithreading for multiple resources, scheduling threads, and invoking functions. Agent functions implement agent-specific functionality. Follow the performance guidelines below when configuring agents.

Monitoring resource type and agent configuration

By default, VCS monitors each resource every 60 seconds. You can change this by modifying the `MonitorInterval` attribute for the resource type. You may consider reducing monitor frequency for non-critical or resources with expensive monitor operations. Note that reducing monitor frequency also means that VCS may take longer to detect a resource fault.

By default, VCS also monitors offline resources. This ensures that if someone brings the resource online outside of VCS control, VCS detects it and flags a

concurrency violation for failover groups. To reduce the monitoring frequency of offline resources, modify the `OfflineMonitorInterval` attribute for the resource type.

The VCS agent framework uses multithreading to allow multiple resource operations to run in parallel for the same type of resources. For example, a single Mount agent handles all mount resources. The number of agent threads for most resource types is 10 by default. To change the default, modify the `NumThreads` attribute for the resource type. The maximum value of the `NumThreads` attribute is 30.

Continuing with this example, the Mount agent schedules the `monitor` function for all mount resources, based on the `MonitorInterval` or `OfflineMonitorInterval` attributes. If the number of mount resources is more than `NumThreads`, the monitor operation for some mount resources may be required to wait to execute the `monitor` function until the thread becomes free.

Additional considerations for modifying the `NumThreads` attribute include:

- If you have only one or two resources of a given type, you can set `NumThreads` to a lower value.
- If you have many resources of a given type, evaluate the time it takes for the `monitor` function to execute and the available CPU power for monitoring. For example, if you have 50 mount points, you may want to increase `NumThreads` to get the ideal performance for the Mount agent without affecting overall system performance.

You can also adjust how often VCS monitors various functions by modifying their associated attributes. The attributes `MonitorTimeout`, `OnlineTimeout`, and `OfflineTimeout` indicate the maximum time (in seconds) within which the monitor, online, and offline functions must complete or else be terminated. The default for the `MonitorTimeout` attribute is 60 seconds. The defaults for the `OnlineTimeout` and `OfflineTimeout` attributes is 300 seconds. For best results, Symantec recommends measuring the time it takes to bring a resource online, take it offline, and monitor before modifying the defaults. Issue an online or offline command to measure the time it takes for each action. To measure how long it takes to monitor a resource, fault the resource and issue a probe, or bring the resource online outside of VCS control and issue a probe.

Agents typically run with normal priority. When you develop agents, consider the following:

- If you write a custom agent, write the monitor function using C or C++. If you write a script-based monitor, VCS must invoke a new process each time with the monitor. This can be costly if you have many resources of that type.
- If monitoring the resources is proving costly, you can divide it into cursory, or shallow monitoring, and the more extensive deep (or in-depth)

monitoring. Whether to use shallow or deep monitoring depends on your configuration requirements.

Additional considerations for agents

Properly configure the attribute `SystemList` for your service group. For example, if you know that a service group can go online on `sysa` and `sysb` only, *do not* include other systems in the `SystemList`. This saves additional agent processes and monitoring overhead.

The VCS graphical user interfaces

The VCS graphical user interfaces, Cluster Manager (Java Console) and Cluster Management Console maintain a persistent connection to HAD, from which they receive regular updates regarding cluster status. For best results, run the GUIs on a system outside the cluster to avoid impact on node performance.

How cluster operations affect performance

This section describes how operations on systems, resources, and service groups in the cluster affect performance.

Booting a cluster system

When a cluster system boots, the kernel drivers and VCS process start in a particular order. If it is the first system in the cluster, VCS reads the cluster configuration file `main.cf` and builds an “in-memory configuration database. This is the `LOCAL_BUILD` state. After building the configuration database, the system transitions into the `RUNNING` mode. If another system joins the cluster while the first system is in the `LOCAL_BUILD` state, it must wait until the first system transitions into `RUNNING` mode. The time it takes to build the configuration depends on the number of service groups in the configuration and their dependencies, and the number of resources per group and resource dependencies. VCS creates an object for each system, service group, type, and resource. Typically, the number of systems, service groups and types are few, so the number of resources and resource dependencies determine how long it takes to build the configuration database and get VCS into `RUNNING` mode. If a system joins a cluster in which at least one system is in `RUNNING` mode, it builds the configuration from the lowest-numbered system in that mode.

Note: Bringing service groups online as part of AutoStart occurs after VCS transitions to `RUNNING` mode.

When a resource comes online

The online function of an agent brings the resource online. This function may return before the resource is fully online. The subsequent monitor determines if the resource is online, then reports that information to VCS. The time it takes to bring a resource online equals the time for the resource to go online, plus the time for the subsequent monitor to execute and report to VCS.

Most resources are online when the online function finishes. The agent schedules the monitor immediately after the function finishes, so the first monitor detects the resource as online. However, for some resources, such as a database server, recovery can take longer. In this case, the time it takes to bring a resource online depends on the amount of data to recover. It may take multiple monitor intervals before a database server is reported online. When this occurs, it is important to have the correct values configured for the `OnlineTimeout` and `OnlineWaitLimit` attributes of the database server resource type.

When a resource goes offline

Similar to the online function, the offline function takes the resource offline and may return before the resource is actually offline. Subsequent monitoring confirms whether the resource is offline. The time it takes to offline a resource equals the time it takes for the resource to go offline, plus the duration of subsequent monitoring and reporting to VCS that the resource is offline. Most resources are typically offline when the offline function finishes. The agent schedules the monitor immediately after the offline function finishes, so the first monitor detects the resource as offline.

When a service group comes online

The time it takes to bring a service group online depends on the number of resources in the service group, the service group dependency structure, and the time to bring the group's resources online. For example, if service group G1 has three resources, R1, R2, and R3 (where R1 depends on R2 and R2 depends on R3), VCS first onlines R3. When R3 is online, VCS onlines R2. When R2 is online, VCS onlines R1. The time it takes to online G1 equals the time it takes to bring all resources online. However, if R1 depends on both R2 and R3, but there was no dependency between them, the online operation of R2 and R3 is started in parallel. When both are online, R1 is brought online. The time it takes to online the group is Max (the time to online R2 and R3), plus the time to online R1. Typically, broader service group trees allow more parallel operations and can be brought online faster. More complex service group trees do not allow much parallelism and serializes the group online operation.

When a service group goes offline

Taking service groups offline works from the top down, as opposed to the online operation, which works from the bottom up. The time it takes to offline a service group depends on the number of resources in the service group and the time to offline the group's resources. For example, if service group G1 has three resources, R1, R2, and R3, VCS first offlines R1. When R1 is offline, VCS offlines R2. When R2 is offline, VCS offlines R3. The time it takes to offline G1 equals the time it takes for all resources to go offline.

When a resource fails

The time it takes to detect a resource fault or failure depends on the MonitorInterval attribute for the resource type. When a resource faults, the next monitor detects it. The agent may not declare the resource as faulted if the ToleranceLimit attribute is set to non-zero. If the monitor function reports offline more often than the number set in ToleranceLimit, the resource is

declared faulted. However, if the resource remains online for the interval designated in the `ConfInterval` attribute, previous reports of offline are not counted against `ToleranceLimit`.

When the agent determines that the resource is faulted, it calls the `clean` function (if implemented) to verify that the resource is completely offline. The monitor following `clean` verifies the offline. The agent then tries to restart the resource according to the number set in the `RestartLimit` attribute (if the value of the attribute is non-zero) before it gives up and informs HAD that the resource is faulted. However, if the resource remains online for the interval designated in `ConfInterval`, earlier attempts to restart are not counted against `RestartLimit`.

In most cases, `ToleranceLimit` is 0. The time it takes to detect a resource failure is the time it takes the agent monitor to detect failure, plus the time to clean up the resource if the `clean` function is implemented. Therefore, the time it takes to detect failure depends on the `MonitorInterval`, the efficiency of the monitor and `clean` (if implemented) functions, and the `ToleranceLimit` (if set).

In some cases, the failed resource may hang and may also cause the monitor to hang. For example, if the database server is hung and the monitor tries to query, the monitor will also hang. If the `monitor` function is hung, the agent eventually kills the thread running the function. By default, the agent times out the `monitor` function after 60 seconds. This can be adjusted by changing the `MonitorTimeout` attribute. The agent retries `monitor` after the `MonitorInterval`. If the `monitor` function times out consecutively for the number of times designated in the attribute `FaultOnMonitorTimeouts`, the agent treats the resource as faulted. The agent calls `clean`, if implemented. The default value of `FaultOnMonitorTimeouts` is 4, and can be changed according to the type. A high value of this parameter delays detection of a fault if the resource is hung. If the resource is hung and causes the `monitor` function to hang, the time to detect it depends on `MonitorTimeout`, `FaultOnMonitorTimeouts`, and the efficiency of `monitor` and `clean` (if implemented).

When a system fails

When a system crashes or is powered off, it stops sending heartbeats to other systems in the cluster. By default, other systems in the cluster wait 21 seconds before declaring it dead. The time of 21 seconds derives from 16 seconds default value for LLT peer inactive timeout, plus 5 seconds default value for GAB stable timeout. The default peer inactive timeout is 16 seconds, and can be modified in the `/etc/llttab` file. For example, to specify 12 seconds:

```
set-timer peerinact:1200
```

Note: After modifying the peer inactive timeout, you must unconfigure, then restart LLT before the change is implemented. To unconfigure LLT, type `lltconfig -U`. To restart LLT, type `lltconfig -c`.

GAB stable timeout can be changed by specifying:

```
gabconfig -t timeout_value_milliseconds
```

Though this can be done, we *do not* recommend changing the values of the LLT peer inactive timeout and GAB stable timeout.

If a system boots, it becomes unavailable until the reboot is complete. The reboot process kills all processes, including HAD. When the VCS process is killed, other systems in the cluster mark all service groups that can go online on the rebooted system as autodisabled. The `AutoDisabled` flag is cleared when the system goes offline. As long as the system goes offline within the interval specified in the `ShutdownTimeout` value, VCS treats this as a system reboot. The `ShutdownTimeout` default value can be changed by modifying the attribute.

See “[System attributes](#)” on page 680.

When a network link fails

If a system loses a network link to the cluster, other systems stop receiving heartbeats over the links from that system. LLT detects this failure and waits for 16 seconds (default value for LLT peer inactive timeout) before declaring that the system lost a link.

You can modify the LLT peer inactive timeout value in the `/etc/llttab` file. For example, to specify 12 seconds:

```
set-timer peerinact:1200
```

Note: After modifying the peer inactive timeout, you must unconfigure, then restart LLT before the change is implemented. To unconfigure LLT, type `lltconfig -U`. To restart LLT, type `lltconfig -c`.

When a system panics

There are several instances in which GAB will intentionally panic a system, including if it detects an internal protocol error or discovers an LLT node-ID conflict. Under severe load conditions, if VCS cannot heartbeat with GAB, it may induce a panic or a VCS restart.

This section describes the scenarios when GAB panics a system.

Client process failure

If a client process fails to heartbeat to GAB, the process is killed. If the process hangs or cannot be killed, GAB halts the system. This could also occur if the node is overloaded and HAD cannot run on the node.

If the `-k` option is used in the `gabconfig` command, GAB tries to kill the client process until successful, which may have an impact on the entire cluster. If the `-b` option is used in `gabconfig`, GAB does not try to kill the client process. Instead, it panics the system when the client process fails to heartbeat. This option cannot be turned off once set.

HAD heartbeats with GAB at regular intervals. The heartbeat timeout is specified by HAD when it registers with GAB; the default is 15 seconds. If HAD gets stuck within the kernel and cannot heartbeat with GAB within the specified timeout, GAB tries to kill HAD by sending a SIGABRT signal. If it does not succeed, GAB sends a SIGKILL and closes the port. By default, GAB tries to kill HAD five times before closing the port. The number of times GAB tries to kill HAD is a kernel tunable parameter, `gab_kill_ntries`, and is configurable. The minimum value for this tunable is 3 and the maximum is 10.

This is an indication to other nodes that HAD on this node has been killed. Should HAD recover from its stuck state, it first processes pending signals. Here it will receive the SIGKILL first and get killed.

After sending a SIGKILL, GAB waits for a specific amount of time for HAD to get killed. If HAD survives beyond this time limit, GAB panics the system. This time limit is a kernel tunable parameter, `gab_isolate_time` and is configurable. The minimum value for this timer is 16 seconds and maximum is 4 minutes.

Registration monitoring

The registration monitoring features lets you configure GAB behavior when HAD is killed and does not reconnect after a specified time interval.

This scenario may occur in the following situations:

- The system is very busy and the hashadow process cannot restart HAD.
- The HAD and hashadow processes were killed by user intervention.
- The hashadow process restarted HAD, but HAD could not register.

- A hardware/DIMM failure causes termination of the HAD and hashadow processes.

When this occurs, the registration monitoring timer starts. GAB takes action if HAD does not register within the time defined by the `VCS_GAB_RMTIMEOUT` parameter, which is defined in the `vcseenv` file. The default value for `VCS_GAB_RMTIMEOUT` is 200 seconds.

When HAD cannot register after the specified time period, GAB logs a message every 15 seconds saying it will panic the system.

You can control GAB behavior in this situation by setting the `VCS_GAB_RMACTION` parameter in the `vcseenv` file.

- To configure GAB to panic the system in this situation, set:

```
VCS_GAB_RMACTION=PANIC
```

In this configuration, killing the HAD and hashadow processes results in a panic unless you start HAD within the registration monitoring timeout interval.

- To configure GAB to log a message in this situation, set:

```
VCS_GAB_RMACTION=SYSLOG
```

The default value of this parameter is `SYSLOG`, which configures GAB to log a message when HAD does not reconnect after the specified time interval.

In this scenario, you can choose to restart HAD (using `hastart`) or unconfigure GAB (using `gabconfig -U`).

When you enable registration monitoring, GAB takes no action if the HAD process unregisters with GAB normally, that is if you stop HAD using the `hastop` command.

Network failure

If a network partition occurs, a cluster can “split into two or more separate sub-clusters. When two clusters join as one, VCS designates that one system be ejected. GAB prints diagnostic messages and sends iofence messages to the system being ejected. The system receiving the iofence messages tries to kill the client process. The `-k` option applied here. If the `-j` option is used in `gabconfig`, the system is halted when the iofence message is received.

GAB panics the system on receiving an iofence message on kernel client port and tries to kill only for clients running in user land. The `-k` option does not work for clients running in the kernel.

Quick reopen

If a system leaves cluster and tries to join the cluster before the new cluster is configured (default is five seconds), the system is sent an iofence message with

reason set to “quick reopen. When the system receives the message, it tries to kill the client process.

When a service group switches over

The time it takes to switch a service group equals the time to offline a service group on the source system, plus the time to bring the service group online on the target system.

When a service group fails over

The time it takes to fail over a service group when a resource faults equals

- the time it takes to detect the resource fault
- the time it takes to offline the service group on source system
- the time it takes for the VCS policy module to select target system
- the time it takes to bring the service group online on target system

The time it takes to fail over a service group when a system faults equals

- the time it takes to detect system fault
- the time it takes to offline the service group on source system
- the time it takes for the VCS policy module to select target system
- the time it takes to bring the service group online on target system

The time it takes the VCS policy module to determine the target system is negligible in comparison to the other factors.

If you have a firm group dependency and the child group faults, VCS offlines all immediate and non-immediate parent groups before bringing the child group online on the target system. Therefore, the time it takes a parent group to be brought online also depends on the time it takes the child group to be brought online.

Scheduling class and priority configuration

VCS allows you to specify priorities and scheduling classes for VCS processes. VCS supports the following scheduling classes:

- RealTime (specified as “RT in the configuration file)
- TimeSharing (specified as “TS in the configuration file)

Priority ranges

The following table displays the platform-specific priority range for RealTime, TimeSharing, and SRM scheduling (SHR) processes.

Table 20-1 Priority ranges

Platform	Scheduling Class	Default Priority Range Weak / Strong	Priority Range Using #ps Commands
AIX	RT	126 / 50	126 / 50
	TS	60	Priority varies with CPU consumption. Note: On AIX, use #ps -ael

Default scheduling classes and priorities

The following table lists the default class and priority values used by VCS. The class and priority of trigger processes are determined by the attributes ProcessClass (default = TS) and ProcessPriority (default = ""). Both attributes can be modified according to the class and priority at which the trigger processes run.

Table 20-2 Default scheduling classes and priorities

Process	Engine	Process created by engine	Agent	Script
Default Scheduling Class	RT	TS	TS	TS
Default Priority (AIX)	52 (Strongest + 2)	0	0	0

Note: For standard configurations, Symantec recommends using the default values for scheduling unless specific configuration requirements dictate otherwise.

Note that the default priority value is platform-specific. When priority is set to "" (empty string), VCS converts the priority to a value specific to the platform on which the system is running. For TS, the default priority equals the strongest priority supported by the TimeSharing class. For RT, the default priority equals two less than the strongest priority supported by the RealTime class. So, if the strongest priority supported by the RealTime class is 59, the default priority for the RT class is 57. For SHR (on Solaris only), the default priority is the strongest priority support by the SHR class.

Configuring priorities for LLT threads for AIX

On an highly loaded system, LLT threads might not be able to run because of relatively low priority settings. This situation may result in heartbeat failures across cluster nodes.

You can modify the priorities for the service thread by configuring kernel tunables in the file `/etc/llt.conf`.

LLT thread	Default priority	Priority range	Tunable to configure thread priority
LLT service thread	40	40 to 60	<code>llt_thread_pri</code>

Note that the priorities must lie within the priority range. If you specify a priority outside the range, LLT resets it to a value within the range.

To specify priority value for LLT service thread

- ◆ Specify priority for the tunables by setting the following entry in the file `/etc/llt.conf`:

```
llt_thread_pri=<priority value>
```

Note that these changes take effect when the LLT module reloads.

Sample `llt.conf` file

```
llt_maxnids=32
llt_maxports=32
llt_nominpad=0
llt_basetimer=50000
llt_thread_pri=40
```

Monitoring CPU usage

VCS includes a system attribute, `CPUUsageMonitoring`, which monitors CPU usage on a specific system and notifies the administrator when usage has been exceeded.

The default values for the `CPUUsageMonitoring` attribute are:

- `Enabled = 0`
- `NotifyThreshold = 0`
- `NotifyTimeLimit = 0`
- `ActionThreshold = 0`
- `ActionTimeLimit = 0`
- `Action = NONE`.

The values for `ActionTimeLimit` and `NotifyTimeLimit` represent the time in seconds. The values for `ActionThreshold` and `NotifyThreshold` represent the threshold in terms of CPU percentage utilization.

If `Enabled` is set to 1, HAD monitors the usage and updates `CPUUsage` attribute. If `Enabled` is set to 0 (default), HAD does not monitor the usage.

If the system's CPU usage continuously exceeds the value set in `NotifyThreshold` for a duration greater than the value set in `NotifyTimeLimit`, HAD sends notification via an SNMP trap or SMTP message.

If the CPU usage continuously exceeds the value set in `NotifyThreshold` for a duration greater than the value set in `NotifyTimeLimit`, subsequent notifications are sent after five minutes to avoid sending notifications too frequently (if the `NotifyTimeLimit` value is set to a value less than five minutes). In this case, notification is sent after the first interval of `NotifyTimeLimit`. As CPU usage continues to exceed the threshold value, notifications are sent after five minutes. If the values of `NotifyThreshold` or `NotifyTimeLimit` are set to 0, no notification is sent.

If system's CPU usage exceeds the value set in `ActionThreshold` continuously for a duration greater than the value set in `ActionTimeLimit`, the specified action is taken. If the CPU usage continuously exceeds the `ActionThreshold` for a duration greater than the value set in `ActionTimeLimit`, subsequent action is taken after five minutes to avoid taking action too frequently (if the `ActionTimeLimit` value is set to less than five minutes). In this case action is taken after the first interval of `ActionTimeLimit`. As CPU usage continues to exceed the threshold value, action is taken after five minutes. If the values of `ActionThreshold` or `ActionTimeLimit` are set to 0, no action is taken. Actions can have one of the following values:

NONE: No action will be taken and the message is logged in the VCS engine log.

REBOOT: System is rebooted.

CUSTOM: The cpususage trigger is invoked.

VCS agent statistics

You can configure VCS to track the time taken for monitoring resources.

You can use these statistics to configure the MonitorTimeout attribute.

You can also detect potential problems with resources and systems on which resources are online by analyzing the trends in the time taken by the resource's monitor cycle. Note that VCS keeps track of monitor cycle times for online resources only.

VCS calculates the time taken for a monitor cycle to complete and computes an average of monitor times after a specific number of monitor cycles and stores the average in a resource-level attribute.

VCS also tracks increasing trends in the monitor cycle times and sends notifications about sudden and gradual increases in monitor times.

VCS uses the following parameters to compute the average monitor time and to detect increasing trends in monitor cycle times:

- *Frequency*: The number of monitor cycles after which the monitor time average is computed and sent to the VCS engine.
For example, if Frequency is set to 10, VCS computes the average monitor time after every 10 monitor cycles.
- *ExpectedValue*: The expected monitor time (in milliseconds) for a resource. VCS sends a notification if the actual monitor time exceeds the expected monitor time by the ValueThreshold. So, if you set this attribute to 5000 for a FileOnOff resource, and if ValueThreshold is set to 40%, VCS will send a notification only when the monitor cycle for the FileOnOff resource exceeds the expected time by over 40%, that is 7000 milliseconds.
- *ValueThreshold*: The maximum permissible deviation (in percent) from the expected monitor time. When the time for a monitor cycle exceeds this limit, VCS sends a notification about the sudden increase or decrease in monitor time.
For example, a value of 100 means that VCS sends a notification if the actual monitor time deviates from the expected time by over 100%. VCS sends these notifications conservatively. If 12 consecutive monitor cycles exceed the threshold limit, VCS sends a notification for the first spike, and then a collective notification for the next 10 consecutive spikes.

- *AvgThreshold*: The threshold value (in percent) for increase in the average monitor cycle time for a resource.
VCS maintains a running average of the time taken by the monitor cycles of a resource. The first such computed running average is used as a benchmark average. If the current running average for a resource differs from the benchmark average by more than this threshold value, VCS regards this as a sign of gradual increase or decrease in monitor cycle times and sends a notification about it for the resource. Whenever such an event occurs, VCS resets the internally maintained benchmark average to this new average. VCS sends notifications regardless of whether the deviation is an increase or decrease in the monitor cycle time.
For example, a value of 25 means that if the actual average monitor time is 25% more than the benchmark monitor time average, VCS sends a notification.

Tracking monitor cycle times

VCS marks sudden changes in monitor times by comparing the time taken for each monitor cycle with the *ExpectedValue*. If this difference exceeds the *ValueThreshold*, VCS sends a notification about the sudden change in monitor time. Note that VCS sends this notification only if monitor time increases.

VCS marks gradual changes in monitor times by comparing the benchmark average and the moving average of monitor cycle times. VCS computes the benchmark average after a certain number of monitor cycles and computes the moving average after every monitor cycle. If the current moving average exceeds the benchmark average by more than the *AvgThreshold*, VCS sends a notification about this gradual change in the monitor cycle time.

VCS attributes enabling agent statistics

This section describes the attributes that enable VCS agent statistics.

MonitorStatsParam	<p>A resource type-level attribute, which stores the required parameter values for calculating monitor time statistics.</p> <pre>static str MonitorStatsParam = { Frequency = 10, ExpectedValue = 3000, ValueThreshold = 100, AvgThreshold = 40 }</pre> <ul style="list-style-type: none">■ <i>Frequency</i>: Defines the number of monitor cycles after which the average monitor cycle time should be computed and sent to the engine. If configured, the value for this attribute must be between 1 and 30. It is set to 0 by default.■ <i>ExpectedValue</i>: The expected monitor time in milliseconds for all resources of this type. Default=3000.■ <i>ValueThreshold</i>: The acceptable percentage difference between the expected monitor cycle time (<i>ExpectedValue</i>) and the actual monitor cycle time. Default=100.■ <i>AvgThreshold</i>: The acceptable percentage difference between the benchmark average and the moving average of monitor cycle times. Default=40
MonitorTimeStats	<p>Stores the average time taken by a number of monitor cycles specified by the <i>Frequency</i> attribute along with a timestamp value of when the average was computed.</p> <pre>str MonitorTimeStats{} = { Avg = "0", TS = "" }</pre> <p>This attribute is updated periodically after a number of monitor cycles specified by the <i>Frequency</i> attribute. If <i>Frequency</i> is set to 10, the attribute stores the average of 10 monitor cycle times and is updated after every 10 monitor cycles.</p> <p>The default value for this attribute is 0.</p>
ComputeStats	<p>A flag that specifies whether VCS keeps track of the monitor times for the resource.</p> <pre>bool ComputeStats = 0</pre> <p>The value 0 indicates that VCS will not keep track of the time taken by the monitor routine for the resource. The value 1 indicates that VCS keeps track of the monitor time for the resource.</p> <p>The default value for this attribute is 0.</p>

About VXFEN tunable parameters

Table 20-3 describes tunable parameters for the VXFEN driver.

Table 20-3 VXFEN tunable parameters

vxfen Parameter	Description and Values: Default, Minimum, and Maximum
vxfen_deblog_sz	<p>Size of debug log in bytes</p> <ul style="list-style-type: none">ValuesDefault: 65536Minimum: 65536Maximum: 256K
vxfen_max_delay and vxfen_min_delay (See below.)	<p>In the event of a network partition, the smaller sub-cluster delays before racing for the coordinator disks. The time delayed allows a larger sub-cluster to win the race for the coordinator disks. The vxfen_max_delay and vxfen_min_delay parameters define the delay in seconds.</p>
vxfen_max_delay	<p>Specifies the maximum number of seconds that the smaller sub-cluster waits before racing with larger clusters for control of the coordinator disks.</p> <p>This value must be greater than the vxfen_min_delay value.</p> <ul style="list-style-type: none">ValuesDefault: 60Minimum: 0Maximum: 600
vxfen_min_delay	<p>Specifies the minimum number of seconds that the smaller sub-cluster waits before racing with larger sub-clusters for control of the coordinator disks. This value must be smaller than the vxfen_max_delay value.</p> <ul style="list-style-type: none">ValuesDefault: 1Minimum: 0Maximum: 600
vxfen_vxfnd_tmt	<p>Specifies the time in seconds that the I/O fencing driver VxPEN waits for the I/O fencing daemon VXFEND to return after completing a given task.</p> <ul style="list-style-type: none">ValuesDefault: 60Minimum: 10Maximum: 600

See “[Configuring the VXFEN parameters](#)” on page 583.

Configuring the VXFEN parameters

For the parameter changes to take effect, reconfigure the VXFEN module.

To change the VXFEN tunable parameter, use the `chdev` command, unload the driver, and reload the driver. Change the parameter on each cluster system. The example procedure changes the value of the `vxfen_min_delay` parameter.

To change the VXFEN tunable parameter

- 1 Check the status of the driver and start it if necessary:

```
# /etc/methods/vxfenext -status
```

To start the driver:

```
# /etc/init.d/vxfen.rc start
```

- 2 List the current value of the tunable parameter:

For example:

```
# lsattr -El vxfen
vxfen_deblog_sz 65536 N/A True
vxfen_max_delay 3 N/A True
vxfen_min_delay 1 N/A True
vxfen_vxfnd_tmt 60 N/A True
```

The current value is 1 (the default).

- 3 Enter:

```
# chdev -l vxfen -P -a vxfen_min_delay=30
```

- 4 Ensure that Veritas Cluster Server is shut down. Unload and reload the driver after changing the value of the tunable:

```
# /etc/methods/vxfenext -stop
```

```
# /etc/methods/vxfenext -start
```

- 5 Perform the commands on each cluster system to change the parameter.

Troubleshooting and recovery for VCS

- [Logging](#)
- [Troubleshooting the VCS engine](#)
- [Troubleshooting VCS startup](#)
- [Troubleshooting service groups](#)
- [Troubleshooting resources](#)
- [Troubleshooting I/O fencing](#)
- [Troubleshooting notification](#)
- [Troubleshooting VCS configuration backup and restore](#)
- [Troubleshooting and recovery for global clusters](#)
- [Troubleshooting licensing](#)

Logging

VCS generates two error message logs: the engine log and the agent log. Log file names are appended by letters. Letter A indicates the first log file, B the second, C the third, and so on.

The engine log is located at `/var/VRTSvcs/log/engine_A.log`. The format of engine log messages is:

Timestamp (Year/MM/DD) | Mnemonic | Severity | UMI | Message Text

- *Timestamp*: the date and time the message was generated.
- *Mnemonic*: the string ID that represents the product (for example, VCS).
- *Severity*: levels include CRITICAL, ERROR, WARNING, NOTICE, and INFO (most to least severe, respectively).
- *UMI*: a unique message ID.
- *Message Text*: the actual message generated by VCS.

A typical engine log resembles:

```
2003/02/10 16:08:09 VCS INFO V-16-1-10077 received new
cluster membership.
```

The agent log is located at `/var/VRTSvcs/log/agent_A.log`. The format of agent log messages resembles:

Timestamp (Year/MM/DD) | Mnemonic | Severity | UMI | Agent Type |
Resource Name | Entry Point | Message Text

A typical agent log resembles:

```
2003/02/23 10:38:23 VCS WARNING V-16-2-23331
Oracle:VRT:monitor:Open for ora_lgwr failed, setting cookie to
null.
```

Note that the logs on all nodes may not be identical because

- VCS logs local events on the local nodes.
- All nodes may not be running when an event occurs.

VCS 5.1 prints the warning and error messages to STDERR.

If the VCS engine, Command Server, or any of the VCS agents encounter some problem, then First Failure Data Capture (FFDC) logs are generated and dumped along with other core dumps and stack traces to the following location:

- For VCS engine: `$VCS_DIAG/diag/had`
- For Command Server: `$VCS_DIAG/diag/CmdServer`
- For VCS agents: `$VCS_DIAG/diag/agents/type`, where *type* represents the specific agent type.

If the debug logging is not turned on, these FFDC logs are useful to analyze the issues that require professional support.

Enabling debug logs for agents

This section describes how to enable debug logs for VCS agents.

To enable debug logs for agents

- 1 Set the configuration to read-write:
- 2 Enable logging and set the desired log levels. The following example depicts the command for the IPMultiNIC resource type.

```
haconf -makerw
```

```
hatype -modify IPMultiNIC LogDbg DBG_1 DBG_2 DBG_4 DBG_21
```

See the description of the LogDbg attribute for more information.

See “[Resource type attributes](#)” on page 650.

- 3 For script-base agents, run the halog command to add the messages to the engine log:

```
halog -addtags DBG_1 DBG_2 DBG_4 DBG_21
```

- 4 Save the configuration.

```
haconf -dump -makero
```

For script entry points, log messages from all instances of the agent appear on all nodes in the engine log. For C++ entry points, messages from an instance of the agent appear in the agent log on the node on which the agent is running.

If DBG_AGDEBUG is set, the agent framework logs for an instance of the agent appear in the agent log on the node on which the agent is running.

Gathering LLT and GAB information for support analysis

You must run the getcomms script to gather LLT and GAB information when you encounter issues with LLT and GAB.

To gather LLT and GAB information for support analysis

- ◆ Run the following command on each node:

```
# getcomms -local
```

Troubleshoot and fix the issue.

See “[Troubleshooting LLT](#)” on page 590.

See “[Troubleshooting GAB](#)” on page 592.

If the issue cannot be fixed, then contact Symantec technical support with the file /tmp/commslog.*time_stamp*.tar that the getcomms script generates.

Message catalogs

VCS includes multilingual support for message catalogs. These binary message catalogs (BMCs), are stored in the following default locations. The variable *language* represents a two-letter abbreviation.

```
/opt/VRTS/messages/language/module_name
```

The VCS command-line interface displays error and success messages in VCS-supported languages. The `hamsg` command displays the VCS engine logs in VCS-supported languages.

The BMCs are:

<code>gcoconfig.bmc</code>	gcoconfig messages
<code>VRTSvcsHbfbw.bmc</code>	Heartbeat framework messages
<code>VRTSvcsTriggers.bmc</code>	VCS trigger messages
<code>VRTSvcsWac.bmc</code>	Wide-area connector process messages
<code>vxfen*.bmc</code>	Fencing messages
<code>gab.bmc</code>	GAB command-line interface messages
<code>hagetcf.bmc</code>	hagetcf messages
<code>llt.bmc</code>	LLT command-line interface messages
<code>VRTSvcsAgfw.bmc</code>	Agent framework messages
<code>VRTSvcsAlerts.bmc</code>	VCS alert messages
<code>VRTSvcsApi.bmc</code>	VCS API messages
<code>VRTSvcsCommon.bmc</code>	Common modules messages
<code>VRTSvcsHad.bmc</code>	VCS engine (HAD) messages
<code>VRTSvcsplatformAgent.bmc</code>	VCS bundled agent messages
<code>VRTSvcsplatformagent_name.bmc</code>	VCS enterprise agent messages

Troubleshooting the VCS engine

HAD diagnostics

When the VCS engine HAD dumps core, the core is written to the directory `$VCS_DIAG/diag/had`. The default value for variable `$VCS_DIAG` is `/var/VRTSvcs/`.

When HAD core dumps, review the contents of the `$VCS_DIAG/diag/had` directory. See the following logs for more information:

- Operating system console log
- Engine log
- hashadow log

VCS runs the script `/opt/VRTSvcs/bin/vcs_diag` to collect diagnostic information when HAD and GAB encounter heartbeat problems. The diagnostic information is stored in the `$VCS_DIAG/diag/had` directory.

When HAD starts, it renames the directory to `had.timestamp`, where *timestamp* represents the time at which the directory was renamed.

DNS configuration issues cause GAB to kill HAD

If HAD is periodically killed by GAB for no apparent reason, review the HAD core files for any DNS resolver functions (`res_send()`, `res_query()`, `res_search()` etc) in the stack trace. The presence of DNS resolver functions may indicate DNS configuration issues.

The VCS High Availability Daemon (HAD) uses the `gethostbyname()` function. On UNIX platforms, if the file `/etc/nsswitch.conf` has DNS in the hosts entry, a call to the `gethostbyname()` function may lead to calls to DNS resolver methods.

If the name servers specified in the `/etc/resolve.conf` are not reachable or if there are any DNS configuration issues, the DNS resolver methods called may block HAD, leading to HAD not sending heartbeats to GAB in a timely manner.

Seeding and I/O fencing

When I/O fencing starts up, a check is done to make sure the systems that have keys on the coordinator points are also in the GAB membership. If the `gabconfig` command in `/etc/gabtab` allows the cluster to seed with less than the full number of systems in the cluster, or the cluster is forced to seed with the `gabconfig -c -x` command, it is likely that this check will not match. In this case, the fencing module will detect a possible split-brain condition, print an error and HAD will not start.

It is recommended to let the cluster automatically seed when all members of the cluster can exchange heartbeat signals to each other. In this case, all systems perform the I/O fencing key placement after they are already in the GAB membership.

Preonline IP check

You can enable a preonline check of a failover IP address to protect against network partitioning. The check pings a service group's configured IP address to verify it is not already in use. If it is, the service group is not brought online.

A second check verifies the system is connected to its public and private networks. If the system receives no response from a broadcast ping to the public network and a check of the private networks, it determines the system is isolated and does not bring the service group online.

To enable the preonline IP check

- ◆ Move the preonline trigger script from the sample triggers directory into the triggers directory:

```
# cp /opt/VRTSvcs/bin/sample_triggers/preonline_ipc  
/opt/VRTSvcs/bin/triggers/preonline
```

Change the file permissions to make it executable.

Troubleshooting LLT

This section includes error messages associated with LLT and provides descriptions and the recommended action.

LLT startup script displays errors

If more than one system on the network has the same LLT node ID and the same cluster ID, then the LLT startup script displays error messages similar to the following:

```
LLT lltnconfig ERROR V-14-2-15238 node 1 already exists in cluster  
8383 and has the address - 00:18:8B:E4:DE:27
```

```
LLT lltnconfig ERROR V-14-2-15241 LLT not configured, use -o to  
override this warning
```

```
LLT lltnconfig ERROR V-14-2-15664 LLT could not configure any link
```

Recommended Action: Ensure that all systems on the network have unique clusterid-nodeid pair. You can use the `ltdump -D` command to get the list of unique clusterid-nodeid pairs connected to the network. This utility is available only for LLT-over-ethernet.

LLT detects cross links usage

If LLT detects more than one link of a system that is connected to the same network, then LLT logs a warning message similar to the following in the syslog:

```
LLT WARNING V-14-1-10498 recvarpack cross links? links 0 and 2 saw  
the same peer link number 1 for node 1
```

Recommended Action: This is an informational message. LLT supports cross links. However, if this cross links is not an intentional network setup, then make sure that no two links from the same system go to the same network. That is, the LLT links need to be on separate networks.

LLT link status messages

[Table 21-4](#) describes the LLT logs messages such as trouble, active, inactive, or expired in the syslog for the links.

Table 21-4 LLT link status messages

Message	Description and Recommended action
LLT INFO V-14-1-10205 link 1 (<i>link_name</i>) node 1 in trouble	<p>This message implies that LLT did not receive any heartbeats on the indicated link from the indicated peer node for LLT peertrouble time. The default LLT peertrouble time is 2s for hi-pri links and 4s for lo-pri links.</p> <p><i>Recommended action:</i> If these messages sporadically appear in the syslog, you can ignore them. If these messages flood the syslog, then perform one of the following:</p> <ul style="list-style-type: none"> ■ Increase the peertrouble time to a higher value (but significantly lower than the peerinact value). Run the following command: <pre>lltconfig -T peertrouble:<value> for hipri link</pre> <pre>lltconfig - T peertroublelo:<value> for lopri links.</pre> See the lltconfig(1m) manual page for details. ■ Replace the LLT link. See “Adding and removing LLT links” on page 187.
LLT INFO V-14-1-10024 link 0 (<i>link_name</i>) node 1 active	<p>This message implies that LLT started seeing heartbeats on this link from that node.</p> <p><i>Recommended action:</i> No action is required. This message is informational.</p>
LLT INFO V-14-1-10032 link 1 (<i>link_name</i>) node 1 inactive 5 sec (510)	<p>This message implies that LLT did not receive any heartbeats on the indicated link from the indicated peer node for the indicated amount of time.</p> <p><i>Recommended action:</i> If the peer node has not actually gone down, check for the following:</p> <ul style="list-style-type: none"> ■ Check if the link has got physically disconnected from the system or switch. ■ Check for the link health and replace the link if necessary. See “Adding and removing LLT links” on page 187.

Table 21-4 LLT link status messages

Message	Description and Recommended action
LLT INFO V-14-1-10510 sent hbreq (NULL) on link 1 (<i>link_name</i>) node 1. 4 more to go. LLT INFO V-14-1-10510 sent hbreq (NULL) on link 1 (<i>link_name</i>) node 1. 3 more to go. LLT INFO V-14-1-10510 sent hbreq (NULL) on link 1 (<i>link_name</i>) node 1. 2 more to go. LLT INFO V-14-1-10032 link 1 (<i>link_name</i>) node 1 inactive 6 sec (510) LLT INFO V-14-1-10510 sent hbreq (NULL) on link 1 (<i>link_name</i>) node 1. 1 more to go. LLT INFO V-14-1-10510 sent hbreq (NULL) on link 1 (<i>link_name</i>) node 1. 0 more to go. LLT INFO V-14-1-10032 link 1 (<i>link_name</i>) node 1 inactive 7 sec (510) LLT INFO V-14-1-10509 link 1 (<i>link_name</i>) node 1 expired	<p>This message implies that LLT did not receive any heartbeats on the indicated link from the indicated peer node for more than LLT peerinact time. LLT attempts to request heartbeat mechanism (sends 5 hbreqs to the peer node) and if the peer node does not respond, LLT marks this link as “expired” for that peer node.</p> <p><i>Recommended action:</i> If the peer node has not actually gone down, check for the following:</p> <ul style="list-style-type: none">■ Check if the link has got physically disconnected from the system or switch.■ Check for the link health and replace the link if necessary. See “Adding and removing LLT links” on page 187.
LLT INFO V-14-1-10499 recvarpreq link 0 for node 1 addr change from 00:00:00:00:00:00 to 00:18:8B:E4:DE:27	<p>This message is logged when LLT learns the peer node’s address.</p> <p><i>Recommended action:</i> No action is required. This message is informational.</p>

Troubleshooting GAB

This section includes error messages associated with GAB and provides descriptions and the recommended action.

GAB timer issues

GAB timer only wakes up a thread that does the real work. If either the operating system did not get a chance to schedule the thread or the operating system did not fire the timer itself, then GAB might log a message similar to the following:

GAB INFO V-15-1-20124 timer not called for 11 seconds

Recommended Action: If this issue occurs rarely at times of high load, and does not cause any detrimental effects, the issue is benign. If the issue recurs, or if the issue causes repeated VCS daemon restarts or node panics, you must increase the priority of the thread (gab_aix_timerpri tunable parameter).

If problem persists, collect the following and contact Symantec support:

- sar or perfmgr data
- Any VCS daemon diagnostics
- Any panic dumps
- commslog on all nodes

Delay in port reopen

If a GAB port was closed and reopened before LLT state cleanup action is completed, then GAB logs a message similar to the following:

```
GAB INFO V-15-1-20102 Port v: delayed reopen
```

Recommended Action: If this issue occurs during a GAB reconfiguration, and does not recur, the issue is benign. If the issue persists, collect commslog from each node, and contact Symantec support.

Node panics due to client process failure

If VCS daemon does not heartbeat with GAB within the configured timeout specified in VCS_GAB_TIMEOUT (default 30sec) environment variable, the node panics with a message similar to the following:

```
GAB Port h halting node due to client process failure at 3:109
```

GABs attempt (five retries) to kill the VCS daemon fails if VCS daemon is stuck in the kernel in an uninterruptible state or the system is heavily loaded that the VCS daemon cannot die with a SIGKILL.

Recommended Action:

- In case of performance issues, increase the value of the VCS_GAB_TIMEOUT environment variable to allow VCS more time to heartbeat.
- In case of a kernel problem, configure GAB to not panic but continue to attempt killing the VCS daemon. Do the following:
 - Run the following command on each node:


```
gabconfig -k
```
 - Add the “-k” option to the gabconfig command in the /etc/gabtab file:


```
gabconfig -c -k -n 6
```
- In case the problem persists, collect sar or similar output and crash dumps and contact Symantec support.

Troubleshooting VCS startup

This section includes error messages associated with starting VCS (shown in bold text), and provides descriptions of each error and the recommended action.

“VCS:10622 local configuration missing”

“VCS:10623 local configuration invalid”

The local configuration is invalid.

Recommended Action: Start the VCS engine, HAD, on another system that has a valid configuration file. The system with the configuration error “pulls” the valid configuration from the other system.

Another method is to correct the configuration file on the local system and force VCS to reread the configuration file. If the file appears valid, verify that is not an earlier version.

Type the following commands to verify the configuration:

```
# cd /etc/VRTSvcs/conf/config
# hacf -verify .
```

“VCS:11032 registration failed. Exiting”

GAB was not registered or has become unregistered.

Recommended Action: GAB is registered by the `gabconfig` command in the file `/etc/gabtab`. Verify that the file exists and that it contains the command `gabconfig -c`.

GAB can become unregistered if LLT is set up incorrectly. Verify that the configuration is correct in `/etc/llttab`. If the LLT configuration is incorrect, make the appropriate changes and reboot.

“Waiting for cluster membership.”

This indicates that GAB may not be seeded. If this is the case, the command `gabconfig -a` does not show any members, and the following messages may appear on the console or in the event log.

```
GAB: Port a registration waiting for seed port membership
GAB: Port h registration waiting for seed port membership
```

Troubleshooting service groups

This section cites the most common problems associated with bringing service groups online and taking them offline. Bold text provides a description of the problem. Recommended action is also included, where applicable.

VCS does not automatically start service group.

VCS does not automatically start a failover service group if the VCS engine (HAD) in the cluster was restarted by the hashadow process.

This behavior prevents service groups from coming online automatically due to events such as GAB killing HAD because to high load, or HAD committing suicide to rectify unexpected error conditions.

System is not in RUNNING state.

Recommended Action: Type `hasys -display system` to verify the system is running.

See “[System states](#)” on page 638.

Service group not configured to run on the system.

The SystemList attribute of the group may not contain the name of the system.

Recommended Action: Use the output of the command `hagrp -display service_group` to verify the system name.

Service group not configured to autostart.

If the service group is not starting automatically on the system, the group may not be configured to AutoStart, or may not be configured to AutoStart on that particular system.

Recommended Action: Use the output of the command `hagrp -display service_group` to verify the values of the AutoStart and AutoStartList attributes.

Service group is frozen.

Recommended Action: Use the output of the command `hagrp -display service_group` to verify the value of the Frozen and TFrozen attributes. Use the command `hagrp -unfreeze` to unfreeze the group. Note that VCS will not take a frozen service group offline.

Failover service group is online on another system.

The group is a failover group and is online or partially online on another system.

Recommended Action: Use the output of the command `hagrp -display service_group` to verify the value of the State attribute. Use the command `hagrp -offline` to offline the group on another system.

A critical resource faulted.

Output of the command `hagrp -display service_group` indicates that the service group has faulted.

Recommended Action: Use the command `hares -clear` to clear the fault.

Service group autodisabled.

When VCS does not know the status of a service group on a particular system, it autodisables the service group on that system. Autodisabling occurs under the following conditions:

- When the VCS engine, HAD, is not running on the system.
- When all resources within the service group are not probed on the system.

Under these conditions, all service groups that include the system in their SystemList attribute are autodisabled. *This does not apply to systems that are powered off.*

Recommended Action: Use the output of the command `hagrp -display service_group` to verify the value of the AutoDisabled attribute.

Caution: To bring a group online manually after VCS has autodisabled the group, make sure that the group is not fully or partially active on any system that has the AutoDisabled attribute set to 1 by VCS. Specifically, verify that all resources that may be corrupted by being active on multiple systems are brought down on the designated systems. Then, clear the AutoDisabled attribute for each system:

```
# hagrp -autoenable service_group -sys system
```

Service group is waiting for the resource to be brought online/taken offline.

Recommended Action: Review the IState attribute of all resources in the service group to locate which resource is waiting to go online (or which is waiting to be taken offline). Use the `hastatus` command to help identify the resource. See the engine and agent logs in `/var/VRTSvcs/log` for information on why the resource is unable to be brought online or be taken offline.

To clear this state, make sure all resources waiting to go online/offline do not bring themselves online/offline. Use the command `hagrp -flush` to clear the internal state of VCS. You can now bring the service group online or take it offline on another system.

Service group is waiting for a dependency to be met.

Recommended Action: To see which dependencies have not been met, type `hagrp -dep service_group` to view service group dependencies, or `hares -dep resource` to view resource dependencies.

Service group not fully probed.

This occurs if the agent processes have not monitored each resource in the service group. When the VCS engine, HAD, starts, it immediately “probes” to find the initial state of all of resources. (It cannot probe if the agent is not returning a value.) A service group must be probed on all systems included in the SystemList attribute before VCS attempts to bring the group online as part of AutoStart. This ensures that even if the service group was online prior to VCS being brought up, VCS will not inadvertently bring the service group online on another system.

Recommended Action: Use the output of `hagrp -display service_group` to see the value of the ProbesPending attribute for the system’s service group. (It should be zero.) To determine which resources are not probed, verify the local Probed attribute for each resource on the specified system. Zero means waiting for probe result, 1 means probed, and 2 means VCS not booted. See the engine and agent logs for information.

Troubleshooting resources

This section cites the most common problems associated with bringing resources online and taking them offline. Bold text provides a description of the problem. Recommended action is also included, where applicable.

Service group brought online due to failover.

VCS attempts to bring resources online that were already online on the failed system, or were in the process of going online. Each parent resource must wait for its child resources to be brought online before starting.

Recommended Action: Verify that the child resources are online.

Waiting for service group states.

The state of the service group prevents VCS from bringing the resource online.

Recommended Action: Review the state of the service group.

See “[Cluster and system states](#)” on page 635.

Waiting for child resources.

One or more child resources of parent resource are offline.

Recommended Action: Bring the child resources online first.

Waiting for parent resources.

One or more parent resources are online.

Recommended Action: Take the parent resources offline first.

Waiting for resource to respond.

The resource is waiting to come online or go offline, as indicated. VCS directed the agent to run an online entry point for the resource.

Recommended Action: Verify the resource's IState attribute. See the engine and agent logs in /var/VRTSvcs/log/engine_A.log and /var/VRTSvcs/log/agent_A.log for information on why the resource cannot be brought online.

Agent not running.

The resource's agent process is not running.

Recommended Action: Use `hastatus -summary` to see if the agent is listed as faulted. Restart the agent:

```
# haagent -start resource_type -sys system
```

Invalid agent argument list.

The scripts are receiving incorrect arguments.

Recommended Action: Verify that the arguments to the scripts are correct. Use the output of `hares -display resource` to see the value of the ArgListValues attribute. If the ArgList attribute was dynamically changed, stop the agent and restart it.

To stop the agent:

```
# haagent -stop resource_type -sys system
```

To restart the agent:

```
# haagent -start resource_type -sys system
```

The Monitor entry point of the disk group agent returns ONLINE even if the disk group is disabled.

This is expected agent behavior. VCS assumes that data is being read from or written to the volumes and does not declare the resource as offline. This prevents potential data corruption that could be caused by the disk group being imported on two hosts.

You can deport a disabled disk group when all I/O operations are completed or when all volumes are closed. You can then reimport the disk group to the same system.

Note: A disk group is disabled if data including the kernel log, configuration copies, or headers in the private region of a significant number of disks is invalid or inaccessible. Volumes can perform read-write operations if no changes are required to the private regions of the disks.

Troubleshooting I/O fencing

Headings indicate likely symptoms or procedures required for a solution.

Node is unable to join cluster while another node is being ejected

A cluster that is currently fencing out (ejecting) a node from the cluster prevents a new node from joining the cluster until the fencing operation is completed. The following are example messages that appear on the console for the new node:

```
...VCS FEN ERROR V-11-1-25 ... Unable to join running cluster
...VCS FEN ERROR V-11-1-25 ... since cluster is currently
fencing
...VCS FEN ERROR V-11-1-25 ... a node out of the cluster.

...VCS GAB.. Port b closed
```

If you see these messages when the new node is booting, the vxfen startup script on the node makes up to five attempts to join the cluster. If this is not sufficient to allow the node to join the cluster, restart the new node or attempt to restart vxfen driver with the command:

```
# /etc/init.d/vxfen.rc start
```

vxferststhdw fails when SCSI TEST UNIT READY command fails

If you see a message resembling:

```
Issuing SCSI TEST UNIT READY to disk reserved by other node
FAILED.
Contact the storage provider to have the hardware configuration
fixed.
```

The disk array does not support returning success for a SCSI TEST UNIT READY command when another host has the disk reserved using SCSI-3 persistent reservations. This happens with Hitachi Data Systems 99XX arrays if bit 186 of the system mode option is not enabled.

Ignore “DISK OPERATION ERROR” message during restart

When you restart the systems after you install Veritas Cluster Server and configure I/O fencing, you may safely ignore error messages resembling the following:

```
.
.
DCB47997 0506172404 T H hdisk13 DISK OPERATION ERROR
DCB47997 0506172404 T H hdisk13 DISK OPERATION ERROR
DCB47997 0506172404 T H hdisk17 DISK OPERATION ERROR
.
.
```


Removing existing keys from disks

Review the procedure to remove specific registration and reservation keys created by another node from a disk.

See [“See “Predicting VCS behavior using VCS Simulator” on page 261.” on page 244.](#)

If you want to clear all the pre-existing keys, use `vxfscclearpre` utility.

See [“About the vxfscclearpre utility” on page 340.](#)

To remove the registration and reservation keys from disk

- 1 Create a file to contain the access names of the disks:

```
# vi /tmp/disklist
```

For example:

```
/dev/rhdisk74
```

- 2 Read the existing keys:

```
# vxfsccadm -s all -f /tmp/disklist
```

The output from this command displays the key:

```
Device Name: /dev/rhdisk74
Total Number Of Keys: 1
key[0]:
[Numeric Format]: 86,70,66,69,65,68,48,50
[Character Format]: VFBEAD02
[Node Format]: Cluster ID: 48813 Node ID: 2
Node Name: unknown
```

- 3 If you know on which node the key was created, log in to that node and enter the following command:

```
# vxfsccadm -x -kA1 -f /tmp/disklist
```

The key is removed.

- 4 If you do not know on which node the key was created, follow [step 5](#) through [step 7](#) to remove the key.

- 5 Register a second key “A2” temporarily with the disk:

```
# vxfsccadm -m -k A2 -f /tmp/disklist
```

Registration completed for disk path /dev/rhdisk74

- 6 Remove the first key from the disk by preempting it with the second key:

```
# vxfsccadm -p -kA2 -f /tmp/disklist -vA1
```

```
key: A2----- preempted the key: A1----- on disk
/dev/rhdisk74
```

- 7 Remove the temporary key assigned in [step 5](#).

```
# vxfsccadm -x -kA2 -f /tmp/disklist
```

Deleted the key : [A2-----] from device /dev/rhdisk74

No registration keys exist for the disk.

System panics to prevent potential data corruption

When a node experiences a split brain condition and is ejected from the cluster, it panics and displays the following console message:

```
VXFEN:vxfen_plat_panic: Local cluster node ejected from cluster
to prevent potential data corruption.
```

How vxfen driver checks for pre-existing split brain condition

The vxfen driver functions to prevent an ejected node from rejoining the cluster after the failure of the private network links and before the private network links are repaired.

For example, suppose the cluster of system 1 and system 2 is functioning normally when the private network links are broken. Also suppose system 1 is the ejected system. When system 1 restarts before the private network links are restored, its membership configuration does not show system 2; however, when it attempts to register with the coordinator disks, it discovers system 2 is registered with them. Given this conflicting information about system 2, system 1 does not join the cluster and returns an error from vxfenconfig that resembles:

```
vxfenconfig: ERROR: There exists the potential for a preexisting
split-brain. The coordinator disks list no nodes which are in
the current membership. However, they also list nodes which are
not in the current membership.
```

```
I/O Fencing Disabled!
```

Also, the following information is displayed on the console:

```
<date> <system name> vxfen: WARNING: Potentially a preexisting
<date> <system name> split-brain.
<date> <system name> Dropping out of cluster.
<date> <system name> Refer to user documentation for steps
<date> <system name> required to clear preexisting split-brain.
<date> <system name>
<date> <system name> I/O Fencing DISABLED!
<date> <system name>
<date> <system name> gab: GAB:20032: Port b closed
```

However, the same error can occur when the private network links are working and both systems go down, system 1 restarts, and system 2 fails to come back up. From the view of the cluster from system 1, system 2 may still have the registrations on the coordinator disks.

Case 1: system 2 up, system 1 ejected (actual potential split brain)

Determine if system1 is up or not. If it is up and running, shut it down and repair the private network links to remove the split brain condition. restart system 1.

Case 2: system 2 down, system 1 ejected (apparent potential split brain)

- 1 Physically verify that system 2 is down.
- 2 Verify the systems currently registered with the coordinator disks. Use the `vxfsadm` command as follows:

```
# vxfsadm -s all -f /etc/vxfentab
```

 The output of this command identifies the keys registered with the coordinator disks.
- 3 Clear the keys on the coordinator disks as well as the data disks using the `vxfsclrpre` command.
 See [“Clearing keys after split brain using vxfsclrpre command”](#) on page 604.
- 4 Make any necessary repairs to system 2 and restart.

Cluster ID on the I/O fencing key of coordinator disk does not match the local cluster’s ID

If you accidentally assign coordinator disks of a cluster to another cluster, then the fencing driver displays an error message similar to the following when you start I/O fencing:

```
000068 06:37:33 2bdd5845 0 ... 3066 0 VXFEN WARNING V-11-1-56
Coordinator disk has key with cluster id 48813
which does not match local cluster id 57069
```

The warning implies that the local cluster with the cluster ID 57069 has keys. However, the disk also has keys for cluster with ID 48813 which indicates that nodes from the cluster with cluster id 48813 potentially use the same coordinator disk.

You can run the following commands to verify whether these disks are used by another cluster. Run the following commands on one of the nodes in the local cluster. For example, on galaxy:

```
galaxy> # lltsat -C
57069

galaxy> # cat /etc/vxfentab
/dev/vx/rdmp/disk_7
/dev/vx/rdmp/disk_8
/dev/vx/rdmp/disk_9
```

```
galaxy> # vxfenadm -s /dev/vx/rmp/disk_7
Reading SCSI Registration Keys...
Device Name: /dev/vx/rmp/disk_7
Total Number Of Keys: 1
key[0]:
  [Character Format]: VFBEAD00
  [Node Format]: Cluster ID: 48813 Node ID: 0   Node Name: unknown
```

Recommended action: You must use a unique set of coordinator disks for each cluster. If the other cluster does not use these coordinator disks, then clear the keys using the `vxfcntlpre` command before you use them as coordinator disks in the local cluster.

See [“About the vxfcntlpre utility”](#) on page 340.

Clearing keys after split brain using vxfcntlpre command

When you have encountered a split brain condition, use the `vxfcntlpre` command to remove SCSI-3 registrations and reservations on the coordinator disks as well as on the data disks in all shared disk groups.

See [“About the vxfcntlpre utility”](#) on page 340.

Registered keys are lost on the coordinator disks

If the coordinator disks lose the keys that are registered, the cluster might panic when a split-brain occurs.

Recommended Action: Use the `vxfenswap` utility to replace the coordinator disks with the same disks. The `vxfenswap` utility registers the missing keys during the disk replacement.

See [“Refreshing lost keys on coordinator disks”](#) on page 348.

Replacing defective disks when the cluster is offline

If the disk becomes defective or inoperable and you want to switch to a new diskgroup in a cluster that is offline, then perform the following procedure.

Note: In a cluster that is online, you can replace the disks using the `vxfenswap` utility.

See [“About the vxfenswap utility”](#) on page 341.

Review the following information to:

- Replace coordinator disk in the coordinator disk group
- Destroy a coordinator disk group

Note the following about the procedure:

- When adding a disk, add the disk to the disk group `vx fenceoorddg` and retest the group for support of SCSI-3 persistent reservations.
- You can destroy the coordinator disk group such that no registration keys remain on the disks. The disks can then be used elsewhere.

To remove and replace a disk in the coordinator disk group

- 1 Log in as superuser on one of the cluster nodes.
- 2 If VCS is running, shut it down:


```
# hstop -all
```

 Make sure that the port `h` is closed on all the nodes. Run the following command to verify that the port `h` is closed:


```
# gabconfig -a
```
- 3 Stop I/O fencing on all nodes:


```
# /etc/init.d/vxfen.rc stop
```

 This removes any registration keys on the disks.
- 4 Import the coordinator disk group. The file `/etc/vxfendg` includes the name of the disk group (typically, `vx fenceoorddg`) that contains the coordinator disks, so use the command:


```
# vxdg -tfc import `cat /etc/vxfendg`
```

 where:
 - t specifies that the disk group is imported only until the node restarts.
 - f specifies that the import is to be done forcibly, which is necessary if one or more disks is not accessible.
 - C specifies that any import locks are removed.
- 5 To remove disks from the disk group, use the VxVM disk administrator utility, `vx diskadm`.
 You may also destroy the existing coordinator disk group. For example:
 - Verify whether the coordinator attribute is set to on.


```
# vxdg list vx fenceoorddg | grep flags: | grep coordinator
```
 - If the coordinator attribute value is set to on, you must turn off this attribute for the coordinator disk group.


```
# vxdg -g vx fenceoordg set coordinator=off
```
 - Destroy the disk group.


```
# vxdg destroy vx fenceoorddg
```
- 6 Add the new disk to the node, initialize it as a VxVM disk, and add it to the `vx fenceoorddg` disk group.
 - On any node, create the disk group by specifying the device names:


```
vx dg init vx fenceoorddg rhdisk75 rhdisk76 rhdisk77
```

- Set the coordinator attribute value as “on” for the coordinator disk group.
`vx dg -g vxfencoord dg set coordinator=on`
 See *Veritas Cluster Server Installation Guide* for more information.
- 7 Test the recreated disk group for SCSI-3 persistent reservations compliance.
 See “[Testing the coordinator disk group using vx fentsth dw -c](#)” on page 332.
- 8 After replacing disks in a coordinator disk group, deport the disk group:
`# vx dg deport `cat /etc/vxfendg``
- 9 On each node, start the I/O fencing driver:
`# /etc/init.d/vxfen.rc start`
- 10 Verify that the I/O fencing module has started and is enabled.
`# gabconfig -a`
 Make sure that port b membership exists in the output for all nodes in the cluster.
`# vx fenadm -d`
 Make sure that I/O fencing mode is not disabled in the output.
- 11 If necessary, restart VCS on each node:
`# hastart`

The vx fenswap utility exits if rcp or scp commands are not functional

The vx fenswap utility displays an error message if rcp or scp commands are not functional:

Recommended Action: Verify whether the rcp or scp commands function properly. Make sure that you do not use echo or cat to print messages in the .bashrc file for the nodes. If the vx fenswap operation is unsuccessful, use the vx fenswap -cancel command if required to roll back any changes that the utility made.

See “[About the vx fenswap utility](#)” on page 341.

Troubleshooting server-based I/O fencing

All CP server operations and messages are logged in the /var/VRTScps/log directory in a detailed and easy to read format. The entries are sorted by date and time. The logs can be used for troubleshooting purposes or to review for any possible security issue on the system hosting the CP server.

The following files contain logs and text files that may be useful in understanding and troubleshooting a CP server:

- /var/VRTScps/log/cpsrvr_[ABC].log
- /var/VRTSat/vrtsat_broker.txt (Security related)

If the vxcperv process fails on the CP server, then review the following diagnostic files:

- /var/VRTScps/diag/FFDC_CPS_pid_vxcperv.log
- var/VRTScps/diag/stack_pid_vxcperv.txt

Note: If the vxcperv process fails on the CP server, these files are present in addition to a core file. VCS restarts the vxcperv process automatically in such situations.

Issues related to the CP server service group

If you cannot bring up the CPSSG service group after the CP server configuration, verify that the CPSSG service group and its resources are valid and properly configured in the VCS configuration.

Check the VCS engine log to see if any of the CPSSG service group resources are FAULTED.

Review the sample dependency graphs to make sure the required resources are configured correctly.

See [“About the CP server service group”](#) on page 300..

Checking the connectivity of CP server using the cpsadm command

The CP server can be tested using the cpsadm command. The following cpsadm command tests whether a CP server is up and running at a process level:

```
cpsadm -s cp_server -a ping_cps
```

where cp_server is the virtual IP address or virtual hostname on which the CP server is listening. Running commands on VCS cluster nodes requires the environment variables CPS_USERNAME and CPS_DOMAINTYPE to be set.

Troubleshooting issues related to server-based fencing on VCS nodes

The file /var/VRTSvcs/log/vxfen/vxfend_[ABC].log contains logs and text files that may be useful in understanding and/or troubleshooting fencing-related issues on a VCS cluster node.

Issues during fencing start up on VCS cluster node

The following issues may occur during fencing start up on the VCS cluster node:

The cpsadm command on the VCS cluster node gives connection error

If you receive a connection error message after issuing the `cpsadm` command on the VCS cluster, perform the following actions:

- Ensure that the CP server is reachable from all the VCS cluster nodes.
- Check that the correct CP server virtual IP or virtual hostname and port number is being used by the VCS cluster nodes (Check the `/etc/vxfenmode` file).
- Ensure that the running CP server is using the same virtual IP and port number.

Authorization failure

Authorization failure occurs when the CP server's VCS cluster nodes or users are not added in the CP server configuration. Therefore, fencing on the VCS cluster node is not allowed to access the CP server and register itself on the CP server. Fencing fails to come up if it fails to register with a majority of the coordination points. To resolve this issue, add the VCS cluster node and user in the CP server configuration and restart fencing.

See the *Veritas Cluster Server Installation Guide* for more information.

Authentication failure

If secure communication has been configured between the CP server and the VCS cluster nodes, authentication failure can occur due to the following causes:

- Symantec Product Authentication Services is not properly configured on the CP server and/or the VCS cluster.
- The CP server and the VCS cluster nodes use the same root broker but the certificate hash of the root broker is not same on the VCS cluster and the CP server. Run the following command on both the CP server and the VCS cluster to see the certificate hash:

```
# cpsat showalltrustedcreds
```

- The CP server and the VCS cluster nodes use different root brokers, and trust is not established between the authentication brokers.
- The CP server and VCS cluster do not have the same security setting. In order to configure secure communication, both the CP server and the VCS cluster must have same security setting.

In order to have the same security setting, the security parameter must have same value in the `/etc/vxcps.conf` file on CP server and in the `/etc/vxfenmode` file on the VCS cluster nodes

Preexisting split-brain

To illustrate preexisting split-brain, assume there are three CP servers acting as coordination points. One of the three CP servers then becomes inaccessible. While in this state, also one client node leaves the cluster. When the inaccessible CP server restarts, it has a stale registration from the node which left the VCS

cluster. In this case, no new nodes can join the cluster. Each node that attempts to join the cluster gets a list of registrations from the CP server. One CP server includes an extra registration (of the node which left earlier). This makes the joiner node conclude that there exists a preexisting split brain between the joiner node and the node which is represented by the stale registration. The situation is similar to that of preexisting split brain, with coordinator disks, where the problem is solved by the administrator running the `vxfcntlclearpre` command. A similar solution is required using the `cpsadm` command.

The following `cpsadm` command can be used to clear a registration on a CP server:

```
cpsadm -s cp_server -a unreg_node -c cluster_name -n node_id
```

where *cp_server* is the virtual IP address or virtual hostname on which the CP server is listening, *cluster_name* is the VCS name for the VCS cluster, and *node_id* specifies the node ID of VCS cluster node.

After removing all stale registrations, the joiner node will be able to join the cluster.

Issues during online migration of coordination points

During online migration of coordination points using the `vxfenswap` utility, the operation is automatically rolled back if a failure is encountered during validation of coordination points from all the cluster nodes.

Validation failure of the new set of coordination points can occur in the following circumstances:

- The `/etc/vxfenmode` file is not updated on all the VCS cluster nodes, because new coordination points on the node were being picked up from an old `/etc/vxfenmode` file.
- The coordination points listed in the `/etc/vxfenmode` file on the different VCS cluster nodes are not the same. If different coordination points are listed in the `/etc/vxfenmode` file on the cluster nodes, then the operation fails due to failure during the coordination point snapshot check.
- There is no network connectivity from one or more cluster nodes to the CP server.
- Cluster/nodes/users for the VCS cluster nodes have not been added to the new CP servers, thereby causing authorization failure.

Vxfen service group activity after issuing the `vxfenswap` command

After issuing the `vxfenswap` command, the Coordination Point agent reads the details of coordination points from the `vxfenconfig -l` output and starts monitoring the registrations on them.

During vxfenswap, when the vxfenmode file is being changed by the user, the Coordination Point agent does not move to FAULTED state but continues monitoring the old set of coordination points.

As long as the changes to vxfenmode file are not committed or the new set of coordination points are not re-elected in vxfenconfig -l output, the Coordination Point agent continues monitoring the old set of coordination points it read from vxfenconfig -l output in every monitor cycle.

The status of the Coordination Point agent (either ONLINE or FAULTED) depends upon the accessibility of the coordination points, the registrations on these coordination points, and the fault tolerance value.

When the changes to vxfenmode file are committed and reflected in the vxfenconfig -l output, then the Coordination Point agent reads the new set of coordination points and proceeds to monitor them in its new monitor cycle.

See [“Configuring the Coordination Point agent”](#) on page 263.

Troubleshooting server-based I/O fencing in mixed mode

The following procedure can be use to troubleshoot a mixed I/O fencing configuration (configuration using both coordinator disks and CP server for I/O fencing). This procedure involves using the following commands to obtain I/O fencing information:

- To obtain I/O fencing cluster information on the coordinator disks, run the following command on one of the cluster nodes:

```
# vxfenadm -s diskname
```

Any keys other than the valid keys used by the cluster nodes that appear in the command output are spurious keys.
- To obtain I/O fencing cluster information on the CP server, run the following command on one of the cluster nodes:

```
cpsadm -s cp_server -a list_membership -c cluster_name
```

where *cp_server* is the virtual IP address or virtual hostname on which the CP server is listening, and *cluster name* is the name of the VCS cluster.
Nodes which are not in GAB membership, but registered with CP server indicate a pre-existing network partition.
Note that when running this command on the VCS cluster nodes, you need to first export the CPS_USERNAME and CPS_DOMAINTYPE variables.
- To obtain the user name, run the following command on the CP server:

```
# cpsadm -s cp_server -a list_users
```

where *cp_server* is the virtual IP address or virtual hostname on which the CP server is listening.
The CPS_DOMAINTYPE value is vx.
The following are export variable command examples:

```
# export CPS_USERNAME=_HA_VCS_test-system@HA_SERVICES@test-
system.symantec.com
# export CPS_DOMAINTYPE=vx
```

Once a pre-existing network partition is detected using the above commands, all spurious keys on the coordinator disks or CP server must be removed by the administrator.

Troubleshooting mixed I/O fencing configuration (coordinator disks and CP server)

- 1 Review the current I/O fencing configuration by accessing and viewing the information in the vxfenmode file.

Enter the following command on one of the VCS cluster nodes:

```
# cat /etc/vxfenmode
```

The command returns:

```
vxfen_mode=customized
vxfen_mechanism=cps
scsi3_disk_policy=dmp
security=0
cps1=[10.140.94.101]:14250
vxfgndg=vxfencoordg
```

- 2 Review the I/O fencing cluster information.

Enter the following command on one of the cluster nodes:

```
# vxfenadm -d
```

The command returns:

```
I/O Fencing Cluster Information:
=====
Fencing Protocol Version: 201
Fencing Mode: Customized
Fencing Mechanism: cps
Cluster Members:
* 0 (galaxy)
  1 (nebula)
RFSM State Information:
node 0 in state 8 (running)
node 1 in state 8 (running)
```

- 3 Review the SCSI registration keys for the coordinator disks used in the I/O fencing configuration.

Enter the vxfenadm -s command on each of the VCS cluster nodes.

```
# vxfenadm -s /dev/vx/rmdp/3pardata0_190
```

The command returns:

```
Device Name: /dev/vx/rmdp/3pardata0_190
Total Number Of Keys: 2
key[0]:
[Numeric Format]: 86,70,66,69,65,68,48,48
[Character Format]: VFBEAD00
```

```
[Node Format]: Cluster ID: 57069 Node ID: 0 Node Name: galaxy
key[1]:
[Numeric Format]: 86,70,66,69,65,68,48,49
[Character Format]: VFBEAD01
* [Node Format]: Cluster ID: 57069 Node ID: 1 Node Name: nebula
```

Run the command on another node:

```
# vxfenadm -s /dev/vx/rdmp/3pardata0_191
```

The command returns:

```
Device Name: /dev/vx/rdmp/3pardata0_191
Total Number Of Keys: 2
key[0]:
[Numeric Format]: 86,70,66,69,65,68,48,48
[Character Format]: VFBEAD00
[Node Format]: Cluster ID: 57069 Node ID: 0 Node Name: galaxy
key[1]:
[Numeric Format]: 86,70,66,69,65,68,48,49
[Character Format]: VFBEAD01
* [Node Format]: Cluster ID: 57069 Node ID: 1 Node Name: nebula
```

- 4 Review the CP server information about the cluster nodes. On the CPS server, run the `cpsadm list nodes` command to review a list of nodes in the cluster.

```
# cpsadm -s cp_server -a list_nodes
```

where *cp_server* is the virtual IP address or virtual hostname on which the CP server is listening.

- 5 Review the CP server list membership. On the CP server, run the following command to review the list membership.

```
# cpsadm -s cp_server -a list_membership -c cluster_name
```

where *cp_server* is the virtual IP address or virtual hostname on which the CP server is listening, and *cluster_name* is the name of the VCS cluster.

For example:

```
# cpsadm -s 10.140.94.101 -a list_membership -c gl-ss2
```

The command returns:

```
List of registered nodes: 0 1
```

Checking keys on coordination points when `vxfen_mechanism` value is set to `cps`

When I/O fencing is configured in customized mode and the `vxfen_mechanism` value is set to `cps`, the recommended way of reading keys from the coordination points (coordinator disks and CP servers) is as follows:

- For coordinator disks, the disks can be put in a file and then information about them supplied to the `vxfenadm` command.

For example:

```
# vxfenadm -s all -f file_name
```

- For CP servers, the cpsadm command can be used to obtain the membership of the VCS cluster.

For example:

```
# cpsadm -s cp_server -a list_membership -c cluster_name
```

Where *cp_server* is the virtual IP address or virtual hostname on which CP server is configured, and *cluster_name* is the name of the VCS cluster.

Troubleshooting notification

Occasionally you may encounter problems when using VCS notification. This section cites the most common problems and the recommended actions. Bold text provides a description of the problem.

Notifier is configured but traps are not seen on SNMP console.

Recommended Action: Verify the version of SNMP traps supported by the console: VCS notifier sends SNMP v2.0 traps. If you are using HP OpenView Network Node Manager as the SNMP, verify events for VCS are configured using xnmevents. You may also try restarting the OpenView daemon (ovw) if, after merging VCS events in vcs_trapd, the events are not listed in the OpenView Network Node Manager Event configuration.

By default, notifier assumes the community string is public. If your SNMP console was configured with a different community, reconfigure it according to the notifier configuration. See the *Veritas Cluster Server Bundled Agents Reference Guide* for more information on NotifierMgr.

Troubleshooting VCS configuration backup and restore

This section cites the problem you may encounter when using the hasnap command to backup and restore VCS configuration files.

Error connecting to remote nodes in the cluster.

The hasnap command is a distributed command in the sense that it tries to backup and restore files from all cluster nodes in a single session. It needs to establish connection with all cluster nodes from the node where the command is executed. The connection may fail for one of the following reasons:

- The `hasnap` command retrieves the list of cluster nodes from the `llthosts` configuration file. However, the node names in this file may not always be DNS resolvable, in which case the command cannot establish connection with the remote nodes.
Recommended Action: For each node in the cluster, map the VCS node names to the actual DNS-resolvable names using the `hasnap` configuration file `/opt/VRTSvcs/cutil/conf/vcsmappings.properties`.
- The `hasnap` command uses the VCS Command Server Daemon running on the remote nodes to establish connection. The connection fails if the Daemon is not running on the remote node.
Recommended Action: Verify the VCS Command Server Daemon is running on all cluster nodes. Start it by running the following command:

```
# /opt/VRTSvcs/bin/CmdServer
```
- The remote node might be currently down or unreachable.
Recommended Action: Run the `hasnap` command again after the bringing the remote node online.

Troubleshooting and recovery for global clusters

This section describes the concept of disaster declaration and provides troubleshooting tips for configurations using global clusters.

Disaster declaration

When a cluster in a global cluster transitions to the `FAULTED` state because it can no longer be contacted, failover executions depend on whether the cause was due to a split-brain, temporary outage, or a permanent disaster at the remote cluster.

If you choose to take action on the failure of a cluster in a global cluster, VCS prompts you to declare the type of failure.

- *Disaster*, implying permanent loss of the primary data center
- *Outage*, implying the primary may return to its current form in some time
- *Disconnect*, implying a split-brain condition; both clusters are up, but the link between them is broken
- *Replica*, implying that data on the takeover target has been made consistent from a backup source and that the RVGPrimary can initiate a takeover when the service group is brought online. This option applies to VVR environments only.

You can select the groups to be failed over to the local cluster, in which case VCS brings the selected groups online on a node based on the group's `FailOverPolicy` attribute. It also marks the groups as being offline in the other cluster. If you do not select any service groups to fail over, VCS takes no action except implicitly marking the service groups as offline on the downed cluster.

Lost heartbeats and the inquiry mechanism

The loss of internal and all external heartbeats between any two clusters indicates that the remote cluster is faulted, or that all communication links between the two clusters are broken (a wide-area split-brain).

VCS queries clusters to confirm the remote cluster to which heartbeats have been lost is truly down. This mechanism is referred to as inquiry. If in a two-cluster configuration a connector loses all heartbeats to the other connector, it must consider the remote cluster faulted. If there are more than two clusters and a connector loses all heartbeats to a second cluster, it queries the remaining connectors before declaring the cluster faulted. If the other connectors view the cluster as running, the querying connector transitions the cluster to the `UNKNOWN` state, a process that minimizes false cluster faults. If all connectors

report that the cluster is faulted, the querying connector also considers it faulted and transitions the remote cluster state to `FAULTED`.

VCS alerts

VCS alerts are identified by the alert ID, which is comprised of the following elements:

- `alert_type`—The type of the alert
See “[Types of alerts](#).”
- `cluster`—The cluster on which the alert was generated
- `system`—The system on which this alert was generated
- `object`—The name of the VCS object for which this alert was generated.
This could be a cluster or a service group.

Alerts are generated in the following format:

`alert_type-cluster-system-object`

For example:

`GNOFAILA-Cluster1-oracle_grp`

This is an alert of type `GNOFAILA` generated on cluster `Cluster1` for the service group `oracle_grp`.

Types of alerts

VCS generates the following types of alerts.

- `CFAULT`—Indicates that a cluster has faulted
- `GNOFAILA`—Indicates that a global group is unable to fail over within the cluster where it was online. This alert is displayed if the `ClusterFailOverPolicy` attribute is set to `Manual` and the wide-area connector (`wac`) is properly configured and running at the time of the fault.
- `GNOFAIL`—Indicates that a global group is unable to fail over to any system within the cluster or in a remote cluster.
Some reasons why a global group may not be able to fail over to a remote cluster:
 - The `ClusterFailOverPolicy` is set to either `Auto` or `Connected` and VCS is unable to determine a valid remote cluster to which to automatically fail the group over.
 - The `ClusterFailOverPolicy` attribute is set to `Connected` and the cluster in which the group has faulted cannot communicate with one or more remote clusters in the group's `ClusterList`.

- The wide-area connector (wac) is not online or is incorrectly configured in the cluster in which the group has faulted

Managing alerts

Alerts require user intervention. You can respond to an alert in the following ways:

- If the reason for the alert can be ignored, use the Alerts dialog box in the Java console or the `haalert` command to delete the alert. You must provide a comment as to why you are deleting the alert; VCS logs the comment to engine log.
- Take an action on administrative alerts that have actions associated with them.
- VCS deletes or *negates* some alerts when a negating event for the alert occurs.

An administrative alert will continue to live if none of the above actions are performed and the VCS engine (HAD) is running on at least one node in the cluster. If HAD is not running on any node in the cluster, the administrative alert is lost.

Actions associated with alerts

This section describes the actions you can perform on the following types of alerts:

- CFAULT—When the alert is presented, clicking **Take Action** guides you through the process of failing over the global groups that were online in the cluster before the cluster faulted.
- GNOFAILA—When the alert is presented, clicking **Take Action** guides you through the process of failing over the global group to a remote cluster on which the group is configured to run.
- GNOFAIL—There are no associated actions provided by the consoles for this alert

Negating events

VCS deletes a CFAULT alert when the faulted cluster goes back to the running state

VCS deletes the GNOFAILA and GNOFAIL alerts in response to the following events:

- The faulted group's state changes from FAULTED to ONLINE.

- The group's fault is cleared.
- The group is deleted from the cluster where alert was generated.

Concurrency violation at startup

VCS may report a concurrency violation when you add a cluster to the ClusterList of the service group. A concurrency violation means that the service group is online on two nodes simultaneously.

Recommended Action: Verify the state of the service group in each cluster before making the service group global.

Troubleshooting the steward process

When you start the steward, it blocks the command prompt and prints messages to the standard output. To stop the steward, run the following command from a different command prompt of the same system:

If the steward is running in secure mode	<code>steward -stop - secure</code>
If the steward is not running in secure mode	<code>steward -stop</code>

In addition to the standard output, the steward can log to its own log files:

- `steward_A.log`
- `steward-err_A.log`

Use the `tststew` utility to verify that:

- The steward process is running
- The steward process is sending the right response

Troubleshooting licensing

This section cites problems you may encounter with VCS licensing. It provides instructions on how to validate license keys and lists the error messages associated with licensing.

Validating license keys

The `installvcs` script handles most license key validations. However, if you install a VCS key outside of `installvcs` (using `vxlicinst`, for example), you can validate the key using the procedure described below.

- 1 The `vxlicinst` command handles some of the basic validations:

node lock: Ensures that you are installing a node-locked key on the correct system

demo hard end date: Ensures that you are not installing an expired demo key

- 2 Run the `vxlicrep` command to make sure a VCS key is installed on the system. Review the output of the command.

```
Veritas License Manager vxlicrep utility version <version>
...
system
License Key                = XXXX-XXXX-XXXX-XXXX-XXXX-XXXX
Product Name               = Veritas Cluster Server
License Type               = PERMANENT
OEM ID                     = 4095
Features :=
Platform                   = <platform>
Version                    = <version>
Tier                       = Unused
Reserved                   = 0
Mode                       = VCS
Global Cluster Option= Enabled
```

- 3 Look for the following in the command output:
 Make sure the *Product Name* lists the name of your purchased component, for example, Veritas Cluster Server. If the command output does not return the product name, you do not have a VCS key installed.
 If the output shows the *License Type* for a VCS key as DEMO, ensure that the Demo End Date does not display a past date.
 Make sure the *Mode* attribute displays the correct value.
 If you have purchased a license key for the Global Cluster Option, make sure its status is Enabled.
- 4 Start VCS. If HAD rejects a license key, see the licensing error message at the end of the `engine_A` log file.

Licensing error messages

This section lists the error messages associated with licensing. These messages are logged to the file `/var/VRTSvcs/log/engine_A.log`.

[Licensing] Insufficient memory to perform operation

The system does not have adequate resources to perform licensing operations.

[Licensing] No valid VCS license keys were found

No valid VCS keys were found on the system.

[Licensing] Unable to find a valid base VCS license key

No valid base VCS key was found on the system.

[Licensing] License key can not be used on this OS platform

This message indicates that the license key was meant for a different platform. For example, a license key meant for Windows is used on a Solaris platform.

[Licensing] VCS evaluation period has expired

The VCS base demo key has expired

[Licensing] License key can not be used on this system

Indicates that you have installed a key that was meant for a different system (i.e. node-locked keys)

[Licensing] Unable to initialize the licensing framework

This is a VCS internal message. Call Veritas Technical Support.

[Licensing] QuickStart is not supported in this release

VCS QuickStart is not supported in this version of VCS.

[Licensing] Your evaluation period for the feature has expired. This feature will not be enabled the next time VCS starts

The evaluation period for the specified VCS feature has expired.

Appendixes

- [Appendix A, “VCS user privileges—administration matrices” on page 623](#)
- [Appendix B, “VCS commands: quick reference” on page 631](#)
- [Appendix C, “Cluster and system states” on page 635](#)
- [Appendix D, “VCS attributes” on page 643](#)
- [Appendix E, “Configuring VCS for partitions” on page 699](#)
- [Appendix F, “Accessibility and VCS” on page 713](#)

VCS user privileges— administration matrices

- [About administration matrices](#)
- [Administration matrices](#)

About administration matrices

In general, users with Guest privileges can execute the following command options: -display, -state, and -value.

Users with privileges for Group Operator and Cluster Operator can execute the following options: -online, -offline, and -switch.

Users with Group Administrator and Cluster Administrator privileges can execute the following options -add, -delete, and -modify.

See “[About the VCS user privilege model](#)” on page 71.

Administration matrices

Review the matrices in the following section to determine which command options can be executed within a specific user role. Checkmarks denote the command and option can be executed. A dash indicates they cannot.

Agent Operations (haagent)

Table A-1 lists agent operations and required privileges.

Table A-1 User privileges for agent operations					
Agent Operation	Guest	Group Operator	Group Admin.	Cluster Operator	Cluster Admin.
Start agent	–	–	–	✓	✓
Stop agent	–	–	–	✓	✓
Display info	✓	✓	✓	✓	✓
List agents	✓	✓	✓	✓	✓

Attribute Operations (haattr)

Table A-2 lists attribute operations and required privileges.

Table A-2 User privileges for attribute operations					
Attribute Operations	Guest	Group Operator	Group Admin.	Cluster Operator	Cluster Admin.
Add	–	–	–	–	✓
Change default value	–	–	–	–	✓
Delete	–	–	–	–	✓
Display	✓	✓	✓	✓	✓

Cluster Operations (haclus, haconf)

Table A-3 lists cluster operations and required privileges.

Table A-3 User privileges for cluster operations

Cluster Operations	Cluster Guest	Group Operator	Group Admin.	Cluster Operator	Cluster Admin.
Display	✓	✓	✓	✓	✓
Modify	–	–	–	–	✓
Add	–	–	–	–	✓
Delete	–	–	–	–	✓
Declare	–	–	–	✓	✓
View state or status	✓	✓	✓	✓	✓
Update license					✓
Make configuration read-write	–	–	✓	–	✓
Save configuration	–	–	✓	–	✓
Make configuration read-only	–	–	✓	–	✓

Service group operations (hagrp)

Table A-4 lists service group operations and required privileges.

Table A-4 User privileges for service group operations

Service Group Operations	Guest	Group Operator	Group Admin.	Cluster Operator	Cluster Admin.
Add and delete	–	–	–	–	✓
Link and unlink	–	–	–	–	✓
Clear	–	✓	✓	✓	✓
Bring online	–	✓	✓	✓	✓
Take offline	–	✓	✓	✓	✓
View state	✓	✓	✓	✓	✓
Switch	–	✓	✓	✓	✓
Freeze/unfreeze	–	✓	✓	✓	✓
Freeze/unfreeze persistent	–	–	✓	–	✓
Enable	–	–	✓	–	✓
Disable	–	–	✓	–	✓
Modify	–	–	✓	–	✓
Display	✓	✓	✓	✓	✓
View dependencies	✓	✓	✓	✓	✓
View resources	✓	✓	✓	✓	✓
List	✓	✓	✓	✓	✓
Enable resources	–	–	✓	–	✓
Disable resources	–	–	✓	–	✓
Flush	–	✓	✓	✓	✓
Autoenable	–	✓	✓	✓	✓
Ignore	–	✓	✓	✓	✓

Heartbeat operations (hahb)

[Table A-5](#) lists heartbeat operations and required privileges.

Table A-5 User privileges for heartbeat operations

Heartbeat Operations	Guest	Group Operator	Group Admin.	Cluster Operator	Cluster Admin.
Add	–	–	–	–	✓
Delete	–	–	–	–	✓
Make local	–	–	–	–	✓
Make global	–	–	–	–	✓
Display	✓	✓	✓	✓	✓
View state	✓	✓	✓	✓	✓
List	✓	✓	✓	✓	✓

Log operations (halog)

[Table A-6](#) lists log operations and required privileges.

Table A-6 User privileges for log operations

Log operations	Guest	Group Operator	Group Admin.	Cluster Operator	Cluster Admin.
Enable debug tags	–	–	–	–	✓
Delete debug tags	–	–	–	–	✓
Add messages to log file	–	–	–	✓	✓
Display	✓	✓	✓	✓	✓
Display log file info	✓	✓	✓	✓	✓

Resource operations (hares)

Table A-7 lists resource operations and required privileges.

Table A-7 User privileges for resource operations

Resource operations	Guest	Group Operator	Group Admin.	Cluster Operator	Cluster Admin.
Add	–	–	✓	–	✓
Delete	–	–	✓	–	✓
Make local	–	–	✓	–	✓
Make global	–	–	✓	–	✓
Link and unlink	–	–	✓	–	✓
Clear	–	✓	✓	✓	✓
Bring online	–	✓	✓	✓	✓
Take offline	–	✓	✓	✓	✓
Modify	–	–	✓	–	✓
View state	✓	✓	✓	✓	✓
Display	✓	✓	✓	✓	✓
View dependencies	✓	✓	✓	✓	✓
List, Value	✓	✓	✓	✓	✓
Probe	–	✓	✓	✓	✓
Override	–	–	✓	–	✓
Remove overrides	–	–	✓	–	✓
Run an action	–	✓	✓	✓	✓
Refresh info	–	✓	✓	✓	✓
Flush info	–	✓	✓	✓	✓

System operations (hasys)

[Table A-8](#) lists system operations and required privileges.

Table A-8 User privileges for system operations

System operations	Guest	Group Operator	Group Admin.	Cluster Operator	Cluster Admin.
Add	–	–	–	–	✓
Delete	–	–	–	–	✓
Freeze and unfreeze	–	–	–	✓	✓
Freeze and unfreeze persistent	–	–	–	–	✓
Freeze and evacuate	–	–	–	–	✓
Display	✓	✓	✓	✓	✓
Start forcibly	–	–	–	–	✓
Modify	–	–	–	–	✓
View state	✓	✓	✓	✓	✓
List	✓	✓	✓	✓	✓
Update license	–	–	–	–	✓

Resource type operations (hatype)

Table A-9 lists resource type operations and required privileges.

Table A-9 User privileges for resource type operations

Resource type operations	Guest	Group Operator	Group Admin.	Cluster Operator	Cluster Admin.
Add	–	–	–	–	✓
Delete	–	–	–	–	✓
Display	✓	✓	✓	✓	✓
View resources	✓	✓	✓	✓	✓
Modify	–	–	–	–	✓
List	✓	✓	✓	✓	✓

User operations (hauser)

Table A-10 lists user operations and required privileges.

Table A-10 User privileges for user operations

User operations	Guest	Group Operator	Group Admin.	Cluster Operator	Cluster Admin.
Add	–	–	–	–	✓
Delete	–	–	–	–	✓
Update	–	✓	✓	✓	✓
Display	✓	✓	✓	✓	✓
List	✓	✓	✓	✓	✓
Modify privileges	–	–	✓	–	✓

VCS commands: quick reference

- [About this quick reference for VCS commands](#)
- [VCS command line reference](#)

About this quick reference for VCS commands

This section lists commonly-used VCS commands and options. For more information about VCS commands and available options, see the man pages associated with VCS commands.

VCS command line reference

Cluster operations	Command line
Start VCS	hastart [-force] -stale <i>system_name</i> hasys -force <i>system_name</i>
Stop VCS	hastop -local [-force -evacuate] hastop -sys <i>system_name</i> [-force -evacuate] hastop -all [-force]
Enable/disable read-write access to VCS configuration	haconf -makerw haconf -dump -makero
Add user	hauser -add <i>user_name</i>

Service group and resource operations	Command line
Configure service groups	hagrp -add -delete <i>group_name</i> hagrp -modify <i>group</i> <i>attribute</i> <i>value</i> hagrp -link <i>parent_group</i> <i>child_group</i> <i>dependency</i> <i>firm</i> <i>soft</i>
Configure resources	hares -add <i>resource</i> <i>type</i> <i>group_name</i> hares -delete <i>resource_name</i> hares -modify <i>resource</i> <i>attribute</i> <i>value</i> hares -link <i>parent_resource</i> <i>child_resource</i>
Configure agents or resource types	hatype -modify <i>type</i> <i>attribute</i> <i>value</i>
Bring service groups online and take them offline	hagrp -online <i>service_group</i> -sys <i>system_name</i> hagrp -offline <i>service_group</i> -sys <i>system_name</i>
Bring resources online and take them offline	hares -online <i>resource_name</i> -sys <i>system_name</i> hares -offline <i>resource_name</i> -sys <i>system_name</i>
Freezing/unfreezing service groups	hagrp -freeze <i>group_name</i> [-persistent] hagrp -unfreeze <i>group_name</i> [-persistent]
Freezing/unfreezing systems	hasys -freeze [-persistent][-evacuate] <i>system_name</i> hasys -unfreeze [-persistent] <i>system_name</i>

Status and verification	Command line
Cluster status	<code>hastatus -summary</code>
LLT status/verification	<code>lltconfig</code> <code>lltconfig -a list</code> <code>lltstat</code> <code>lltstat -nvv</code>
GAB status/verification	<code>gabconfig -a</code>
Communication	Command line
Starting and stopping LLT	<code>lltconfig -c</code> <code>lltconfig -U</code>
Starting and stopping GAB	<code>gabconfig -c -n <i>seed_number</i></code> <code>gabconfig -U</code>
System operations	Command line
List systems in a cluster	<code>hasys -list</code>
Retrieve detailed information about each system	<code>hasys -display <i>system_name</i></code>

Cluster and system states

- Remote cluster states
- System states

Remote cluster states

In global clusters, the “health” of the remote clusters is monitored and maintained by the wide-area connector process. The connector process uses heartbeats, such as ICMP, to monitor the state of remote clusters. The state is then communicated to HAD, which then uses the information to take appropriate action when required. For example, when a cluster is shut down gracefully, the connector transitions its local cluster state to `EXITING` and notifies the remote clusters of the new state. When the cluster exits and the remote connectors lose their TCP/IP connection to it, each remote connector transitions their view of the cluster to `EXITED`.

To enable wide-area network heartbeats, the wide-area connector process must be up and running. For wide-area connectors to connect to remote clusters, at least one heartbeat to the specified cluster must report the state as `ALIVE`.

There are three heartbeat states for remote clusters: `HBUNKNOWN`, `HBALIVE`, and `HBDEAD`.

[Table C-11](#) on page 637 provides a list of VCS remote cluster states and their descriptions.

See “[Examples of system state transitions](#)” on page 641.

Table C-11 VCS state definitions

State	Definition
INIT	The initial state of the cluster. This is the default state.
BUILD	The local cluster is receiving the initial snapshot from the remote cluster.
RUNNING	Indicates the remote cluster is running and connected to the local cluster.
LOST_HB	The connector process on the local cluster is not receiving heartbeats from the remote cluster
LOST_CONN	The connector process on the local cluster has lost the TCP/IP connection to the remote cluster.
UNKNOWN	The connector process on the local cluster determines the remote cluster is down, but another remote cluster sends a response indicating otherwise.
FAULTED	The remote cluster is down.
EXITING	The remote cluster is exiting gracefully.
EXITED	The remote cluster exited gracefully.
INQUIRY	The connector process on the local cluster is querying other clusters on which heartbeats were lost.
TRANSITIONING	The connector process on the remote cluster is failing over to another node in the cluster.

Examples of cluster state transitions

- If a remote cluster joins the global cluster configuration, the other clusters in the configuration transition their “view” of the remote cluster to the `RUNNING` state:
INIT -> BUILD -> RUNNING
- If a cluster loses all heartbeats to a remote cluster in the `RUNNING` state, inquiries are sent. If all inquiry responses indicate the remote cluster is actually down, the cluster transitions the remote cluster state to `FAULTED`:
RUNNING -> LOST_HB -> INQUIRY -> FAULTED
- If at least one response does not indicate the cluster is down, the cluster transitions the remote cluster state to `UNKNOWN`:
RUNNING -> LOST_HB -> INQUIRY -> UNKNOWN
- When the ClusterService service group, which maintains the connector process as highly available, fails over to another system in the cluster, the remote clusters transition their view of that cluster to `TRANSITIONING`, then back to `RUNNING` after the failover is successful:
RUNNING -> TRANSITIONING -> BUILD -> RUNNING
- When a remote cluster in a `RUNNING` state is stopped (by taking the ClusterService service group offline), the remote cluster transitions to `EXITED`:
RUNNING -> EXITING -> EXITED

System states

Whenever the VCS engine is running on a system, it is in one of the states described in the table below. States indicate a system’s current mode of operation. When the engine is started on a new system, it identifies the other systems available in the cluster and their states of operation. If a cluster system is in the state of `RUNNING`, the new system retrieves the configuration information from that system. Changes made to the configuration while it is being retrieved are applied to the new system before it enters the `RUNNING` state. If no other systems are up and in the state of `RUNNING` or `ADMIN_WAIT`, and the new system has a configuration that is not invalid, the engine transitions to the state `LOCAL_BUILD`, and builds the configuration from disk. If the configuration is invalid, the system transitions to the state of `STALE_ADMIN_WAIT`.

[Table C-12](#) on page 639 provides a list of VCS system states and their descriptions.

See [“Examples of system state transitions”](#) on page 641.

Table C-12 VCS system states

State	Definition
ADMIN_WAIT	The running configuration was lost. A system transitions into this state for the following reasons: <ul style="list-style-type: none">■ The last system in the RUNNING configuration leaves the cluster before another system takes a snapshot of its configuration and transitions to the RUNNING state.■ A system in LOCAL_BUILD state tries to build the configuration from disk and receives an unexpected error from hacf indicating the configuration is invalid.
CURRENT_DISCOVER_WAIT	The system has joined the cluster and its configuration file is valid. The system is waiting for information from other systems before it determines how to transition to another state.
CURRENT_PEER_WAIT	The system has a valid configuration file and another system is doing a build from disk (LOCAL_BUILD). When its peer finishes the build, this system transitions to the state REMOTE_BUILD.
EXITING	The system is leaving the cluster.
EXITED	The system has left the cluster.
EXITING_FORCIBLY	An <code>hastop -force</code> command has forced the system to leave the cluster.
FAULTED	The system has left the cluster unexpectedly.
INITING	The system has joined the cluster. This is the initial state for all systems.
LEAVING	The system is leaving the cluster gracefully. When the agents have been stopped, and when the current configuration is written to disk, the system transitions to EXITING.
LOCAL_BUILD	The system is building the running configuration from the disk configuration.
REMOTE_BUILD	The system is building a running configuration that it obtained from a peer in a RUNNING state.

Table C-12 VCS system states

State	Definition
RUNNING	The system is an active member of the cluster.
STALE_ADMIN_WAIT	<p>The system has an invalid configuration and there is no other system in the state of RUNNING from which to retrieve a configuration. If a system with a valid configuration is started, that system enters the LOCAL_BUILD state.</p> <p>Systems in STALE_ADMIN_WAIT transition to STALE_PEER_WAIT.</p>
STALE_DISCOVER_WAIT	The system has joined the cluster with an invalid configuration file. It is waiting for information from any of its peers before determining how to transition to another state.
STALE_PEER_WAIT	The system has an invalid configuration file and another system is doing a build from disk (LOCAL_BUILD). When its peer finishes the build, this system transitions to the state REMOTE_BUILD.
UNKNOWN	The system has not joined the cluster because it does not have a system entry in the configuration.

Examples of system state transitions

- If VCS is started on a system, and if that system is the only one in the cluster with a valid configuration, the system transitions to the RUNNING state:
INITING -> CURRENT_DISCOVER_WAIT -> LOCAL_BUILD -> RUNNING
- If VCS is started on a system with a valid configuration file, and if at least one other system is already in the RUNNING state, the new system transitions to the RUNNING state:
INITING -> CURRENT_DISCOVER_WAIT -> REMOTE_BUILD -> RUNNING
- If VCS is started on a system with an invalid configuration file, and if at least one other system is already in the RUNNING state, the new system transitions to the RUNNING state:
INITING -> STALE_DISCOVER_WAIT -> REMOTE_BUILD -> RUNNING
- If VCS is started on a system with an invalid configuration file, and if all other systems are in STALE_ADMIN_WAIT state, the system transitions to the STALE_ADMIN_WAIT state as shown below. A system stays in this state until another system with a valid configuration file is started.
INITING -> STALE_DISCOVER_WAIT -> STALE_ADMIN_WAIT
- If VCS is started on a system with a valid configuration file, and if other systems are in the ADMIN_WAIT state, the new system transitions to the ADMIN_WAIT state.
INITING -> CURRENT_DISCOVER_WAIT -> ADMIN_WAIT
- If VCS is started on a system with an invalid configuration file, and if other systems are in the ADMIN_WAIT state, the new system transitions to the ADMIN_WAIT state.
INITING -> STALE_DISCOVER_WAIT -> ADMIN_WAIT
- When a system in RUNNING state is stopped with the `hastop` command, it transitions to the EXITED state as shown below. During the LEAVING state, any online system resources are taken offline. When all of the system's resources are taken offline and the agents are stopped, the system transitions to the EXITING state, then EXITED.
RUNNING -> LEAVING -> EXITING -> EXITED

VCS attributes

- [About attributes and their definitions](#)
- [Resource attributes](#)
- [Resource type attributes](#)
- [Service group attributes](#)
- [System attributes](#)
- [Cluster attributes](#)
- [Heartbeat attributes \(for global clusters\)](#)

About attributes and their definitions

- In addition to the attributes listed in this appendix, see the *Veritas Cluster Server Agent Developer's Guide*.
- You can modify the values of attributes labelled user-defined from the command line or graphical user interface, or by manually modifying the main.cf configuration file. You can change the default values to better suit your environment and enhance performance.
When changing the values of attributes, be aware that VCS attributes interact with each other. After changing the value of an attribute, observe the cluster systems to confirm that unexpected behavior does not impair performance.
 - VCS sets the values of attributes that are labeled, “system use only.” These are read-only attributes. They contain important information about the state of the cluster.
 - Agents set the values of attributes labeled “agent-defined.” These are read-only attributes.
- See “[About VCS attributes](#)” on page 57.

Resource attributes

[Table D-1](#) lists resource attributes.

Table D-1 Resource attributes

Resource Attributes	Description
ArgListValues (agent-defined)	List of arguments passed to the resource's agent on each system.This attribute is resource- and system-specific, meaning that the list of values passed to the agent depend on which system and resource they are intended. <ul style="list-style-type: none">■ Type and dimension: string-vector■ Default: non-applicable.
AutoStart (user-defined)	Indicates the resource is brought online when the service group is brought online. <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 1

Table D-1 Resource attributes

Resource Attributes	Description
ComputeStats (user-defined)	<p>Indicates to agent framework whether or not to calculate the resource's monitor statistics.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0
ConfidenceLevel (agent-defined)	<p>Indicates the level of confidence in an online resource. Values range from 0–100. Note that some VCS agents may not take advantage of this attribute and may always set it to 0. Set the level to 100 if the attribute is not used.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 0
ContainerOpts (system use only)	<p>The resource level attribute ContainerOpts specifies information that passes to the agent that controls the resources. These values are only effective when you set the ContainerInfo service group attribute.</p> <ul style="list-style-type: none">■ RunInContainer When the value of the RunInContainer key is 1, the agent function (entry point) for that resource runs inside of the local container. When the value of the RunInContainer key is 0, the agent function (entry point) for that resource runs outside the local container (in the global environment).■ PassCInfo When the value of the PassCInfo key is 1, the agent function receives the container information that is defined in the service group's ContainerInfo attribute. An example use of this value is to pass the name of the container to the agent. When the value of the PassCInfo key is 0, the agent function does not receive the container information that is defined in the service group's ContainerInfo attribute.
Critical (user-defined)	<p>Indicates whether a fault of this resource should trigger a failover of the entire group or not. If Critical is 0, the resource fault will not cause group failover.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 1

Table D-1 Resource attributes

Resource Attributes	Description
Enabled (user-defined)	<p>Indicates agents monitor the resource.</p> <p>If a resource is created dynamically while VCS is running, you must enable the resource before VCS monitors it. For more information on how to add or enable resources, see the chapters on administering VCS from the command line and graphical user interfaces.</p> <p>When Enabled is set to 0, it implies a disabled resource.</p> <p>See “Troubleshooting VCS startup” on page 593.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: If you specify the resource in main.cf prior to starting VCS, the default value for this attribute is 1, otherwise it is 0.
Flags (system use only)	<p>Provides additional information for the state of a resource. Primarily this attribute raises flags pertaining to the resource. Values:</p> <p>NORMAL—Standard working order.</p> <p>RESTARTING —The resource faulted and that the agent is attempting to restart the resource on the same system.</p> <p>STATE UNKNOWN—The latest monitor call by the agent could not determine if the resource was online or offline.</p> <p>MONITOR TIMEDOUT —The latest monitor call by the agent was terminated because it exceeded the maximum time specified by the static attribute MonitorTimeout.</p> <p>UNABLE TO OFFLINE—The agent attempted to offline the resource but the resource did not go offline. This flag is also set when a resource faults and the clean function completes successfully, but the subsequent monitor hangs or is unable to determine resource status.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable.
Group (system use only)	<p>String name of the service group to which the resource belongs.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: Not applicable.

Table D-1 Resource attributes

Resource Attributes	Description
IState (system use only)	<p>The internal state of a resource. In addition to the State attribute, this attribute shows to which state the resource is transitioning.</p> <p>Values:</p> <p>NOT WAITING—Resource is not in transition.</p> <p>WAITING TO GO ONLINE—Agent notified to bring the resource online but procedure not yet complete.</p> <p>WAITING FOR CHILDREN ONLINE—Resource to be brought online, but resource depends on at least one offline resource. Resource transitions to WAITING TO GO ONLINE when all children are online.</p> <p>WAITING TO GO OFFLINE—Agent notified to take the resource offline but procedure not yet complete.</p> <p>WAITING TO GO OFFLINE (propagate)—Same as above, but when completed the resource's children will also be offline.</p> <p>WAITING TO GO ONLINE (reverse)—Resource waiting to be brought online, but when it is online it attempts to go offline. Typically this is the result of issuing an offline command while resource was waiting to go online.</p> <p>WAITING TO GO OFFLINE (reverse/propagate)—Same as above, but resource propagates the offline operation.</p> <p>Note: Although this attribute accepts integer types, the command line indicates the text representations.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 1 NOT WAITING
LastOnline (system use only)	<p>Indicates the system name on which the resource was last online. This attribute is set by VCS.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: Not applicable
MonitorOnly (system use only)	<p>Indicates if the resource can be brought online or taken offline. If set to 0, resource can be brought online or taken offline. If set to 1, resource can only be monitored.</p> <p>Note: This attribute can only be modified by the command <code>hagrp -freeze</code>.</p> <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: 0

Table D-1 Resource attributes

Resource Attributes	Description
MonitorTimeStats (system use only)	<p>Valid keys are Average and TS. Average is the average time taken by the monitor function over the last Frequency number of monitor cycles. TS is the timestamp indicating when the engine updated the resource's Average value.</p> <ul style="list-style-type: none">■ Type and dimension: string-association■ Default: Average = 0 TS = ""
Name (system use only)	<p>Contains the actual name of the resource.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: Not applicable.
Path (system use only)	<p>Set to 1 to identify a resource as a member of a path in the dependency tree to be taken offline on a specific system after a resource faults.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0
Probed (system use only)	<p>Indicates whether the state of the resource has been determined by the agent by running the monitor function.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0
ResourceInfo (system use only)	<p>This attribute has three predefined keys:</p> <p>State: values are Valid, Invalid, or Stale</p> <p>Msg: output of the info function captured on stdout by the agent framework</p> <p>TS: timestamp indicating when the ResourceInfo attribute was updated by the agent framework</p> <ul style="list-style-type: none">■ Type and dimension: string-association■ Default: State = Valid Msg = "" TS = ""

Table D-1 Resource attributes

Resource Attributes	Description
ResourceOwner (user-defined)	<p>Used for VCS email notification and logging. VCS sends email notification to the person designated in this attribute when an event occurs related to the resource. VCS also logs the owner name when an event occurs.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: If ResourceOwner is not specified in main.cf, the default value is “unknown”.
Signaled (system use only)	<p>Indicates whether a resource has been traversed. Used when bringing a service group online or taking it offline.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-association ■ Default: Not applicable.
Start (system use only)	<p>Indicates whether a resource was started (the process of bringing it online was initiated) on a system.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer -scalar ■ Default: 0
State (system use only)	<p>Resource state displays the state of the resource and the flags associated with the resource. (Flags are also captured by the Flags attribute.) This attribute and Flags present a comprehensive view of the resource's current state. Values:</p> <p>ONLINE</p> <p>OFFLINE</p> <p>FAULTED</p> <p>ONLINE STATE UNKNOWN</p> <p>ONLINE MONITOR TIMEDOUT</p> <p>ONLINE UNABLE TO OFFLINE</p> <p>OFFLINE STATE UNKNOWN</p> <p>FAULTED RESTARTING</p> <p>A FAULTED resource is physically offline, though unintentionally.</p> <p>Note: Although this attribute accepts integer types, the command line indicates the text representations.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer -scalar ■ Default: 0

Table D-1 Resource attributes

Resource Attributes	Description
TriggerEvent (system use only)	A flag that turns Events on or off. <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0

Resource type attributes

Table D-2 lists the resource type attributes.

You can override some static attributes for resource types.

See “[Overriding resource type static attributes](#)” on page 152.

For more information on any attribute listed below, see the chapter on setting agent parameters in the *Veritas Cluster Server Agent Developer’s Guide*.

Table D-2 Resource type attributes

Resource type attributes	Description
ActionTimeout (user-defined)	Timeout value for the Action function. <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 30 seconds
AgentClass (user-defined)	Indicates the scheduling class for the VCS agent process. Use only one of the following sets of attributes to configure scheduling class and priority for VCS: <ul style="list-style-type: none">■ AgentClass, AgentPriority, ScriptClass, and ScriptPriorityOr■ OnlineClass, OnlinePriority, EPClass, and EPPriority■ Type and dimension: string-scalar■ Default: TS

Table D-2 Resource type attributes

Resource type attributes	Description
AgentDirectory (user-defined)	<p>Complete path of the directory in which the agent binary and scripts are located.</p> <p>Agents look for binaries and scripts in the following directories:</p> <ul style="list-style-type: none"> ■ Directory specified by the AgentDirectory attribute ■ /opt/VRTSvcs/bin/type/ ■ /opt/VRTSagents/ha/bin/type/ <p>If none of the above directories exist, the agent does not start.</p> <p>Use this attribute in conjunction with the AgentFile attribute to specify a different location or different binary for the agent.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default = ""
AgentFailedOn (system use only)	<p>A list of systems on which the agent for the resource type has failed.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-keylist ■ Default: Not applicable.
AgentFile (user-defined)	<p>Complete name and path of the binary for an agent. If you do not specify a value for this attribute, VCS uses the agent binary at the path defined by the AgentDirectory attribute.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default = ""
AgentPriority (user-defined)	<p>Indicates the priority in which the agent process runs.</p> <p>Use only one of the following sets of attributes to configure scheduling class and priority for VCS:</p> <ul style="list-style-type: none"> ■ AgentClass, AgentPriority, ScriptClass, and ScriptPriority Or ■ OnlineClass, OnlinePriority, EPClass, and EPPriority ■ Type and dimension: string-scalar ■ Default: 0
AgentReplyTimeout (user-defined)	<p>The number of seconds the engine waits to receive a heartbeat from the agent before restarting the agent.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 130 seconds

Table D-2 Resource type attributes

Resource type attributes	Description
AgentStartTimeout (user-defined)	<p>The number of seconds after starting the agent that the engine waits for the initial agent “handshake” before restarting the agent.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 60 seconds
ArgList (user-defined)	<p>An ordered list of attributes whose values are passed to the open, close, online, offline, monitor, clean, info, and action functions.</p> <ul style="list-style-type: none">■ Type and dimension: string-vector■ Default: {} (none)
AttrChangedTimeout (user-defined) Note: This attribute can be overridden.	<p>Maximum time (in seconds) within which the attr_changed function must complete or be terminated.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 60 seconds
CleanRetryLimit (user-defined)	<p>Number of times to retry the clean function before moving a resource to ADMIN_WAIT state. If set to 0, clean is re-tried indefinitely.</p> <p>The valid values of this attribute are in the range of 0-1024.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 0
CleanTimeout (user-defined) Note: This attribute can be overridden.	<p>Maximum time (in seconds) within which the clean function must complete or else be terminated.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 60 seconds
CloseTimeout (user-defined) Note: This attribute can be overridden.	<p>Maximum time (in seconds) within which the close function must complete or else be terminated.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 60 seconds
ConfInterval (user-defined) Note: This attribute can be overridden.	<p>When a resource has remained online for the specified time (in seconds), previous faults and restart attempts are ignored by the agent. (See ToleranceLimit and RestartLimit attributes for details.)</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 600 seconds

Table D-2 Resource type attributes

Resource type attributes	Description
EPClass	<p>Enables you to control the scheduling class for the agent functions (entry points) other than the online entry point whether the entry point is in C or scripts.</p> <p>The following values are valid for this attribute:</p> <ul style="list-style-type: none"> ■ RT (Real Time) ■ TS (Time Sharing) ■ -1—indicates that VCS does not use this attribute to control the scheduling class of entry points. <p>Use only one of the following sets of attributes to configure scheduling class and priority for VCS:</p> <ul style="list-style-type: none"> ■ AgentClass, AgentPriority, ScriptClass, and ScriptPriority Or ■ OnlineClass, OnlinePriority, EPClass, and EPPriority ■ Type and dimension: string-scalar ■ Default: -1

Table D-2 Resource type attributes

Resource type attributes	Description
EPPriority	<p>Enables you to control the scheduling priority for the agent functions (entry points) other than the online entry point. The attribute controls the agent function priority whether the entry point is in C or scripts.</p> <p>The following values are valid for this attribute:</p> <ul style="list-style-type: none">■ 0—indicates the default priority value for the configured scheduling class as given by the EPClass attribute for the operating system.■ Greater than 0—indicates a value greater than the default priority for the operating system. Symantec recommends a value of greater than 0 for this attribute. A system that has a higher load requires a greater value.■ -1—indicates that VCS does not use this attribute to control the scheduling priority of entry points. <p>For example:</p> <ul style="list-style-type: none">■ (Solaris) If the EPClass attribute is set to TS, and the value of the EPPriority attribute is set to 0, then the base priority for entry points is set to 59, as that is the default value set by the operating system.■ (Solaris) If the EPClass attribute is set to TS, and the value of the EPPriority attribute is set to -20, then the scheduling priority of the entry point would be 39 (59 base value and - 20 configured value)■ (Solaris) If the EPClass attribute is set to RT, and the value of the EPPriority attribute is set to 0, then the base priority for entry points is set to 100 by the operating system. <p>Use only one of the following sets of attributes to configure scheduling class and priority for VCS:</p> <ul style="list-style-type: none">■ AgentClass, AgentPriority, ScriptClass, and ScriptPriorityOr■ OnlineClass, OnlinePriority, EPClass, and EPPriority■ Type and dimension: string-scalar■ Default: -1

Table D-2 Resource type attributes

Resource type attributes	Description
<p>ExternalStateChange (user-defined)</p> <p>Note: This attribute can be overridden.</p>	<p>Defines how VCS handles service group state when resources are intentionally brought online or taken offline outside of VCS control.</p> <p>The attribute can take the following values:</p> <p>OnlineGroup: If the configured application is started outside of VCS control, VCS brings the corresponding service group online.</p> <p>OfflineGroup: If the configured application is stopped outside of VCS control, VCS takes the corresponding service group offline.</p> <p>OfflineHold: If a configured application is stopped outside of VCS control, VCS sets the state of the corresponding VCS resource as offline. VCS does not take any parent resources or the service group offline.</p> <p>OfflineHold and OfflineGroup are mutually exclusive.</p>
<p>FaultOnMonitor Timeouts (user-defined)</p> <p>Note: This attribute can be overridden.</p>	<p>When a monitor times out as many times as the value specified, the corresponding resource is brought down by calling the clean function. The resource is then marked FAULTED, or it is restarted, depending on the value set in the RestartLimit attribute.</p> <p>When FaultOnMonitorTimeouts is set to 0, monitor failures are not considered indicative of a resource fault. A low value may lead to spurious resource faults, especially on heavily loaded systems.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 4
<p>FaultPropagation (user-defined)</p> <p>Note: This attribute can be overridden.</p>	<p>Specifies if VCS should propagate the fault up to parent resources and take the entire service group offline when a resource faults.</p> <p>The value 1 indicates that when a resource faults, VCS fails over the service group, if the group's AutoFailOver attribute is set to 1. If The value 0 indicates that when a resource faults, VCS does not take other resources offline, regardless of the value of the Critical attribute. The service group does not fail over on resource fault.</p> <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: 1

Table D-2 Resource type attributes

Resource type attributes	Description
FireDrill (user-defined)	<p>Specifies whether or not fire drill is enabled for resource type. If set to 1, fire drill is enabled. If set to 0, it is disabled.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0
InfoInterval (user-defined)	<p>Duration (in seconds) after which the info function is invoked by the agent framework for ONLINE resources of the particular resource type.</p> <p>If set to 0, the agent framework does not periodically invoke the info function. To manually invoke the info function, use the command <code>hares -refreshinfo</code>. If the value you designate is 30, for example, the function is invoked every 30 seconds for all ONLINE resources of the particular resource type.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 0
IntentionalOffline (user-defined)	<p>Defines how VCS reacts when a configured application is intentionally stopped outside of VCS control.</p> <p>Add this attribute for agents that support detection of an intentional offline outside of VCS control. Note that the intentional offline feature is available for agents registered as V51 or later.</p> <p>The value 0 instructs the agent to register a fault and initiate the failover of a service group when the supported resource is taken offline outside of VCS control.</p> <p>The value 1 instructs VCS to take the resource offline when the corresponding application is stopped outside of VCS control.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0
InfoTimeout (user-defined)	<p>Timeout value for info function. If function does not complete by the designated time, the agent framework cancels the function's thread.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 30 seconds

Table D-2 Resource type attributes

Resource type attributes	Description
LogDbg (user-defined)	<p>Indicates the debug severities enabled for the resource type or agent framework. Debug severities used by the agent functions are in the range of DBG_1–DBG_21. The debug messages from the agent framework are logged with the severities DBG_AGINFO, DBG_AGDEBUG and DBG_AGTRACE, representing the least to most verbose.</p> <ul style="list-style-type: none">■ Type and dimension: string-keylist■ Default: {} (none)
LogFileSize (user-defined)	<p>Specifies the size (in bytes) of the agent log file. Minimum value is 64 KB. Maximum value is 134217728 bytes (128MB).</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 33554432 (32MB)
MonitorInterval (user-defined) Note: This attribute can be overridden.	<p>Duration (in seconds) between two consecutive monitor calls for an ONLINE or transitioning resource.</p> <p>A low value may impact performance if many resources of the same type exist. A high value may delay detection of a faulted resource.</p> <p>Note: The value of this attribute for the MultiNICB type must be less than its value for the IPMultiNICB type. See the <i>Veritas Cluster Server Bundled Agents Reference Guide</i> for more information.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 60 seconds

Table D-2 Resource type attributes

Resource type attributes	Description
MonitorStatsParam (user-defined)	<p>Stores the required parameter values for calculating monitor time statistics.</p> <pre>static str MonitorStatsParam = {Frequency = 10, ExpectedValue = 3000, ValueThreshold = 100, AvgThreshold = 40}</pre> <p>Frequency: The number of monitor cycles after which the average monitor cycle time should be computed and sent to the engine. If configured, the value for this attribute must be between 1 and 30. The value 0 indicates that the monitor cycle time should not be computed. Default=0.</p> <p>ExpectedValue: The expected monitor time in milliseconds for all resources of this type. Default=100.</p> <p>ValueThreshold: The acceptable percentage difference between the expected monitor cycle time (ExpectedValue) and the actual monitor cycle time. Default=100.</p> <p>AvgThreshold: The acceptable percentage difference between the benchmark average and the moving average of monitor cycle times. Default=40.</p> <ul style="list-style-type: none">■ Type and dimension: integer-association■ Default: Different value for each parameter.
MonitorTimeout (user-defined) Note: This attribute can be overridden.	<p>Maximum time (in seconds) within which the monitor function must complete or else be terminated.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 60 seconds
NumThreads (user-defined)	<p>Number of threads used within the agent process for managing resources. This number does not include threads used for other internal purposes.</p> <p>If the number of resources being managed by the agent is less than or equal to the NumThreads value, only that many number of threads are created in the agent. Addition of more resources does not create more service threads. Similarly deletion of resources causes service threads to exit. Thus, setting NumThreads to 1 forces the agent to just use 1 service thread no matter what the resource count is. The agent framework limits the value of this attribute to 30.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 10

Table D-2 Resource type attributes

Resource type attributes	Description
OfflineMonitorInterval (user-defined) Note: This attribute can be overridden.	Duration (in seconds) between two consecutive monitor calls for an OFFLINE resource. If set to 0, OFFLINE resources are not monitored. <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 300 seconds
OfflineProcScanInterval	This attribute is obsolete, do not use.
OfflineTimeout (user-defined) Note: This attribute can be overridden.	Maximum time (in seconds) within which the offline function must complete or else be terminated. <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 300 seconds
OfflineWaitLimit (user-defined) Note: This attribute can be overridden.	Number of monitor intervals to wait for the resource to go offline after completing the offline procedure. Increase the value of this attribute if the resource is likely to take a longer time to go offline. <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 0
OnlineClass	Enables you to control the scheduling class for the online agent function (entry point). This attribute controls the class whether the entry point is in C or scripts. The following values are valid for this attribute: <ul style="list-style-type: none"> ■ RT (Real Time) ■ TS (Time Sharing) ■ -1—indicates that VCS does not use this attribute to control the scheduling class of entry points. Use only one of the following sets of attributes to configure scheduling class and priority for VCS: <ul style="list-style-type: none"> ■ AgentClass, AgentPriority, ScriptClass, and ScriptPriority Or ■ OnlineClass, OnlinePriority, EPClass, and EPPriority <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: -1

Table D-2 Resource type attributes

Resource type attributes	Description
OnlinePriority	<p>Enables you to control the scheduling priority for the online agent function (entry point). This attribute controls the priority whether the entry point is in C or scripts.</p> <p>The following values are valid for this attribute:</p> <ul style="list-style-type: none">■ 0—indicates the default priority value for the configured scheduling class as given by the OnlineClass for the operating system. Symantec recommends that you set the value of the OnlinePriority attribute to 0.■ Greater than 0—indicates a value greater than the default priority for the operating system.■ -1—indicates that VCS does not use this attribute to control the scheduling priority of entry points. <p>For example:</p> <ul style="list-style-type: none">■ (Solaris) If the OnlineClass attribute is set to TS, and the value of the OnlinePriority attribute is set to 0, then the base priority for entry points is set to 59, as that is the default value set by the operating system.■ (Solaris) If the OnlineClass attribute is set to TS, and the value of the OnlinePriority attribute is set to -20, then the scheduling priority of the entry point would be 39 (59 base value and - 20 configured value)■ (Solaris) If the OnlineClass attribute is set to RT, and the value of the OnlinePriority attribute is set to 0, then the base priority for entry points is set to 100 by the operating system. <p>Use only one of the following sets of attributes to configure scheduling class and priority for VCS:</p> <ul style="list-style-type: none">■ AgentClass, AgentPriority, ScriptClass, and ScriptPriorityOr■ OnlineClass, OnlinePriority, EPClass, and EPPriority■ Type and dimension: string-scalar■ Default: -1
OnlineRetryLimit (user-defined) Note: This attribute can be overridden.	<p>Number of times to retry the <code>online</code> operation if the attempt to online a resource is unsuccessful. This parameter is meaningful only if the clean operation is implemented.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 0

Table D-2 Resource type attributes

Resource type attributes	Description
OnlineTimeout (user-defined) Note: This attribute can be overridden.	Maximum time (in seconds) within which the online function must complete or else be terminated. <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 300 seconds
OnlineWaitLimit (user-defined) Note: This attribute can be overridden.	Number of monitor intervals to wait for the resource to come online after completing the online procedure. Increase the value of this attribute if the resource is likely to take a longer time to come online. <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 2
OpenTimeout (user-defined) Note: This attribute can be overridden.	Maximum time (in seconds) within which the open function must complete or else be terminated. <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 60 seconds
Operations (user-defined)	Indicates valid operations for resources of the resource type. Values are OnOnly (can online only), OnOff (can online and offline), None (cannot online or offline). <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: OnOff
ProcScanInterval	This attribute is obsolete, do not use.
RestartLimit (user-defined) Note: This attribute can be overridden.	Number of times to retry bringing a resource online when it is taken offline unexpectedly and before VCS declares it FAULTED. <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 0
ScriptClass (user-defined)	Indicates the scheduling class of the script processes (for example, online) created by the agent. Use only one of the following sets of attributes to configure scheduling class and priority for VCS: <ul style="list-style-type: none"> ■ AgentClass, AgentPriority, ScriptClass, and ScriptPriority Or ■ OnlineClass, OnlinePriority, EPClass, and EPPriority ■ Type and dimension: string-scalar ■ Default: TS

Table D-2 Resource type attributes

Resource type attributes	Description
ScriptPriority (user-defined)	<p>Indicates the priority of the script processes created by the agent.</p> <p>Use only one of the following sets of attributes to configure scheduling class and priority for VCS:</p> <ul style="list-style-type: none">■ AgentClass, AgentPriority, ScriptClass, and ScriptPriorityOr■ OnlineClass, OnlinePriority, EPClass, and EPPriority■ Type and dimension: string-scalar■ Default: 0
SourceFile (user-defined)	<p>File from which the configuration is read. Do not configure this attribute in main.cf.</p> <p>Make sure the path exists on all nodes before running a command that configures this attribute.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: .\types.cf
SupportedActions (user-defined)	<p>Valid action tokens for the resource type.</p> <ul style="list-style-type: none">■ Type and dimension: string-vector■ Default: {}
ToleranceLimit (user-defined) Note: This attribute can be overridden.	<p>After a resource goes online, the number of times the monitor function should return OFFLINE before declaring the resource FAULTED.</p> <p>A large value could delay detection of a genuinely faulted resource.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 0

Service group attributes

[Table D-3](#) lists the service group attributes.

Table D-3 Service group attributes

Service Group Attributes	Definition
ActiveCount (system use only)	<p>Number of resources in a service group that are active (online or waiting to go online). When the number drops to zero, the service group is considered offline.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable.
AdministratorGroups (user-defined)	<p>List of operating system user account groups that have administrative privileges on the service group.</p> <p>This attribute applies to clusters running in secure mode.</p> <ul style="list-style-type: none">■ Type and dimension: string-keylist■ Default: {} (none)
Administrators (user-defined)	<p>List of VCS users with privileges to administer the group.</p> <p>Note: A Group Administrator can perform all operations related to a specific service group, but cannot perform generic cluster operations.</p> <p>See “About the VCS user privilege model” on page 71.</p> <ul style="list-style-type: none">■ Type and dimension: string-keylist■ Default: {} (none)
Authority (user-defined)	<p>Indicates whether or not the local cluster is allowed to bring the service group online. If set to 0, it is not, if set to 1, it is. Only one cluster can have this attribute set to 1 for a specific global group.</p> <p>See “Serialization–The Authority attribute” on page 469.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 0

Table D-3 Service group attributes

Service Group Attributes	Definition
AutoDisabled (system use only)	<p>Indicates that VCS does not know the status of a service group (or specified system for parallel service groups). This can occur because the group is not probed (on specified system for parallel groups) in the SystemList attribute. Or the VCS engine is not running on a node designated in the SystemList attribute, but the node is visible.</p> <p>When VCS does not know the status of a service group on a node but you want VCS to consider the service group enabled, perform this command to change the AutoDisabled value to 0.</p> <pre>hagrp -autoenable grp -sys sys1</pre> <p>This command instructs VCS that even though VCS has marked the service group auto-disabled, you are sure that the service group is not online on sys1. For failover service groups, this is important because the service groups now can be brought online on remaining nodes.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0
AutoFailOver (user-defined)	<p>Indicates whether VCS initiates an automatic failover if the service group faults.</p> <p>The attribute can take the following values:</p> <ul style="list-style-type: none">■ 0—VCS does not fail over the service group.■ 1—VCS automatically fails over the service group if a suitable node exists for failover.■ 2—VCS automatically fails over the service group only if a suitable node exists in the same system zone where the service group was online. <p>To set the value as 2, you must have enabled HA/DR license and the service group must not be hybrid. If you have not defined system zones, the failover behavior is similar to 1.</p> <p>See “Controlling failover on service group or system faults” on page 365.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 1 (enabled)

Table D-3 Service group attributes

Service Group Attributes	Definition
AutoRestart (user-defined)	<p>Restarts a service group after a faulted persistent resource becomes online.</p> <p>See “About service group dependencies” on page 410.</p> <p>Note: This attribute applies only to service groups containing persistent resources.</p> <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: 1 (enabled)
AutoStart (user-defined)	<p>Designates whether a service group is automatically started when VCS is started.</p> <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: 1 (enabled)
AutoStartIfPartial (user-defined)	<p>Indicates whether to initiate bringing a service group online if the group is probed and discovered to be in a PARTIAL state when VCS is started.</p> <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: 1 (enabled)
AutoStartList (user-defined)	<p>List of systems on which, under specific conditions, the service group will be started with VCS (usually at system boot). For example, if a system is a member of a failover service group's AutoStartList attribute, and if the service group is not already running on another system in the cluster, the group is brought online when the system is started.</p> <p>VCS uses the AutoStartPolicy attribute to determine the system on which to bring the service group online.</p> <p>Note: For the service group to start, AutoStart must be enabled and Frozen must be 0. Also, beginning with 1.3.0, you must define the SystemList attribute prior to setting this attribute.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-keylist ■ Default: {} (none)

Table D-3 Service group attributes

Service Group Attributes	Definition
AutoStartPolicy (user-defined)	<p>Sets the policy VCS uses to determine on which system to bring a service group online if multiple systems are available.</p> <p>This attribute has three options:</p> <p>Order (default)—Systems are chosen in the order in which they are defined in the AutoStartList attribute.</p> <p>Load—Systems are chosen in the order of their capacity, as designated in the AvailableCapacity system attribute. System with the highest capacity is chosen first.</p> <p>Priority—Systems are chosen in the order of their priority in the SystemList attribute. Systems with the lowest priority is chosen first.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: Order
ClusterFailOverPolicy (user-defined)	<p>Determines how a global service group behaves when a cluster faults or when a global group faults. The attribute can take the following values:</p> <p>Manual—The group does not fail over to another cluster automatically.</p> <p>Auto—The group fails over to another cluster automatically if it is unable to fail over within the local cluster, or if the entire cluster faults.</p> <p>Connected—The group fails over automatically to another cluster only if it is unable to fail over within the local cluster.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: Manual
ClusterList (user-defined)	<p>Specifies the list of clusters on which the service group is configured to run.</p> <ul style="list-style-type: none">■ Type and dimension: integer-association■ Default: {} (none)

Table D-3 Service group attributes

Service Group Attributes	Definition
ContainerInfo (user-defined)	<p>Specifies if you can use the service group with the container. Assign the following values to the ContainerInfo attribute:</p> <ul style="list-style-type: none"> ■ Name: The name of the container. ■ Type: The type of container. You can set this to Zone or WPAR. ■ Enabled: Specify the value as 1, if you want to enable the container. Specify the value as 0, if you want to disable the container. <p>You can set a per-system value for this attribute.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: {}
CurrentCount (system use only)	<p>Number of systems on which the service group is active.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: Not applicable.
DeferAutoStart (system use only)	<p>Indicates whether HAD defers the auto-start of a global group in local cluster in case the global cluster is not fully connected.</p> <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: Not applicable
Enabled (user-defined)	<p>Indicates if a service group can be failed over or brought online. The attribute can have global or local scope. If you define local (system-specific) scope for this attribute, VCS prevents the service group from coming online on specified systems that have a value of 0 for the attribute. You can use this attribute to prevent failovers on a system when performing maintenance on the system.</p> <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: 1 (enabled)
Evacuate (user-defined)	<p>Indicates if VCS initiates an automatic failover when user issues <code>hastop -local -evacuate</code>.</p> <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: 1
Evacuating (system use only)	<p>Indicates the node ID from which the service group is being evacuated.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: Not applicable

Table D-3 Service group attributes

Service Group Attributes	Definition
Failover (system use only)	<p>Indicates service group is in the process of failing over.</p> <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: Not applicable
FailOverPolicy (user-defined)	<p>Sets the policy VCS uses to determine which system a group fails over to if multiple systems exist. This attribute can take the following values:</p> <p>Priority—The system defined as the lowest priority in the SystemList attribute is chosen.</p> <p>Load—The system defined with the least value in the system's Load attribute is chosen.</p> <p>RoundRobin—Systems are chosen according to how many active service groups they are hosting. The system with the least number of active service groups is chosen first.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: Priority
FaultPropagation (user-defined)	<p>Specifies if VCS should propagate the fault up to parent resources and take the entire service group offline when a resource faults.</p> <p>The value 1 indicates that when a resource faults, VCS fails over the service group, if the group's AutoFailOver attribute is set to 1. If The value 0 indicates that when a resource faults, VCS does not take other resources offline, regardless of the value of the Critical attribute. The service group does not fail over on resource fault.</p> <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: 1
FromQ (system use only)	<p>Indicates the system name from which the service group is failing over. This attribute is specified when service group failover is a direct consequence of the group event, such as a resource fault within the group or a group switch.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-association ■ Default: Not applicable
Frozen (user-defined)	<p>Disables all actions, including autostart, online and offline, and failover, except for monitor actions performed by agents. (This convention is observed by all agents supplied with VCS.)</p> <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: 0 (not frozen)

Table D-3 Service group attributes

Service Group Attributes	Definition
GroupOwner (user-defined)	<p>VCS sends email notification to the person designated in this attribute when an event occurs related to the service group. VCS also logs the owner name when an event occurs.</p> <p>Make sure to set the severity level at which you want notifications to the group owner to at least one recipient defined in the Smtprcipients attribute of the NotifierMngr agent.</p> <p>For example, to send notifications at the Information severity level to user groupadmin@yourcompany.com:</p> <ul style="list-style-type: none"> ■ Set the GroupOwner attribute to groupadmin@company.com. ■ In the list of SMTP recipients in the NotifierMngr configuration, set the associated notification severity for at least one user to Information. <pre>NotifierMngr notification { Smtprcipients = { "admin@company.com" = Information, "superadmin@company.com" = Error} Smtprserver = "mail.company.com" }</pre> ■ Type and dimension: string-scalar ■ Default: If GroupOwner is not specified in main.cf, the default value is "unknown".
Guests (user-defined)	<p>List of operating system user accounts that have Guest privileges on the service group.</p> <p>This attribute applies to clusters running in secure mode.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-keylist ■ Default: ""

Table D-3 Service group attributes

Service Group Attributes	Definition
IntentOnline (system use only)	<p>Indicates whether to keep service groups online or offline.</p> <p>VCS sets this attribute to 1 if an attempt has been made to bring the service group online.</p> <p>For failover groups, VCS sets this attribute to 0 when the group is taken offline.</p> <p>For parallel groups, it is set to 0 for the system when the group is taken offline or when the group faults and can fail over to another system.</p> <p>VCS sets this attribute to 2 for failover groups if VCS attempts to autostart a service group; for example, attempting to bring a service group online on a system from AutoStartList.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable.
LastSuccess (system use only)	<p>Indicates the time when service group was last brought online.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
Load (user-defined)	<p>Integer value expressing total system load this group will put on a system.</p> <p>For example, the administrator may assign a value of 100 to a large production SQL and 15 to a Web server.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 0
ManageFaults (user-defined)	<p>Specifies if VCS manages resource failures within the service group by calling the Clean function for the resources. This attribute can take the following values.</p> <p>NONE—VCS does not call the Clean function for any resource in the group. User intervention is required to handle resource faults.</p> <p>See “Controlling Clean behavior on resource faults” on page 368.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: ALL
ManualOps (user-defined)	<p>Indicates if manual operations are allowed on the service group.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default = 1 (enabled)

Table D-3 Service group attributes

Service Group Attributes	Definition
MigrateQ (system use only)	<p>Indicates the system from which the service group is migrating. This attribute is specified when group failover is an indirect consequence (in situations such as a system shutdown or another group faults and is linked to this group).</p> <ul style="list-style-type: none"> ■ Type and dimension: string-association ■ Default: Not applicable
NumRetries (system use only)	<p>Indicates the number of attempts made to bring a service group online. This attribute is used only if the attribute OnlineRetryLimit is set for the service group.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: Not applicable
OnlineAtUnfreeze (system use only)	<p>When a node or a service group is frozen, the OnlineAtUnfreeze attribute specifies how an offline service group reacts after it or a node is unfrozen.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: Not applicable
OnlineRetryInterval (user-defined)	<p>Indicates the interval, in seconds, during which a service group that has successfully restarted on the same system and faults again should be failed over, even if the attribute OnlineRetryLimit is non-zero. This prevents a group from continuously faulting and restarting on the same system.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 0
OnlineRetryLimit (user-defined)	<p>If non-zero, specifies the number of times the VCS engine tries to restart a faulted service group on the same system on which the group faulted, before it gives up and tries to fail over the group to another system.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 0
OperatorGroups (user-defined)	<p>List of operating system user groups that have Operator privileges on the service group.</p> <p>This attribute applies to clusters running in secure mode.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-keylist ■ Default: ""

Table D-3 Service group attributes

Service Group Attributes	Definition
Operators (user-defined)	<p>List of VCS users with privileges to operate the group. A Group Operator can only perform online/offline, and temporary freeze/unfreeze operations pertaining to a specific group.</p> <p>See “About the VCS user privilege model” on page 71.</p> <ul style="list-style-type: none">■ Type and dimension: string-keylist■ Default: ""
Parallel (user-defined)	<p>Indicates if service group is failover (0), parallel (1), or hybrid(2).</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 0
PathCount (system use only)	<p>Number of resources in path not yet taken offline. When this number drops to zero, the engine may take the entire service group offline if critical fault has occurred.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
PreOnline (user-defined)	<p>Indicates that the VCS engine should not online a service group in response to a manual group online, group autostart, or group failover. The engine should instead run the PreOnline trigger.</p> <p>See “preonline event trigger” on page 454.</p> <p>You can set a local (per-system) value for this attribute to control the firing of PreOnline triggers on each node in the cluster. This attribute is strictly a per system attribute, the you must set the value for each system.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0
PreOnlining (system use only)	<p>Indicates that VCS engine invoked the preonline script; however, the script has not yet returned with group online.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable

Table D-3 Service group attributes

Service Group Attributes	Definition
PreonlineTimeout (user-defined)	<p>Defines the maximum amount of time in seconds the preonline script takes to run the command <code>hagrp -online -nopro</code> for the group. Note that HAD uses this timeout during evacuation only. For example, when a user runs the command <code>hastop -local -evacuate</code> and the Preonline trigger is invoked on the system on which the service groups are being evacuated.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 300
Prerequisites (user-defined)	<p>An unordered set of name=value pairs denoting specific resources required by a service group. If prerequisites are not met, the group cannot go online. The format for Prerequisites is: <code>Prerequisites() = {Name=Value, name2=value2}</code>.</p> <p>Names used in setting Prerequisites are arbitrary and not obtained from the system. Coordinate name=value pairs listed in Prerequisites with the same name=value pairs in Limits().</p> <p>See “About system limits and service group prerequisites” on page 391.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-association

Table D-3 Service group attributes

Service Group Attributes	Definition
PreSwitch (user-defined)	<p>Indicates whether VCS engine should invoke PreSwitch actions in response to a manual service group switch operation.</p> <p>Note: The engine does not invoke the PreSwitch action during a group fault or when you use <code>-any</code> option to switch a group.</p> <p>This attribute must be defined in the global group definition on the remote cluster. This attribute takes the following values:</p> <p>0—VCS engine switches the service group normally.</p> <p>1—VCS engine switches the service group based on the output of PreSwitch action of the resources.</p> <p>If you set the value as 1, the VCS engine looks for any resource in the service group that supports PreSwitch action. If the action is not defined for any resource, the VCS engine switches a service group normally.</p> <p>If the action is defined for one or more resources, then the VCS engine invokes PreSwitch action for those resources. If all the actions succeed, the engine switches the service group. If any of the actions fail, the engine aborts the switch operation.</p> <p>The engine invokes the PreSwitch action in parallel and waits for all the actions to complete to decide whether to perform a switch operation. The VCS engine reports the action's output to the engine log. The PreSwitch action does not change the configuration or the cluster state.</p> <p>See “Administering global service groups” on page 533.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0
PreSwitching (system use only)	<p>Indicates that the VCS engine invoked the agent's PreSwitch action; however, the action is not yet complete.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
PrintTree (user-defined)	<p>Indicates whether or not the resource dependency tree is written to the configuration file. The value 1 indicates the tress is written.</p> <p>Note: For very large configurations, the time taken to print the tree and to dump the configuration is high.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 1

Table D-3 Service group attributes

Service Group Attributes	Definition
Priority (user-defined)	<p>Enables users to designate and prioritize the service group. VCS does not interpret the value; rather, this attribute enables the user to configure the priority of a service group and the sequence of actions required in response to a particular event.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 0
Probed (system use only)	<p>Indicates whether all enabled resources in the group have been detected by their respective agents.</p> <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: Not applicable
ProbesPending (system use only)	<p>The number of resources that remain to be detected by the agent on each system.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: Not applicable
Responding (system use only)	<p>Indicates VCS engine is responding to a failover event and is in the process of bringing the service group online or failing over the node.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: Not applicable
Restart (system use only)	<p>For internal use only.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: Not applicable
SourceFile (system use only)	<p>File from which the configuration is read. Do not configure this attribute in main.cf.</p> <p>Make sure the path exists on all nodes before running a command that configures this attribute.</p> <p>Make sure the path exists on all nodes before configuring this attribute.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: ./main.cf

Table D-3 Service group attributes

Service Group Attributes	Definition
State (system use only)	<p>Group state on each system:</p> <p>OFFLINE— All non-persistent resources are offline.</p> <p>ONLINE —All resources whose AutoStart attribute is equal to 1 are online.</p> <p>FAULTED—At least one critical resource in the group is faulted or is affected by a fault.</p> <p>PARTIAL—At least one, but not all, resources with Operations=OnOff is online, and not all AutoStart resources are online.</p> <p>STARTING—Group is attempting to go online.</p> <p>STOPPING— Group is attempting to go offline.</p> <p>A group state may be a combination of the multiple states described above. For example, OFFLINE FAULTED, OFFLINE STARTED, PARTIAL FAULTED, PARTIAL STARTING, PARTIAL STOPPING, ONLINE STOPPING</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable.
SystemList (user-defined)	<p>List of systems on which the service group is configured to run and their priorities. Lower numbers indicate a preference for the system as a failover target.</p> <p>Note: You must define this attribute prior to setting the AutoStartList attribute.</p> <ul style="list-style-type: none">■ Type and dimension: integer-association■ Default: "" (none)
SystemZones (user-defined)	<p>Indicates the virtual sublists within the SystemList attribute that grant priority in failing over. Values are string/integer pairs. The string key is the name of a system in the SystemList attribute, and the integer is the number of the zone. Systems with the same zone number are members of the same zone. If a service group faults on one system in a zone, it is granted priority to fail over to another system within the same zone, despite the policy granted by the FailOverPolicy attribute.</p> <ul style="list-style-type: none">■ Type and dimension: integer-association■ Default: "" (none)

Table D-3 Service group attributes

Service Group Attributes	Definition
Tag (user-defined)	Identifies special-purpose service groups created for specific VCS products. <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: ""
TargetCount (system use only)	Indicates the number of target systems on which the service group should be brought online. <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: Not applicable.
TFrozen (user-defined)	Indicates if service groups can be brought online or taken offline on nodes in the cluster. Service groups cannot be brought online or taken offline if the value of the attribute is 1. <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: 0 (not frozen)
ToQ (system use only)	Indicates the node name to which the service is failing over. This attribute is specified when service group failover is a direct consequence of the group event, such as a resource fault within the group or a group switch. <ul style="list-style-type: none"> ■ Type and dimension: string-association ■ Default: Not applicable
TriggerEvent (system use only)	For internal use only. <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: Not applicable
TriggerResFault (user-defined)	Defines whether VCS invokes the resfault trigger when a resource faults. The value 0 indicates that VCS does not invoke the trigger. See “ resfault event trigger ” on page 457. <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 1
TriggerResStateChange (user-defined)	Determines whether or not to invoke the resstatechange trigger if resource state changes. See “ resstatechange event trigger ” on page 459. <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: 0 (disabled)

Table D-3 Service group attributes

Service Group Attributes	Definition
TypeDependencies (user-defined)	<div>Creates a dependency (via an ordered list) between resource types specified in the service group list, and all instances of the respective resource type.</div> <div><div>■ Type and dimension: string-keylist</div><div>■ Default: ""</div></div>

Table D-3 Service group attributes

Service Group Attributes	Definition
UserIntGlobal (user-defined)	<p>Use this attribute for any purpose. It is not used by VCS.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 0
UserStrGlobal (user-defined)	<p>VCS uses this attribute in the ClusterService group. Do not modify this attribute in the ClusterService group.</p> <p>Use the attribute for any purpose in other service groups.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: 0
UserIntLocal (user-defined)	<p>Use this attribute for any purpose. It is not used by VCS.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 0
UserStrLocal (user-defined)	<p>Use this attribute for any purpose. It is not used by VCS.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: ""

System attributes

Table D-4 lists the system attributes.

Table D-4 System attributes

System Attributes	Definition
AgentsStopped (system use only)	<p>This attribute is set to 1 on a system when all agents running on the system are stopped.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
AvailableCapacity (system use only)	<p>Indicates system's available capacity when trigger is fired. If this value is negative, the argument contains the prefix % (percentage sign); for example, %-4.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
Capacity (user-defined)	<p>Value expressing total system load capacity. This value is relative to other systems in the cluster and does not reflect any real value associated with a particular system.</p> <p>For example, the administrator may assign a value of 200 to a 16-processor machine and 100 to an 8-processor machine.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 100
ConfigBlockCount (system use only)	<p>Number of 512-byte blocks in configuration when the system joined the cluster.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
ConfigChecksum (system use only)	<p>Sixteen-bit checksum of configuration identifying when the system joined the cluster.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
ConfigDiskState (system use only)	<p>State of configuration on the disk when the system joined the cluster.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
ConfigFile (user-defined)	<p>Directory containing the configuration files.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: ""

Table D-4 System attributes

System Attributes	Definition
ConfigInfoCnt (system use only)	<p>The count of outstanding CONFIG_INFO messages the local node expects from a new membership message. This attribute is non-zero for the brief period during which new membership is processed. When the value returns to 0, the state of all nodes in the cluster is determined.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: Not applicable
ConfigModDate (system use only)	<p>Last modification date of configuration when the system joined the cluster.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: Not applicable
CPUUsage (system use only)	<p>Indicates the system's CPU usage by CPU percentage utilization. This attribute's value is valid if the Enabled value in the CPUUsageMonitoring attribute (below) equals 1. The value of this attribute is updated when there is a change of five percent since the last indicated value.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: Not applicable
CPUUsageMonitoring	<p>Monitors the system's CPU usage using various factors.</p> <p>The values for ActionTimeLimit and NotifyTimeLimit represent the time in seconds. The values for ActionThreshold and NotifyThreshold represent the threshold in terms of CPU percentage utilization.</p> <p>See “Monitoring CPU usage” on page 578.</p> <p>This attribute will be deprecated in a future release. VCS monitors system resources on startup.</p> <p>See “About the HostMonitor daemon” on page 30.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-association ■ Default: Enabled = 0, NotifyThreshold = 0, NotifyTimeLimit = 0, ActionThreshold = 0, ActionTimeLimit = 0, Action = NONE.
CurrentLimits (system use only)	<p>System-maintained calculation of current value of Limits.</p> <p>CurrentLimits = Limits - (additive value of all service group Prerequisites).</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-association ■ Default: Not applicable

Table D-4 System attributes

System Attributes	Definition
DiskHbStatus (system use only)	<p>Deprecated attribute. Indicates status of communication disks on any system.</p> <ul style="list-style-type: none">■ Type and dimension: string-association■ Default: Not applicable
DynamicLoad (user-defined)	<p>System-maintained value of current dynamic load. The value is set external to VCS with the <code>hasys -load</code> command. When you specify the dynamic system load, VCS des not use the static group load.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 0
EngineRestarted (system use only)	<p>Indicates whether the VCS engine (HAD) was restarted by the hashadow process on a node in the cluster. The value 1 indicates that the engine was restarted; 0 indicates it was not restarted.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0
EngineVersion (system use only)	<p>Specifies the major, minor, maintenance-patch, and point-patch version of VCS.</p> <p>The value of EngineVersion attribute is in hexa-decimal format. To retrieve version information:</p> <pre>Major Version: EngineVersion >> 24 & 0xff Minor Version: EngineVersion >> 16 & 0xff Maint Patch: EngineVersion >> 8 & 0xff Point Patch: EngineVersion & 0xff</pre> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
Frozen (user-defined)	<p>Indicates if service groups can be brought online on the system. Groups cannot be brought online if the attribute value is 1.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0
GUIIPAddr (user-defined)	<p>Determines the local IP address that VCS uses to accept connections. Incoming connections over other IP addresses are dropped. If GUIIPAddr is not set, the default behavior is to accept external connections over all configured local IP addresses.</p> <p>See “User privileges for CLI commands” on page 74.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: ““

Table D-4 System attributes

System Attributes	Definition
HostMonitor (system use only)	<p>List of host resources that the HostMonitor daemon monitors.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-keylist ■ Default: { CPU, Swap }
HostUtilization (system use only)	<p>Indicates the usage percentages of the resources on the host as computed by the HostMonitor daemon.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-association ■ Default: Not applicable
LicenseType (system use only)	<p>Indicates the license type of the base VCS key used by the system. Possible values are:</p> <p>0—DEMO</p> <p>1—PERMANENT</p> <p>2—PERMANENT_NODE_LOCK</p> <p>3—DEMO_NODE_LOCK</p> <p>4—NFR</p> <p>5—DEMO_EXTENSION</p> <p>6—NFR_NODE_LOCK</p> <p>7—DEMO_EXTENSION_NODE_LOCK</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: Not applicable
Limits (user-defined)	<p>An unordered set of name=value pairs denoting specific resources available on a system. Names are arbitrary and are set by the administrator for any value. Names are not obtained from the system.</p> <p>The format for Limits is:</p> <p>Limits = {Name=Value, Name2=Value2}.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-association ■ Default: ""

Table D-4 System attributes

System Attributes	Definition
LinkHbStatus (system use only)	<p>Indicates status of private network links on any system.</p> <p>Possible values include the following:</p> <p>LinkHbStatus = { <i>nic1</i> = UP, <i>nic2</i> = DOWN }</p> <p>Where the value UP for <i>nic1</i> means there is at least one peer in the cluster that is visible on <i>nic1</i>.</p> <p>Where the value DOWN for <i>nic2</i> means no peer in the cluster is visible on <i>nic2</i>.</p> <ul style="list-style-type: none">■ Type and dimension: string-association■ Default: Not applicable
LLTNodeId (system use only)	<p>Displays the node ID defined in the file /etc/llttab.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
LoadTimeCounter (system use only)	<p>System-maintained internal counter of how many seconds the system load has been above LoadWarningLevel. This value resets to zero anytime system load drops below the value in LoadWarningLevel.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
LoadTimeThreshold (user-defined)	<p>How long the system load must remain at or above LoadWarningLevel before the LoadWarning trigger is fired. If set to 0 overload calculations are disabled.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 600
LoadWarningLevel (user-defined)	<p>A percentage of total capacity where load has reached a critical limit. If set to 0 overload calculations are disabled.</p> <p>For example, setting LoadWarningLevel = 80 sets the warning level to 80 percent.</p> <p>The value of this attribute can be set from 1 to 100. If set to 1, system load must equal 1 percent of system capacity to begin incrementing the LoadTimeCounter. If set to 100, system load must equal system capacity to increment the LoadTimeCounter.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 80

Table D-4 System attributes

System Attributes	Definition
NoAutoDisable (system use only)	<p>When set to 0, this attribute autodisables service groups when the VCS engine is taken down. Groups remain autodisabled until the engine is brought up (regular membership).</p> <p>This attribute's value is updated whenever a node joins (gets into RUNNING state) or leaves the cluster. This attribute cannot be set manually.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0
NodeId (system use only)	<p>System (node) identification specified in <code>/etc/llttab</code>.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
OnGrpCnt (system use only)	<p>Number of groups that are online, or about to go online, on a system.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
ShutdownTimeout (user-defined)	<p>Determines whether to treat system reboot as a fault for service groups running on the system.</p> <p>On many systems, when a reboot occurs the processes are stopped first, then the system goes down. When the VCS engine is stopped, service groups that include the failed system in their SystemList attributes are autodisabled. However, if the system goes down within the number of seconds designated in ShutdownTimeout, service groups previously online on the failed system are treated as faulted and failed over. Symantec recommends that you set this attribute depending on the average time it takes to shut down the system.</p> <p>If you do not want to treat the system reboot as a fault, set the value for this attribute to 0.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 600 seconds
SourceFile (user-defined)	<p>File from which the configuration is read. Do not configure this attribute in <code>main.cf</code>.</p> <p>Make sure the path exists on all nodes before running a command that configures this attribute.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: <code>./main.cf</code>

Table D-4 System attributes

System Attributes	Definition
SysInfo (system use only)	Provides platform-specific information, including the name, version, and release of the operating system, the name of the system on which it is running, and the hardware type. <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: Not applicable
SysName (system use only)	Indicates the system name. <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: Not applicable
SysState (system use only)	Indicates system states, such as RUNNING, FAULTED, EXITED, etc. <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
SystemLocation (user-defined)	Indicates the location of the system. <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: ""
SystemOwner (user-defined)	This attribute is used for VCS email notification and logging. VCS sends email notification to the person designated in this attribute when an event occurs related to the system. VCS also logs the owner name when an event occurs. <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: "unknown"
TFrozen (user-defined)	Indicates a service group that can be brought online on a node. Service group cannot be brought online if the value of this attribute is 1. <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0
TRSE (system use only)	Indicates in seconds the time to Regular State Exit. Time is calculated as the duration between the events of VCS losing port h membership and of VCS losing port a membership of GAB. <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable

Table D-4 System attributes

System Attributes	Definition
UpDownState (system use only)	<p>This attribute has four values:</p> <p>0 (down): System is powered off, or GAB and LLT are not running on the system.</p> <p>1 (Up but not in cluster membership): GAB and LLT are running but the VCS engine is not.</p> <p>2 (up and in jeopardy): The system is up and part of cluster membership, but only one network link (LLT) remains.</p> <p>3 (up): The system is up and part of cluster membership, and has at least two links to the cluster.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
UserInt (user-defined)	<p>Stores integer values you want to use. VCS does not interpret the value of this attribute.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 0
VCSFeatures (system use only)	<p>Indicates which VCS features are enabled.</p> <p>Possible values are:</p> <p>0—No features enabled (VCS Simulator)</p> <p>1—L3+ is enabled</p> <p>2—Global Cluster Option is enabled</p> <p>Even though the VCSFeatures is an integer attribute, when you query the value with the <code>haclus -value</code> command or the <code>haclus -display</code> command, it displays as the string <code>L10N</code> for value 1 and <code>DR</code> for value 2.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable

Cluster attributes

Table D-5 lists the cluster attributes.

Table D-5 Cluster attributes

Cluster Attributes	Definition
AdministratorGroups (user-defined)	<p>List of operating system user account groups that have administrative privileges on the cluster.</p> <p>This attribute applies to clusters running in secure mode.</p> <ul style="list-style-type: none">■ Type and dimension: string-keylist■ Default: ""
Administrators (user-defined)	<p>Contains list of users with Administrator privileges.</p> <ul style="list-style-type: none">■ Type and dimension: string-keylist■ Default: ""
AutoStartTimeout (user-defined)	<p>If the local cluster cannot communicate with one or more remote clusters, this attribute specifies the number of seconds the VCS engine waits before initiating the AutoStart process for an AutoStart global service group.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 150 seconds
BackupInterval (user-defined)	<p>Time period in minutes after which VCS backs up the configuration files if the configuration is in read-write mode.</p> <p>The value 0 indicates VCS does not back up configuration files. Set this attribute to at least 3.</p> <p>See “Scheduling automatic backups for VCS configuration files” on page 199.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 0

Table D-5 Cluster attributes

Cluster Attributes	Definition
CID (system defined)	<p>The CID provides universally unique identification for a cluster.</p> <p>VCS populates this attribute once the engine passes an hacf-generated snapshot to it. This happens when VCS is about to go to a RUNNING state from the LOCAL_BUILD state.</p> <p>Once VCS receives the snapshot from the engine, it reads the file /etc/vx/.uuids/clusuuid file. VCS uses the file's contents as the value for the CID attribute. The clusuuid file's first line must not be empty. If the file does not exist or is empty VCS then exits gracefully and throws an error.</p> <p>A node that joins a cluster in the RUNNING state receives the CID attribute as part of the REMOTE_BUILD snapshot. Once the node has joined completely, it receives the snapshot. The node reads the file /etc/vx/.uuids/clusuuid to compare the value that it received from the snapshot with value that is present in the file. If the value does not match or if the file does not exist, the joining node exits gracefully and does not join the cluster.</p> <p>To populate the /etc/vx/.uuids/clusuuid file, run the /opt/VRTSvcs/bin/uuidconfig.pl utility.</p> <p>See “Configuring and unconfiguring the cluster UUID value” on page 232.</p> <p>See “About the cluster UUID” on page 27.</p> <p>You cannot change the value of this attribute with the haclus – modify command.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: ""
ClusState (system use only)	<p>Indicates the current state of the cluster.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: Not applicable.
ClusterAddress (user-defined)	<p>Specifies the cluster's virtual IP address (used by a remote cluster when connecting to the local cluster).</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: ""
ClusterLocation (user-defined)	<p>Specifies the location of the cluster.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: ""

Table D-5Cluster attributes

Cluster Attributes	Definition
ClusterName (user-defined)	<p>The name of cluster.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: ""
ClusterOwner (user-defined)	<p>This attribute is used for VCS notification; specifically, VCS sends notifications to persons designated in this attribute when an event occurs related to the cluster.</p> <p>See“About VCS event notification” on page 430.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: “unknown”
ClusterTime (system use only)	<p>The number of seconds since January 1, 1970. This is defined by the lowest node in running state.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: Not applicable
ClusterUUID (system use only)	<p>Unique ID assigned to the cluster by Availability Manager.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: Not applicable
CompareRSM (system use only)	<p>Indicates if VCS engine is to verify that replicated state machine is consistent. This can be set by running the <code>hadebug</code> command.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 0
ConnectorState (system use only)	<p>Indicates the state of the wide-area connector (wac). If 0, wac is not running. If 1, wac is running and communicating with the VCS engine.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable.
CounterInterval (user-defined)	<p>Intervals counted by the attribute GlobalCounter indicating approximately how often a broadcast occurs that will cause the GlobalCounter attribute to increase.</p> <p>The default value of the GlobalCounter increment can be modified by changing CounterInterval. If you increase this attribute to exceed five seconds, consider increasing the default value of the ShutdownTimeout attribute.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 5

Table D-5 Cluster attributes

Cluster Attributes	Definition
CredRenewFrequency (user-defined)	<p>The number of days after which the VCS engine renews its credentials with the authentication broker. For example, the value 5 indicates that credentials are renewed every 5 days; the value 0 indicates that credentials are not renewed.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default = 0
DeleteOnlineResource (user-defined)	<p>Defines whether you can delete online resources. Set this value to 1 to enable deletion of online resources. Set this value to 0 to disable deletion of online resources.</p> <p>You can override this behavior by using the <code>-force</code> option with the <code>hares -delete</code> command.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default = 1
DumpingMembership (system use only)	<p>Indicates that the engine is writing to disk.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable.
EngineClass (user-defined)	<p>The scheduling class for the VCS engine (HAD).</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: RT
EnginePriority (user-defined)	<p>The priority in which HAD runs.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: ""

Table D-5 Cluster attributes

Cluster Attributes	Definition
EngineShutdown (user-defined)	<p>Defines the options for the hastop command. The attribute can assume the following values:</p> <p>Enable—Process all hastop commands. This is the default behavior.</p> <p>Disable—Reject all hastop commands.</p> <p>DisableClusStop—Do not process the hastop -all command; process all other hastop commands.</p> <p>PromptClusStop—Prompt for user confirmation before running the hastop -all command; process all other hastop commands.</p> <p>PromptLocal—Prompt for user confirmation before running the hastop -local command; reject all other hastop commands.</p> <p>PromptAlways—Prompt for user confirmation before running any hastop command.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: Enable
GlobalCounter (system use only)	<p>This counter increases incrementally by one for each counter interval. It increases when the broadcast is received.</p> <p>VCS uses the GlobalCounter attribute to measure the time it takes to shut down a system. By default, the GlobalCounter attribute is updated every five seconds. This default value, combined with the 600-second default value of the ShutdownTimeout attribute, means if system goes down within 120 increments of GlobalCounter, it is treated as a fault. Change the value of the CounterInterval attribute to modify the default value of GlobalCounter increment.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable.
Guests (user-defined)	<p>List of operating system user accounts that have Guest privileges on the cluster.</p> <p>This attribute is valid clusters running in secure mode.</p> <ul style="list-style-type: none">■ Type and dimension: string-keylist■ Default: ""
GroupLimit (user-defined)	<p>Maximum number of service groups.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 200

Table D-5 Cluster attributes

Cluster Attributes	Definition
HacliUserLevel (user-defined)	<p>This attribute has two, case-sensitive values:</p> <p>NONE-hacli is disabled for all users regardless of role.</p> <p>COMMANDROOT-hacli is enabled for root only.</p> <p>Note: The command <code>haclus -modify HacliUserLevel</code> can be executed by root only.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: NONE
HostMonLogLvl (user-defined)	<p>Controls the behavior of the HostMonitor feature.</p> <p>Configure this attribute when you start the cluster. You cannot modify this attribute in a running cluster.</p> <p>This attribute has the following possible values:</p> <p>ALL-The HostMonitor daemon logs messages engine log and to the agent log.</p> <p>HMAgentLog-The HostMonitor daemon does not log messages to the engine log; the daemon logs messages to the HostMonitor agent log.</p> <p>DisableHMAgent-Disables the HostMonitor feature.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: ALL
LockMemory (user-defined)	<p>Controls the locking of VCS engine pages in memory. This attribute has the following values. Values are case-sensitive:</p> <p>ALL: Locks all current and future pages.</p> <p>CURRENT: Locks current pages.</p> <p>NONE: Does not lock any pages.</p> <p>On AIX, this attribute includes only one value, ALL, which is the default. This value locks text and data into memory (process lock).</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: ALL
LogSize (user-defined)	<p>Indicates the size of engine log file in bytes.</p> <p>Minimum value is = 65536 (equal to 64KB)</p> <p>Maximum value = 134217728 (equal to 128MB)</p> <p>Default value = 33554432 (equal to 32MB)</p> <ul style="list-style-type: none"> ■ Type and dimensions: integer -scalar ■ Default value = 33554432

Table D-5 Cluster attributes

Cluster Attributes	Definition
Notifier (system use only)	<p>Indicates the status of the notifier in the cluster; specifically:</p> <p>State—Current state of notifier, such as whether or not it is connected to VCS.</p> <p>Host—The host on which notifier is currently running or was last running. Default = None</p> <p>Severity—The severity level of messages queued by VCS for notifier. Values include Information, Warning, Error, and SevereError. Default = Warning</p> <p>Queue—The size of queue for messages queued by VCS for notifier.</p> <ul style="list-style-type: none">■ Type and dimension: string-association■ Default: Different values for each parameter.
OperatorGroups (user-defined)	<p>List of operating system user groups that have Operator privileges on the cluster.</p> <p>This attribute is valid clusters running in secure mode.</p> <ul style="list-style-type: none">■ Type and dimension: string-keylist■ Default: ""
Operators (user-defined)	<p>List of users with Cluster Operator privileges.</p> <ul style="list-style-type: none">■ Type and dimension: string-keylist■ Default: ""
PanicOnNoMem (user-defined)	<ul style="list-style-type: none">■ Indicate the action that you want VCS engine (HAD) to take if it cannot receive messages from GAB due to low-memory.■ If the value is 0, VCS exits with warnings. If the value is 1, VCS calls the GAB library routine to panic the system.■ Default: 0
PrintMsg (user-defined)	<p>Enables logging TagM messages in engine log if set to 1.</p> <ul style="list-style-type: none">■ Type and dimension: boolean-scalar■ Default: 0
ProcessClass (user-defined)	<p>Indicates the scheduling class for processes created by the VCS engine. For example, triggers.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default = TS
ProcessPriority (user-defined)	<p>The priority of processes created by the VCS engine, for example: triggers.</p> <ul style="list-style-type: none">■ Type and dimension: string-scalar■ Default: ""

Table D-5 Cluster attributes

Cluster Attributes	Definition
ReadOnly (user-defined)	Indicates that cluster is in read-only mode. <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 1
ResourceLimit (user-defined)	Maximum number of resources. <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 5000
SecInfo	For future use.
SecInfoLevel	For future use.
SecureClus (user-defined)	Indicates whether the cluster runs in secure mode. The value 1 indicated the cluster runs in secure mode. This attribute cannot be modified when VCS is running. <ul style="list-style-type: none"> ■ Type and dimension: boolean-scalar ■ Default: 0
SourceFile (user-defined)	File from which the configuration is read. Do not configure this attribute in main.cf. <p>Make sure the path exists on all nodes before running a command that configures this attribute.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: Not applicable.
Stewards (user-defined)	The IP address and hostname of systems running the steward process. <ul style="list-style-type: none"> ■ Type and dimension: string-keylist ■ Default: {}
TypeLimit (user-defined)	Maximum number of resource types. <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 100
UseFence (user-defined)	Indicates whether the cluster uses I/O fencing. <p>The value SCSI3 indicates that the cluster uses either disk-based or server-based I/O fencing; the value NONE indicates it does not.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-scalar ■ Default: NONE

Table D-5 Cluster attributes

Cluster Attributes	Definition
UserNames (user-defined)	<p>List of VCS users.</p> <p>Note: The installer program uses <code>admin</code> as the default user name.</p> <ul style="list-style-type: none">■ Type and dimension: string-association■ Default: {}
VCSFeatures (system use only)	<p>Indicates which VCS features are enabled. Possible values are:</p> <p>0—No features are enabled (VCS Simulator)</p> <p>1—L3+ is enabled</p> <p>2—Global Cluster Option is enabled</p> <p>Even though the <code>VCSFeatures</code> is an integer attribute, when you query the value with the <code>haclus -value</code> command or the <code>haclus -display</code> command, it displays as the string <code>L10N</code> for value 1 and <code>DR</code> for value 2.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable.
VCSMode (system use only)	<p>Denotes the mode for which VCS is licensed.</p> <p>Even though the <code>VCSFeatures</code> is an integer attribute, when you query the value with the <code>haclus -value</code> command or the <code>haclus -display</code> command, it displays as the string <code>UNKNOWN_MODE</code> for value 0 and <code>VCS</code> for value 7.</p> <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
WACPort (user-defined)	<p>The TCP port on which the <code>wac</code> (Wide-Area Connector) process on the local cluster listens for connection from remote clusters.</p> <p>Type and dimension: integer-scalar</p> <ul style="list-style-type: none">■ Default: 14155

Heartbeat attributes (for global clusters)

Table D-6 lists the heartbeat attributes. These attributes apply to global clusters.

Table D-6 Heartbeat attributes

Heartbeat Attributes	Definition
AgentState (system use only)	<p>The state of the heartbeat agent.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: INIT
Arguments (user-defined)	<p>List of arguments to be passed to the agent functions. For the Icmp agent, this attribute can be the IP address of the remote cluster.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-vector ■ Default: ""
AYAInterval (user-defined)	<p>The interval in seconds between two heartbeats.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 60 seconds
AYARetryLimit (user-defined)	<p>The maximum number of lost heartbeats before the agent reports that heartbeat to the cluster is down.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 3
AYATimeout (user-defined)	<p>The maximum time (in seconds) that the agent will wait for a heartbeat AYA function to return ALIVE or DOWN before being canceled.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 30
CleanTimeOut (user-defined)	<p>Number of seconds within which the Clean function must complete or be canceled.</p> <ul style="list-style-type: none"> ■ Type and dimension: integer-scalar ■ Default: 300 seconds
ClusterList (user-defined)	<p>List of remote clusters.</p> <ul style="list-style-type: none"> ■ Type and dimension: string-keylist ■ Default: ""

Table D-6 Heartbeat attributes

Heartbeat Attributes	Definition
InitTimeout (user-defined)	Number of seconds within which the Initialize function must complete or be canceled. <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 300 seconds
LogDbg (user-defined)	The log level for the heartbeat. <ul style="list-style-type: none">■ Type and dimension: string-keylist■ Default: ""
State	The state of the heartbeat. <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: Not applicable
StartTimeout (user-defined)	Number of seconds within which the Start function must complete or be canceled. <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 300 seconds
StopTimeout (user-defined)	Number of seconds within which the Stop function must complete or be canceled without stopping the heartbeat. <ul style="list-style-type: none">■ Type and dimension: integer-scalar■ Default: 300 seconds

Configuring VCS for partitions

- [About VCS support for WPARs](#)
- [Configuring VCS in WPARs](#)
- [Installing the application](#)
- [Verifying the WPAR configuration](#)
- [Maintenance tasks](#)
- [Troubleshooting information](#)

About VCS support for WPARs

VCS provides application management and high availability to applications that run in WPARs. VCS supports only system WPARs, application WPARs are not supported.

Overview of how VCS works with WPARs

You can use VCS to perform the following:

- Start, stop, monitor, and failover a local WPAR.
- Start, stop, monitor, and failover an application that runs in a WPAR.

Installing and configuring WPARs in VCS environments

Install and configure the WPAR. Create the service group with the standard application resource types (application, storage, networking) and the WPAR resource. The WPAR resource is how VCS represents the WPAR and its state. You then configure the service group's ContainerInfo attribute.

Configuring the ContainerInfo attribute

The service group attribute ContainerInfo specifies information about the WPAR. When you have configured and enabled the ContainerInfo attribute, you have enabled the WPAR-aware resources in the service group to work in the WPAR environment.

VCS defines the WPAR information at the level of the service group so that you do not have to define it for each resource. You can specify a per-system value for the ContainerInfo attribute.

Running VCS, its resources, and your applications

VCS and the necessary agents run in the global environment. For applications that run in a WPAR, the agents can run some of their functions (entry points) inside the WPAR. If any resource faults, VCS fails over the service group with the WPAR to another node.

The ContainerInfo attribute

The ContainerInfo attribute has the Name key, Type key, and Enabled key. The Name key defines the name of the WPAR. The Type key lets you select the type of container that you plan to use (WPAR or Zone). The Enabled key enables the WPAR-aware resources within the service group.

You can specify a per-system value for the ContainerInfo attribute.

See [“Service group attributes”](#) on page 663.

The ContainerOpts resource attribute

The ContainerOpts resource attribute determines the following:

- If the resource has this attribute, it is “WPAR-aware”
- Whether the WPAR-aware resource can run in the WPAR
- Whether the container information that is defined in the service group’s ContainerInfo attribute is passed to the resource. An example use of this value is to pass the agent the name of the WPAR.

See [“Resource attributes”](#) on page 644.

Note: Symantec recommends that you do not modify the value of the ContainerOpts attribute, with the exception of the Mount agent.

WPAR-aware resources

[Table E-1](#) lists the ContainerOpts attribute default values for resource types. WPAR-aware resources have predefined default values for the ContainerOpts attribute.

Table E-1 ContainerOpts attribute default values for resource types

Resource Type	RunInContainer	PassCInfo
Application	1	0
DB2	1	0
IP	0	1
IPMultiNICB	0	1
Mount	0	0
Process	1	0
WPAR	0	1

About the Mount agent

You may need to modify the ContainerOpts values for the Mount resource in certain situations. Refer to the *Veritas Cluster Server Bundled Agents Reference Guide* for more information.

About the WPAR agent

The WPAR agent monitors WPARs, brings WPARs online, and takes them offline. For more information about the agent, see the *Veritas Cluster Server Bundled Agents Reference Guide*.

The agent creates a user account with group administrative privileges to enable communication between the global environment and the WPAR, if such an account doesn't exist. In secure clusters, it also renews the authentication certificate before the certificate expires.

Configuring VCS in WPARs

Configuring VCS in WPARs involves the following tasks:

First	Review the prerequisites. See “Prerequisites for configuring VCS in WPARs” on page 703.
Second	Decide on the location of the WPAR root, which is either on local storage or NFS. The WPAR root is the topmost directory in a section of the file system hierarchy in which the WPAR is configured. See “Creating WPAR root on shared storage using NFS” on page 706.
Third	Install the application in the WPAR. See “Installing the application” on page 707.
Fourth	Create the application service group and configure its resources. See “Configuring the service group for the application” on page 708.

Prerequisites for configuring VCS in WPARs

- All nodes that host applications that are configured in WPARs must run the same version of the operating system.
- The WPAR root must be installed on JFS, JFS2, or NFS.
- Mounts must meet one of the following two conditions:
 - Use a loop-back file system. All mounts that the application uses must be part of the WPAR configuration and must be configured in the service group. For example, you can create a WPAR, z-ora, and define the file system containing the application's data to have the mount point as /oradata. When you create the WPAR, you can define a path in the global environment, for example /export/home/oradata, which the mount directory in the WPAR maps to. The MountPoint attribute of the Mount resource for the application is set to /export/home/oradata.
 - Use a direct mount file system. All file system mount points that the application uses that run in a WPAR must be set relative to the WPAR's root. For example, if the Oracle application uses /oradata, and you create the WPAR with the WPAR path as /z_ora, then the mount must be /z_ora/root/oradata. The MountPoint attribute of the Mount resource must be set to this path.

Using custom agents in WPARs

- If you use custom agents to monitor applications running in WPARs, make sure the agents use script-based entry points. VCS does not support running C++ entry points inside a WPAR.
- If the custom agent monitors an application that runs in a WPAR, add the resource type to the APP_TYPES environment variable.
If the custom agent monitors an application running in the global environment, add the resource type to the SYS_TYPES environment variable.
- If you want the custom agent to monitor an application in the WPAR, for the custom agent type, set the following values for the ContainerOpts attribute: RunInContainer = 1 and the PassCInfo = 0.
- If you don't want the custom agent to monitor an application in the WPAR, for the custom agent type, set the following values for the ContainerOpts attribute: RunInContainer = 0 and the PassCInfo = 0.

Deciding on the WPAR root location

Each WPAR has its own section of the file system hierarchy in the WPAR root directory. Processes that run in the WPAR can access files only within the WPAR root. You can set the WPAR root in the following two ways:

- WPAR root on local storage.
In this configuration, you must create a WPAR on each node in the cluster.
- WPAR root on NFS.
In this configuration, create a WPAR on the NFS storage. You need to duplicate the configuration across all the nodes in the cluster.
When you set the WPAR root on NFS, install the WPAR from one node only. The WPAR root can fail over to the other nodes in the cluster. The system software, including the patches, must be identical on each node during the existence of the WPAR.

Creating a WPAR root on local disk

Use the following procedure to create a WPAR root on the local disk on each node in the cluster.

To create a WPAR root on local disks on each node in the cluster

- 1 Create the actual WPAR root directory.

- 2 Use the `mkwpar` command to create the WPAR.

```
mkwpar -n wpar -h host -N ip_info -d wroot -o /tmp/wpar.log
```

Use the following information to replace the appropriate variables:

`wpar` The name of the WPAR.

`hostname` The name of the system where the WPAR is created.

`ip_info` Replace this variable with the information to set the virtual IP address of the system to be the IP address of the WPAR. This value also defines the device name for the NIC associated with the IP address.

If you don't specify the value of the interface or netmask, the global partition's values are used.

Use the following format to replace `ip_info`:

```
interface=interface netmask=netmask address=IPaddress
```

For example:

```
interface='en0' address='172.16.0.0'  
netmask='255.255.255.0'
```

`wroot` Replace this variable with the location of the WPAR root directory, for example, `/wpars`.

- 3 Repeat the command in [step 2](#) to create the WPAR on each system in the service group's SystemList.
- 4 On one of the systems in the SystemList, mount the shared file system containing the application data.
- 5 Start the WPAR.

Creating WPAR root on shared storage using NFS

Use the following procedure to create a WPAR root on shared storage using NFS.

To create WPAR root on shared storage using NFS

- 1 Create a file system on NFS storage for the WPAR root. The file system that is to contain the WPAR root may be in the same file system as the file system containing the shared data.

- 2 Type the following `mkwpar` command to create the WPAR:

```
mkwpar -n wpar -h host -N ip_info -r -M r_fs -M v_fs  
-M h_fs -M t_fs -d wroot
```

Use the following information to replace the appropriate variables:

- | | |
|----------------------|--|
| <code>wpar</code> | The name of the WPAR. |
| <code>host</code> | The name of the system where the WPAR is created. |
| <code>ip_info</code> | Replace this variable with the information to set the virtual IP address of the system to be the IP address of the WPAR. This value also defines the device name for the NIC associated with the IP address. |

If you don't specify the value of the interface or netmask, the global partition's values are used.

Use the following format to replace `ip_info`:

```
interface=interface netmask=netmask address=IPaddress
```

For example:

```
interface='en0' address='172.16.0.0'  
netmask='255.255.255.0'
```

- | | |
|-------------------|---|
| <code>r_fs</code> | Replace this variable with the information to specify the NFS volume to use for the root private file system for the WPAR. For example: |
|-------------------|---|

```
directory=/ vfs=nfs host=host123 dev=/root01
```

- | | |
|-------------------|---|
| <code>v_fs</code> | Replace this variable with the information to specify the NFS volume to use for the /var private file system for the WPAR. For example: |
|-------------------|---|

```
directory=/ var vfs=nfs host=host123 dev=/var01
```

- | | |
|-------------------|---|
| <code>h_fs</code> | Replace this variable with the information to specify the NFS volume to use for the /home private file system for the WPAR. Examples of values: |
|-------------------|---|

```
directory=/home vfs=nfs host=host123 dev=/home01
```

- | | |
|-------------------|--|
| <code>t_fs</code> | Replace this variable with the information to specify the NFS volume to use for the root private file system for the WPAR. Examples of values: |
|-------------------|--|

```
directory=/tmp vfs=nfs host=host123 dev=/tmp01
```

wroot Replace this variable with the location of the WPAR root directory, for example, /wpars.

- 3 Use the `lswpar` command to display information about the WPAR's properties and their values.
- 4 On the system where you created the WPAR, run the command:
`mkwpar -w -o config_file_name -e wparname_just_created`
- 5 On all the other systems copy the configuration file, run the command:
`mkwpar -p -f config_file_name -n wparname_just_created`
- 6 List the WPAR.
- 7 Start the WPAR.
- 8 On one system, mount the shared file system containing the application data.
- 9 Make sure the WPAR created from the first system is in the D state on all other systems in the service group's System List.

Installing the application

Install the application in the WPAR. Perform the following:

- If you have created WPARs on each node in the cluster, install the application identically on all nodes. If you are installing an application that supports a Veritas High Availability agent, see the installation and configuration guide for the agent.
- Install the agent. Agent packages are installed in the global environment and the currently existing WPARs. The operating system installs the agents in future WPARs when they are created.
- In the WPAR, configure all mount points used by the application.
 - If you use `namefs` mounts, verify the global directories are properly mounted inside the WPAR.
 - If you use a direct mount, verify the mount points used by the application have been mounted relative to the WPAR's root. For example, if a WPAR `w_oracle` needs to use `/oracle`, mount the drive at `/wpars/w_oracle/oracle`.

Configuring the service group for the application

The following diagrams illustrates different examples of resource dependencies. In one case the WPAR root is set up on local storage. In the other, WPAR root is set up on shared storage.

Resource dependency diagrams: WPAR root on local disks

Figure E-1 WPAR root on local disks (with direct mount file system)

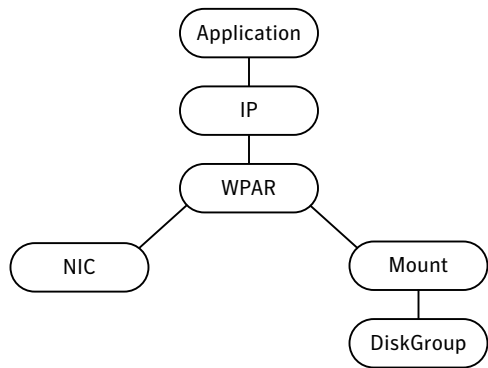
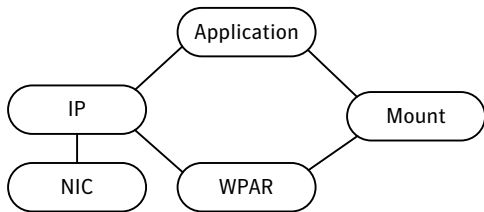
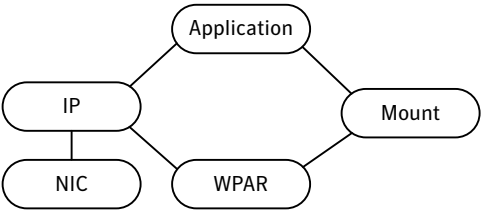


Figure E-2 WPAR root on local disks (file system mounted from inside WPAR)



Resource dependency diagrams: WPAR root on shared storage

Figure E-3 WPAR root on shared storage (with namefs file system)



Modifying the service group configuration

Perform the following procedure to modify a service group’s configuration.

To modify the service group configuration

- 1 Run the hawparsetup script to set up the WPAR configuration.

```
# hawparsetup servicegroup_name WPARres_name WPAR_name password systems
```

<i>servicegroup_name</i>	Name of the application service group.
<i>WPARres_name</i>	Name of the resource configured to monitor the WPAR.
<i>WPAR_name</i>	Name of the WPAR.
<i>password</i>	Password to be assigned to VCS or Security (Symantec Product Authentication Service) user created by the command.
<i>systems</i>	List of systems on which the service group will be configured. Use this option only when creating the service group.

The script adds a resource of type WPAR to the application service group. It also creates a user account with group administrative privileges to enable inter-WPAR communication.
If the application service group does not exist, the script creates a service group with a resource of type WPAR.

- 2 Modify the resource dependencies to reflect your WPAR configuration. See the resource dependency diagrams for more information.
- 3 Save the service group configuration and bring the service group online.

Verifying the WPAR configuration

Run the `hawparverify` command to verify the WPAR configuration. The command verifies the following requirements:

- The systems hosting the service group have the required operating system to run WPARs.
- The service group does not have more than one resource of type WPAR.
- The dependencies of the WPAR resource are correct.

To verify the WPAR configuration

- 1 If you use custom agents make sure the resource type is added to the `APP_TYPES` or `SYS_TYPES` environment variable.
See “[Using custom agents in WPARs](#)” on page 703.
- 2 Run the `hawparverify` command to verify the WPAR configuration.
`# hawparverify servicegroup_name`

Maintenance tasks

Perform the following maintenance tasks when you use WPARs:

- Whenever you make a change that affects the WPAR configuration, you must run the `hawparsetup` command to reconfigure the WPARs in VCS.
See “[Configuring the service group for the application](#)” on page 708.
- Make sure that the WPAR configuration files are consistent on all the nodes at all times.
- When you add a patch or upgrade the operating system on one node, make sure to upgrade the software on all nodes.
- Make sure that the application configuration is identical on all nodes. If you update the application configuration on one node, apply the same updates to all nodes.

Troubleshooting information

VCS HA commands do not work.

Recommended Action:

Verify the VCS packages are installed.

Run the `hawparsetup` script to set up the WPAR configuration. Run the `hawparverify` command to verify the configuration.

Run the `halogin` command from the WPAR.

See [“Logging on to VCS”](#) on page 196.

Verify your VCS credentials. Make sure the password is not changed.

Verify the VxSS certificate is not expired.

Resource does not come online in the WPAR.

Recommended Action:

Verify VCS and the agent packages are installed correctly. Verify the application is installed in the WPAR.

Verify the configuration definition of the agent. Make sure to define the `ContainerType` and `ContainerName` attributes.

About VCS support for Live Partition Mobility

You can use Live Partition Mobility to perform a stateful migration of an LPAR in a VCS environment. During this period, you may see notifications if the migrating node is unable to heartbeat with its peers within LLT's default peer inactive timeout. To avoid false failovers, determine how long the migrating node is unresponsive in your environment. If that time is less than the default LLT peer inactive timeout, VCS operates normally. If not, increase the peer inactive timeout to an appropriate value on all the nodes in the cluster before beginning the migration. Reset the value back to the default after the migration is complete.

See [“When a system fails”](#) on page 571.

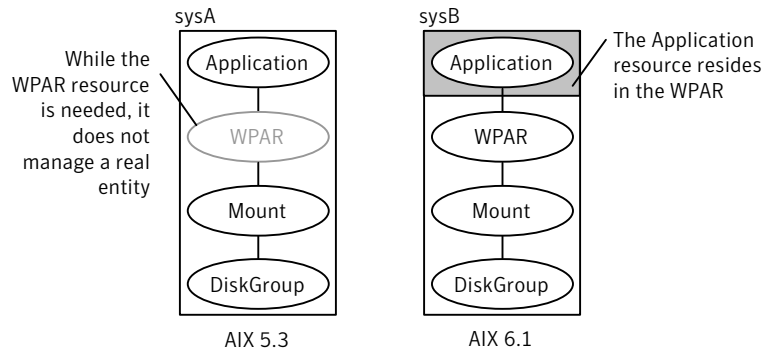
About configuring failovers among physical and virtual servers

You can configure VCS to fail over from a physical system to a virtual system and vice versa. A physical to virtual failover gives an N + N architecture in an N + 1 environment. For example, several physical servers with applications can fail over to containers on another physical server.

Configuring for failovers—a typical setup

In this configuration, you have two physical nodes. One node runs AIX 5.3 (sysA) and another node that runs AIX 6.1 (sysB). The node that runs AIX 6.1 has zones configured.

Figure E-4 An application service group that can fail over onto a WPAR



```
ContainerInfo@sysA = {Name = W1 Type = WPAR Enabled = 2}  
ContainerInfo@sysB = {Name = W1 Type = WPAR Enabled = 1}
```

In the `main.cf` configuration file, define the container name, type of container, and whether it is enabled or not. The following is an example of the `ContainerInfo` lines in the `main.cf` file:

```
ContainerInfo@sysA = {Name = W1, Type = WPAR, Enabled = 2}  
ContainerInfo@sysB = {Name = W1, Type = WPAR, Enabled = 1}
```

On `sysA`, you set the value of `Enabled` to 2 to ignore zones so that the application runs on the physical system. When `sysA` fails over to `sysB`, the application runs inside the zone after the failover because `Enabled` is set to 1 on `sysB`. The application can likewise fail over to `sysA` from `sysB`.

Accessibility and VCS

- [About accessibility in VCS](#)
- [Navigation and keyboard shortcuts](#)
- [Support for accessibility settings](#)
- [Support for assistive technologies](#)

About accessibility in VCS

Symantec products meet federal accessibility requirements for software as defined in Section 508 of the Rehabilitation Act:
<http://www.access-board.gov/508.htm>
Veritas Cluster Server provides shortcuts for major graphical user interface (GUI) operations and menu items. Veritas Cluster Server is compatible with operating system accessibility settings as well as a variety of assistive technologies. All manuals also are provided as accessible PDF files, and the online help is provided as HTML displayed in a compliant viewer.

Navigation and keyboard shortcuts

VCS uses standard operating system navigation keys and keyboard shortcuts. For its unique functions, VCS uses its own navigation keys and keyboard shortcuts which are documented below.

Navigation in the Java Console

The following table lists keyboard navigation rules and shortcuts used in Cluster Manager (Java Console), in addition to those provided by the operating system:

VCS Keyboard Input	Result
[Shift F10]	Opens a context-sensitive pop-up menu
[Spacebar]	Selects an item
[Ctrl Tab]	Navigates outside a table
[F2]	Enables editing a cell

Navigation in the Web Console

- Cluster Management Console supports standard browser-based navigation and shortcut keys for supported browsers.
- All Symantec GUIs use the following keyboard navigation standards:
- Tab moves the focus to the next active area, field, or control, following a preset sequence. Shift+Tab moves the focus in the reverse direction through the sequence.
 - Ctrl+Tab exits any Console area that you internally navigate with Tab.

- Up and Down arrow keys move focus up and down the items of a list.
- Alt in combination with the underlined mnemonic letter for a field or command button shifts the focus to that field or button.
- Either Enter or the Spacebar activates your selection. For example, after pressing Tab to select Next in a wizard panel, press the Spacebar to display the next screen.

Support for accessibility settings

Symantec software responds to operating system accessibility settings. On UNIX systems, you can change the accessibility settings using desktop preferences or desktop controls.

Support for assistive technologies

- Cluster Manager (Java Console) is compatible with JAWS 4.5.
- Though graphics in the documentation can be read by screen readers, setting your screen reader to ignore graphics may improve performance.
- Symantec has not tested screen readers for languages other than English.

Glossary

Agent

A process that starts, stops, and monitors all configured resources of a type, and reports their status to VCS.

Active/Active Configuration

A failover configuration where each system runs a service group. If either fails, the other one takes over and runs both service groups. Also known as a symmetric configuration.

Active/Passive Configuration

A failover configuration consisting of one service group on a primary system, and one dedicated backup system. Also known as an asymmetric configuration.

Authentication Broker

The Veritas Security Services component that serves, one level beneath the root broker, as an intermediate registration authority and a certification authority. The authentication broker can authenticate clients, such as users or services, and grant them a certificate that will become part of the Veritas credential. An authentication broker cannot, however, authenticate other brokers. That task must be performed by the root broker.

See “[Root Broker](#).”

Cluster

One or more computers linked together for the purpose of multiprocessing and high availability. The term is used synonymously with VCS cluster, meaning one or more computers that are part of the same GAB membership.

Daemon Down Node Alive (DDNA)

A situation where the VCS high availability daemon has failed on a system and has not been restarted by the hashadow process.

Disaster Recovery

A solution that supports fail over to a cluster in a remote location in the event that the local cluster becomes unavailable. Disaster recovery global clustering, heartbeating, and replication.

Failover

A failover occurs when a service group faults and is migrated to another system.

GAB

Group Atomic Broadcast (GAB) is a communication mechanism of the VCS engine that manages cluster membership, monitors heartbeat communication, and distributes information throughout the cluster.

Global Service Group

A VCS service group that spans across two or more clusters. The `ClusterList` attribute for the group contains the list of clusters over which the group spans.

hashadow Process

A process that monitors and, when required, restarts HAD.

High Availability Daemon (HAD)

The core VCS process that runs on each system. The HAD process maintains and communicates information about the resources running on the local system and receives information about resources running on other systems in the cluster.

Jeopardy

A node is in *jeopardy* when it is missing one of the two required heartbeat connections. When a node is running with one heartbeat only (in jeopardy), VCS does *not* restart the applications on a new node. This action of disabling failover is a safety mechanism that prevents data corruption.

LLT

Low Latency Transport (LLT) is a communication mechanism of the VCS engine that provides kernel-to-kernel communications and monitors network communications.

main.cf

The file in which the cluster configuration is stored.

Network Partition

If all network connections between any two groups of systems fail simultaneously, a *network partition* occurs. When this happens, systems on both sides of the partition can restart applications from the other side resulting in duplicate services, or “split-brain.” A split-brain occurs when two independent systems configured in a cluster assume they have exclusive access to a given resource (usually a file system or volume). The most serious problem caused by a network partition is that it affects the data on shared disks.

See “[Jeopardy](#).”

See “[Seeding](#).”

Node

The physical host or system on which applications and service groups reside. When systems are linked by VCS, they become nodes in a cluster.

N-to-1

An N-to-1 configuration is based on the concept that multiple, simultaneous server failures are unlikely; therefore, a single backup server can protect multiple active servers. When a server fails, its applications move to the backup server. For example, in a 4-to-1 configuration, one server can protect four servers, which reduces redundancy cost at the server level from 100 percent to 25 percent.

N-to-N

N-to-N refers to multiple service groups running on multiple servers, with each service group capable of being failed over to different servers in the cluster. For example, consider a four-node cluster with each node supporting three critical database instances. If any node fails, each instance is started on a different node, ensuring no single node becomes overloaded.

N-to-M

N-to-M (or Any-to-Any) refers to multiple service groups running on multiple servers, with each service group capable of being failed over to different servers in the same cluster, and also to different servers in a linked cluster. For example, consider a four-node cluster with each node supporting three critical database instances and a linked two-node back-up cluster. If all nodes in the four-node cluster fail, each instance is started on a node in the linked back-up cluster.

Replication

Replication is the synchronization of data between systems where shared storage is not feasible. The systems that are copied may be in local backup clusters or remote failover sites. The major advantage of replication, when compared to traditional backup methods, is that current data is continuously available.

Resources

Individual components that work together to provide application services to the public network. A resource may be a physical component such as a disk or network interface card, a software component such as Oracle8i or a Web server, or a configuration component such as an IP address or mounted file system.

Resource Dependency

A dependency between resources is indicated by the keyword “requires” between two resource names. This indicates the second resource (the child) must be online before the first resource (the parent) can be brought online. Conversely, the parent must be offline before the child can be taken offline. Also, faults of the children are propagated to the parent.

Resource Types

Each resource in a cluster is identified by a unique name and classified according to its type. VCS includes a set of predefined resource types for storage, networking, and application services.

Root Broker

The first authentication broker, which has a self-signed certificate. The root broker has a single private domain that holds only the names of brokers that shall be considered valid. See “[Authentication Broker](#).”

Seeding

Seeding is used to protect a cluster from a pre-existing network partition. By default, when a system comes up, it is not seeded. Systems can be seeded automatically or manually. Only systems that have been seeded can run VCS. Systems are seeded automatically only when: an unseeded system communicates with a seeded system or all systems in the cluster are unseeded and able to communicate with each other.

See “[Network Partition](#).”

Service Group

A service group is a collection of resources working together to provide application services to clients. It typically includes multiple resources, hardware- and software-based, working together to provide a single service.

Service Group Dependency

A mechanism by which two service groups can be linked by a dependency rule.

Shared Storage

Storage devices that are connected to and used by two or more systems.

SNMP Notification

Simple Network Management Protocol (SNMP) developed to manage nodes on an IP network.

State

The current activity status of a resource, group or system.

types.cf

The types.cf file describes standard resource types to the VCS engine; specifically, the data required to control a specific resource.

Virtual IP Address

A unique IP address that associated with the cluster. It may be brought up on any system in the cluster, along with the other resources of the service group. This address, also known as the IP alias should not be confused with the base IP address, which is the IP address that corresponds to the host name of a system.

Index

A

- accessibility
 - assistive technology support 715
 - overview 713
- ActionTimeout attribute 650
- ActiveCount attribute 663
- AdministratorGroups attribute
 - for clusters 688
 - for service groups 663
- Administrators attribute
 - for clusters 688
 - for service groups 663
- agent log
 - format 586
 - location 586
- AgentClass attribute 650
- AgentDirectory attribute 651
- AgentFailedOn attribute 651
- AgentFile attribute 651
- AgentPriority attribute 651
- AgentReplyTimeout attribute 651
- agents
 - classifications of 29
 - DNS 471
 - entry points 28
 - framework 29
 - functions 28
 - Heartbeat 469
 - impact on performance 565
 - RVG 471
 - RVGPrimary 471
 - RVGSnapshot 471
 - starting from command line 223
 - stopping from command line 223
 - Wide-Area Heartbeat 469
- AgentStartTimeout attribute 652
- AgentState attribute 697
- AgentStopped attribute 680
- aggregate notifications, monitoring 443
- alerts
 - deleting from Java Console 176
 - monitoring from Java Console 175
- ArgList attribute 652
- ArgListValues attribute 644
- assistive technology support 715
- association attribute dimension 58
- asymmetric configuration 38
- AttrChangedTimeout attribute 652
- attribute dimensions
 - association 58
 - keylist 58
 - scalar 58
 - vector 58
- attribute types
 - boolean 57
 - integer 57
 - string 57
- attributes
 - about 57
 - editing from Java Console 167
 - for clusters 688
 - for heartbeats 697
 - for resource types 650
 - for resources 644
 - for service groups 663
 - for systems 680
 - local and global 60
 - overriding from command line 240
 - overriding from Java Console 152
- authentication broker 33
- Authority attribute
 - about 469
 - definition 663
- AutoDisabled attribute 664
- AutoFailOver attribute
 - about 365
 - definition 664
- AutoRestart attribute 665
- AutoStart attribute
 - for resources 644
 - for service groups 665
- AutoStartIfPartial attribute 665
- AutoStartList attribute 665

AutoStartPolicy attribute 666
 AutoStartTimeout attribute 688
 AvailableCapacity attribute 680

B

BackupInterval attribute 688
 binary message catalogs
 about 588
 location of 588
 boolean attribute type 57
 bundled agents 29

C

Capacity attribute 680
 Chapter 15, 69, 279, 427
 CleanRetryLimit attribute 652
 CleanTimeout attribute 652
 client process, detecting failure 572
 CloseTimeout attribute 652
 ClusState attribute 689
 Cluster Administrator
 about 72
 adding user as 207
 cluster attributes 688
 Cluster Explorer
 about 92
 accessing 92
 adding resources 143
 adding service groups 121
 adding systems 163
 adding users 117
 autoenabling service groups 134
 bringing resources online 148
 bringing service groups online 126
 changing user passwords 118
 changing user privileges 119
 clearing resource faults 155
 clearing ResourceInfo attribute 161
 closing configuration files 166
 Cluster Query 109
 Command Center 107
 configuration tree 95
 deleting resources 148
 deleting service groups 125
 deleting users 118
 disabling resources 154
 disabling service groups 133
 editing attributes 167

enabling resources 153
 enabling service groups 132
 flushing service groups 135
 freezing service groups 130
 freezing systems 164
 importing resource types 161
 linking resources 156
 linking service groups 136
 logs 173
 modifying system lists for service groups 105
 monitoring group dependencies 98
 monitoring resource dependencies 99
 Notifier Wizard 108
 opening configuration files 165
 probing resources 151
 Properties view 97
 refreshing ResourceInfo attribute 161
 Remote Cluster Status View 103
 Resource View 99
 running HA fire drill 161
 saving configuration files 165
 service group configuration wizard 140
 Service Group View 98
 Status View 96
 switching service groups 129
 System Connectivity View 102
 System Manager 105
 taking resources offline 149
 taking resources offline and propagating 150
 taking service groups offline 128
 tear-off view 95
 Template View 104
 toolbar 93
 unfreezing service groups 131
 unfreezing systems 164
 unlinking resources 158
 unlinking service groups 138
 User Manager 106
 view panel 95

Cluster Guest

about 73
 adding user as 207

Cluster Manager (Java Console). See Java Console

Cluster Monitor

about 86
 adding clusters 112
 administering 112
 behavior during failover 88
 collapsing displays 89

- configuring existing panels 113
 - configuring new panels 112
 - expanding displays 89
 - icon colors 88
 - logging off of a cluster 115
 - logging on to a cluster 114
 - menus 87
 - monitoring cluster connection 88
 - monitoring cluster objects 88
 - panels 88
 - pausing scrolling panels 89
 - toolbar 87
- cluster name, changing in global configuration 537
- Cluster Operator
 - about 73
 - adding user as 207
- Cluster Query
 - in Java Console 109
- ClusterAddress attribute 689
- ClusterFailOverPolicy attribute 666
- clustering
 - criteria for data storage 21
 - criteria for monitor procedure 20
 - criteria for start procedure 20
 - criteria for stop procedure 20
 - license and host name issues 22
- ClusterList attribute 666
- ClusterLocation attribute 689
- ClusterName attribute 690
- ClusterOwner attribute 690
- clusters
 - administering from Java Console 165
 - connecting to Cluster Monitor 112
- ClusterTime attribute 690
- ClusterUUID attribute 690
- Command Center
 - accessing 107
 - adding resources 144
 - adding service groups 122
 - adding systems 163
 - autoenabling service groups 134
 - bringing resources online 148
 - bringing service groups online 127
 - clearing resource faults 155
 - closing configuration files 166
 - deleting resources 148
 - deleting service groups 125
 - deleting systems 163
 - disabling resources 154
 - disabling service groups 133
 - editing attributes 167
 - enabling resources 153
 - enabling service groups 132
 - executing commands 166
 - flushing service groups 135
 - freezing service groups 130
 - freezing systems 164
 - ignoreparent option 150
 - linking resources 156
 - linking service groups 137
 - opening configuration files 165
 - probing resources 151
 - saving configuration files 165
 - switching service groups 129
 - taking resources offline 149
 - taking resources offline and propagating 150
 - taking service groups offline 129
 - unfreezing service groups 131
 - unfreezing systems 164
 - unlinking resources 159
 - unlinking service groups 138
- commands
 - vxfenadm 336
 - vxfcntlpre 340, 604
 - vxkeyless 186
- commands, scripting 241
- CompareRSM attribute 690
- ComputeStats attribute 645
- conditional statements 216
- ConfidenceLevel attribute 645
- ConfigBlockCount attribute 680
- ConfigChecksum attribute 680
- ConfigDiskState attribute 680
- ConfigFile attribute 680
- ConfigInfoCnt attribute 681
- ConfigModDate attribute 681
- configuration
 - closing from Java Console 166
 - dumping 199
 - opening from Java Console 165
 - saving 199
 - saving from Java Console 165
 - saving in VCS Simulator 272
 - setting to read/write 199
 - setting to read-only 199
 - taking snapshots of 199
 - verifying 198
- configuration files

- generating 52
- main.cf 52
- read/write to read-only 208, 210, 211, 212, 213, 214, 219, 220, 221
- restoring from snapshots 199
- taking snapshots of 199
- types.cf 52
- configuration language
 - local and global attributes 60
- configurations
 - asymmetric 38
 - global cluster 49
 - N+1 42
 - N-to-1 40
 - N-to-N 44
 - replicated data 48
 - shared nothing 47
 - shared storage/replicated data 48
 - symmetric 39
- ConfInterval attribute
 - about 374
 - definition 652
- ConnectorState attribute 690
- ContainerInfo attribute 667
- ContainerOpts attribute 645
- coordinator disks 292
- CounterInterval attribute 690
- CPU usage, how VCS monitors 578
- CPUUsage attribute 681
- CPUUsageMonitoring attribute 681
- Critical attribute 645
- CurrentCount attribute 667
- CurrentLimits attribute 681
- custom agents
 - in zones 703
- custom agents, about 29

D

- Daemon Down Node Alive 319
- DDNA 319
- DeferAutoStart attribute 667
- DeleteOnlineResource attribute 691
- dependencies
 - for resources 24
 - for service groups
- disability compliance
 - in Java Console 80
- DiskHbStatus attribute 682
- disks
 - verifying node access 339
- DNS agent 471
- dumping a configuration 199
- DumpingMembership attribute 691
- DynamicLoad attribute 682

E

- Enabled attribute
 - for resources 646
 - for service groups 667
- engine log
 - format 586
 - location 586
- EnginePriority attribute 691
- enterprise agents, about 29
- entry points
 - about 28
 - modifying for performance 566
- environment variables 62
- EPClass attribute
 - definition 653
- EPPriority attribute
 - definition 654
- error messages
 - agent log 586
 - at startup 594
 - engine log 586
 - message catalogs 588
- Evacuate attribute 667
- Evacuating attribute 667
- event triggers
 - about 446
 - in jeopardy 448
 - loadwarning 449
 - location of 446
 - multinicb 450
 - nfs_preonline 451
 - nofailover 452
 - postoffline 452
 - postonline 452
 - preonline 454, 455
 - resadminwait 456
 - resfault 457
 - resnotoff 457
 - resstatechange 459
 - sysoffline 460, 461
 - sysup 460
 - unable_to_restart_had 461, 462
 - using 446

violation 462
 ExternalStateChange attribute 655

F

failback, about 41
 Failover attribute 668
 FailOverPolicy attribute 668
 FaultOnMonitorTimeouts attribute 655
 FaultPropagation attribute
 for resource types 655, 668
 fire drills
 about 492
 for global clusters 492
 for replicated data clusters 546
 FireDrill attribute 656
 Flags attribute 646
 FromQ attribute 668
 Frozen attribute
 for service groups 668
 for systems 682

G

GAB
 about 284
 impact on performance 564
 when a system panics 572
 GAB, about 30
 gab_isolate_time timer 572
 global attributes 60
 global cluster configuration 49
 global clusters
 adding from Java Console 509
 bringing remote groups online 518
 deleting from Java Console 513
 operation 466
 prerequisites for 474
 setting up 478
 switching remote groups 519
 user privileges 75
 global heartbeats
 administering from command line 538
 administering from Java Console 520
 deleting from Java Console 523
 modifying from Java Console 522
 global service groups
 administering from command line 533
 administering from Java Console 516
 querying from command line 526

GlobalCounter attribute 692
 Group Administrator
 about 73
 adding user as 207
 Group attribute 646
 group dependencies. *See* service group dependencies
 Group Membership Services/Atomic Broadcast (GAB) 30
 Group Operator
 about 73
 adding user as 208
 GroupLimit attribute 692
 GroupOwner attribute 669
 Guests attribute
 for clusters 692
 for service groups 669
 GUI. *See* Java Console or Cluster Management Console
 GUIPAddr attribute 682

H

HA fire drill
 about 263
 haagent -display command 212
 haagent -list command 216
 haattr -add command 225
 haattr -default command 226
 haattr -delete command 226
 hacf utility
 about 198
 creating multiple .cf files 198
 loading a configuration 198
 pretty-printing 198
 hacf -verify command 198
 HacliUserLevel attribute
 about 72
 definition 693
 haclus -add command 536
 haclus -declare command 536
 haclus -delete command 536
 haclus -display command
 for global clusters 530
 for local clusters 213
 haclus -list command 530
 haclus -modify command 536
 haclus -state command 530
 haclus -status command 530
 haclus -value command

- for global clusters 530
 - for local clusters 213
- haclus -wait command 241
- haconf -dump -makero command 199
- haconf -makerw command 199
- HAD
 - about 30
 - impact on performance 565
- HAD diagnostics 588
- had -v command 233
- had -version command 233
- hagrp -add command 217
- hagrp -clear command 221
- hagrp -delete command 217
- hagrp -dep command 210
- hagrp -disable command 220
- hagrp -disableresources command 221
- hagrp -display command
 - for global clusters 526
 - for local clusters 210
- hagrp -enable command 220
- hagrp -enableresources command 221
- hagrp -freeze command 220
- hagrp -link command 223
- hagrp -list command
 - for global clusters 527
 - for local clusters 216
- hagrp -modify command 217
- hagrp -offline command
 - for global clusters 533
 - for local clusters 219
- hagrp -online command
 - for global clusters 533
 - for local clusters 219
- hagrp -resources command 210, 527
- hagrp -state command
 - for global clusters 526
 - for local clusters 210
- hagrp -switch command
 - for global clusters 534
 - for local clusters 219
- hagrp -unfreeze command 220
- hagrp -unlink command 223
- hagrp -value command 526
- hagrp -wait command 241
- hahb -add command 538
- hahb -display command 531
- hahb -list command 531
- halogin command 196, 197
- hamsg -info command 214
- hamsg -list command 214
- hanotify utility 434
- hares -action command 535
- hares -add command 224
- hares -clear command 230
- hares -delete command 225
- hares -dep command 211
- hares -display command
 - for global clusters 528
 - for local clusters 211
- hares -global command 211
- hares -info command 535
- hares -link command 229
- hares -list command
 - for global clusters 528
 - for local clusters 216
- hares -local command 227
- hares -modify command 225
- hares -offline command 229
- hares -offprop command 230
- hares -online command 229
- hares -override command 240
- hares -probe command 230
- hares -state command 528
- hares -undo_override command 240
- hares -unlink command 229
- hares -value command 528
- hares -wait command 241
- hashadow process 30
- hasnap -backup command 201
- hasnap -delete command 206
- hasnap -exclude command 205
- hastart command 191
- hastart -onenode command 191
- hastatus command
 - for global clusters 530
 - for local clusters 213
- hastatus -group command 213, 214
- hastatus -summary command 214
- hastop command 193
- hasys -display command
 - for global clusters 529
 - for local clusters 213
- hasys -freeze command 231
- hasys -list command
 - for global clusters 529
 - for local clusters 213
- hasys -modify command 231

- hasys -nodeid command 231
- hasys -state command 529
- hasys -unfreeze command 231, 233
- hasys -value command
 - for global clusters 529
- hasys -wait command 241
- hatype -add command 239
- hatype -delete command 240
- hatype -display command 212
- hatype -list command 212
- hatype -modify command 240
- hatype -resources command 212
- hauser -add command 208
- hauser -addpriv command 208
- hauser -delete command 209
- hauser -delpriv command 208, 209
- hauser -display command 209
- hauser -list command 209
- Heartbeat agent 469
- heartbeat attributes 697
- heartbeats, modifying for global clusters 538
- host name issues 22
- HostMonitor
 - about 30
- HostMonitor attribute 683
- HostUtilization attribute 683

I

- I/O fencing
 - event scenarios 313
 - testing and scenarios 313
- I/O fencing, about 32
- icons
 - colors of 88
 - in Java Console 84
- include clauses, about 53
- InfoInterval attribute 656
- InfoTimeout attribute 656
- injeopardy event trigger 448
- integer attribute type 57
- IntentOnline attribute 670
- Istate attribute 647

J

- Java Console
 - about 34
 - administering clusters 80
 - administering logs 173

- administering resources 143
- administering service groups 121
- administering systems 163
- administering user profiles 117
- administering VCS Simulator 267
- arranging icons 100
- Cluster Explorer 92
- Cluster Manager 84
- Cluster Monitor 86
- Cluster Query 109
- components of 84
- customizing display 90
- disability compliance 80
- icons 84
- impact on performance 567
- logging off of a cluster 115
- logging on to a cluster 114
- overview 80
- running commands from 166
- running virtual fire drill 161
- setting initial display 81
- starting 83
- user profiles 117
- using with ssh 82
- viewing server credentials 111
- viewing user credentials 111

Java Console views

- Properties 97
- Remote Cluster Status 103
- Resource 99
- Service Group 98
- Status 96
- System Connectivity 102
- tear-off option 95

K

- keylist attribute dimension 58
- keywords 61
- keywords, list of 61

L

- LastOnline attribute 647
- LastSuccess attribute 670
- license keys
 - about 185
 - installing 185
 - troubleshooting 618
- LicenseType attribute 683

- licensing issues 22
- Limits attribute 683
- LinkHbStatus attribute 684
- LLT 31
- LLT, about 284
- LLTNodeId attribute 684
- Load attribute 670
- Load policy for SGWM 366
- LoadTimeCounter attribute 684
- LoadTimeThreshold attribute 684
- loadwarning event trigger 449
- LoadWarningLevel attribute 684
- local attributes 60
- LockMemory attribute 693
- LogDbg attribute 657
- LogFileSize attribute 657
- logging
 - agent log 586
 - engine log 586
 - message tags 586
- logs
 - customizing display in Java Console 173
 - searching from Java Console 173
 - viewing from Java Console 110
- LogSize attribute 693
- Low Latency Transport (LLT) 31

M

- main.cf
 - about 52
 - cluster definition 53
 - group dependency clause 54
 - include clauses 53
 - resource definition 53
 - resource dependency clause 53
 - sample configuration 54
 - service group definition 53
 - system definition 53
- ManageFaults attribute
 - about 368
 - definition 670
- ManualOps attribute 670
- message tags, about 586
- MigrateQ attribute 671
- MonitorInterval attribute 657
- MonitorOnly attribute 647
- MonitorStartParam attribute 658
- MonitorTimeout attribute 658
- MonitorTimeStats attribute 648

- multinicb event trigger 450

N

- N+1 configuration 42
- Name attribute 648
- network failure 102
- network links, detecting failure 571
- networks, detecting failure 573
- nfs_preonline event trigger 451
- NoAutoDisable attribute 685
- NodeId attribute 685
- nofailover event trigger 452
- notification
 - about 430
 - deleting messages 432
 - error messages 431
 - error severity levels 431
 - event triggers 446
 - hanotify utility 434
 - message queue 431
 - notifier process 432
 - setting using wizard 170
 - SNMP files 439
 - troubleshooting 613
- Notifier attribute 694
- notifier process 432
- Notifier Resource Configuration wizard 170
- N-to-1 configuration 40
- N-to-N configuration 44
- NumRetries attribute 671
- NumThreads attribute
 - definition 658
 - modifying for performance 566

O

- OfflineMonitorInterval attribute 659
- OfflineProcScanInterval attribute 659
- OfflineTimeout attribute 659
- OfflineWaitLimit attribute 659
- OnGrpCnt attribute 685
- OnlineAtUnfreeze attribute 671
- OnlineClass attribute
 - definition 659
- OnlinePriority attribute
 - definition 660
- OnlineRetryInterval attribute 671
- OnlineRetryLimit attribute
 - for resource types 660

- for service groups 671
- OnlineTimeout attribute 661
- OnlineWaitLimit attribute 661
- On-Off resource 24
- On-Only resource 24
- OpenTimeout attribute 661
- Operations attribute 661
- OperatorGroups attribute
 - for clusters 694
 - for service groups 671
- Operators attribute
 - for clusters 694
 - for service groups 672
- overload warning for SGWM 391

P

- PanicOnNoMem attribute 694
- Parallel attribute 672
- passwords
 - changing from Java Console 118
- Path attribute 648
- PathCount attribute 672
- performance
 - agents 565
 - GAB 564
 - HAD 565
 - impact of VCS 564
 - Java Console 567
 - modifying entry points 566
 - modifying NumThreads attribute 566
 - monitoring CPU usage 578
 - when a cluster is booted 568
 - when a resource fails 569
 - when a resource is brought online 568
 - when a resource is taken offline 569
 - when a service group fails over 574
 - when a service group is brought online 569
 - when a system fails 571
- Persistent resource 24
- postoffline event trigger 452
- postonline event trigger 452
- PreOnline attribute 672
- preonline event trigger 454, 455
- PreOnlineTimeout attribute 673
- PreOnlining attribute 672
- Prerequisites attribute 673
- PreSwitch attribute 674
- PreSwitching attribute 674
- pretty-printing 198

- PrintMsg attribute 694
- PrintTree attribute 674
- priorities
 - defaults 576
 - ranges 576
 - scheduling 574
 - specifying 576
- Priority attribute 675
- priority ranges for sched. classes 574
- privileges. See user privileges
- Probed attribute
 - for resources 648
 - for service groups 675
- ProbesPending attribute 675
- ProcessPriority attribute 694
- ProcScanInterval attribute 661

Q

- quick reopen 573

R

- ReadOnly attribute 695
- Recovering After a Disaster 545
- Remote Cluster Configuration wizard 509
- Remote Cluster States 636
- remote clusters
 - monitoring from Java Console 103
- replicated data clusters
 - about 48
 - setting up 542
- replicated data configuration 48
- resadminwait event trigger 456
- reserved words 61
- reserved words, list of 61
- resfault event trigger 457
- resnotoff event trigger 457
- resource attributes 644
- resource dependencies
 - creating from command line 229
 - creating from Java Console 156
 - displaying from command line 211
 - removing from command line 229
 - removing from Java Console 158
- resource faults
 - clearing from Java Console 155
 - simulating 272
- resource type attributes 650
- resource types

- importing 161
- querying from command line 212
- ResourceInfo attribute
 - clearing from Java Console 161
 - definition 648
 - refreshing from Java Console 161
- ResourceLimit attribute 695
- ResourceOwner attribute 649
- resources
 - about 23
 - adding from command line 224
 - adding from Java Console 143
 - administering from Java Console 143
 - bringing online from command line 229
 - bringing online from Java Console 148
 - categories of 24
 - clearing faults from Java Console 155
 - creating faults in VCS Simulator 272
 - deleting from command line 225
 - deleting from Java Console 148
 - disabling from command line 383
 - disabling from Java Console 154
 - enabling from command line 221
 - enabling from Java Console 153
 - how disabling affects states 385
 - invoking actions 160
 - limitations of disabling 383
 - linking from command line 229
 - linking from Java Console 156
 - On-Off 24
 - On-Only 24
 - Persistent 24
 - probing from Java Console 151
 - querying from command line 211
 - taking offline from command line 229
 - taking offline from Java Console 149
 - troubleshooting 597
 - unlinking from command line 229
 - unlinking from Java Console 158
- Responding attribute 675
- resstatechange event trigger 459
- Restart attribute 675
- RestartLimit attribute
 - about 373
 - definition 661
- root broker 33
- RVG agent 471
- RVGPrimary agent 471
- RVGSnapshot agent 471

S

- saving a configuration 199
- scalar attribute dimension 58
- scheduling classes 574
 - defaults 576
 - priority ranges 576
- ScriptClass attribute 661
- scripting VCS commands 241
- ScriptPriority attribute 662
- SCSI-III Persistent Reservations 307
- SecInfo attribute 695
- SecInfoLevel attribute 695
- secure VCS. See VERITAS Security Services
- SecureClus attribute 695
- server credentials, viewing 111
- service group attributes 663
- service group dependencies
 - about 409
 - autorestart 365
 - benefits of 410
 - categories of 411
 - creating 424
 - creating from Java Console 136
 - limitations of 414
 - manual switch 425
 - removing from Java Console 138
- service group workload management
 - Capacity and Load attributes 390
 - load policy 366
 - load-based autostart 367
 - overload warning 391
 - sample configurations 393
 - SystemZones attribute 367
- service groups
 - adding from command line 217
 - adding from Java Console 121
 - administering from command line 217
 - administering from Java Console 121
 - autoenabling from Java Console 134
 - bringing online from command line 219
 - bringing online from Java Console 126
 - creating using configuration wizard 140
 - deleting from command line 217
 - deleting from Java Console 125
 - disabling from Java Console 133
 - displaying dependencies from command line 210
 - enabling from Java Console 132
 - flushing from command line 222

- flushing from Java Console 135
 - freezing from command line 220
 - freezing from Java Console 130
 - linking from Java Console 136
 - querying from command line 210
 - switching from command line 219
 - switching from Java Console 129
 - taking offline from Java Console 128
 - taking remote groups offline 519
 - troubleshooting 595
 - unfreezing from command line 220
 - unfreezing from Java Console 131
 - unlinking from Java Console 138
 - shared nothing configuration 47
 - shared storage/replicated data configuration 48
 - ShutdownTimeout attribute 685
 - Signaled attribute 649
 - Simulator. See VCS Simulator
 - SNMP 430
 - files for notification 439
 - HP OpenView 439
 - merging events with HP OpenView NNM 439
 - SNMP, supported consoles 430
 - SourceFile attribute
 - for clusters 695
 - for resource types 662
 - for service groups 675
 - for systems 685
 - split-brain
 - in global clusters 472
 - ssh configuration for Java Console 82
 - Start attribute 649
 - State attribute
 - for resources 649
 - for service groups 676
 - steward process
 - about 472
 - Stewards attribute 695
 - string attribute type 57
 - SupportedActions attribute 662
 - Symantec Product Authentication Service
 - about 33
 - authentication broker 33
 - disabling 237
 - enabling 237
 - root broker 33
 - viewing credentials 111
 - symmetric configuration 39
 - SysInfo attribute 686
 - SysName attribute 686
 - sysoffline event trigger 460, 461
 - SysState attribute 686
 - System Attributes 680
 - system attributes 680
 - system states 638
 - SystemList attribute
 - about 54, 218
 - definition 676
 - modifying 218
 - SystemLocation attribute 686
 - SystemOwner attribute 686
 - systems
 - adding from command line 233, 234
 - adding from Java Console 163
 - administering from command line 231
 - administering from Java Console 163
 - bringing online in VCS Simulator 271
 - client process failure 572
 - deleting from Java Console 163
 - detecting failure 571
 - displaying node ID from command line 231
 - freezing from Java Console 164
 - panic 572
 - quick reopen 573
 - starting from command line 191
 - states 638
 - unfreezing from Java Console 164
 - systems and nodes 22
 - SystemZones attribute 676
 - sysup event trigger 460
- ## T
- Tag attribute 677
 - TargetCount attribute 677
 - templates
 - accessing Template View 104
 - adding resources from 145
 - adding service groups from 124
 - TFrozen attribute
 - for service groups 677
 - for systems 686
 - The 467
 - ToleranceLimit attribute 662
 - ToQ attribute 677
 - TriggerEvent attribute
 - for resources 650
 - for service groups 677
 - TriggerResFault attribute 677

TriggerResStateChange attribute 677

triggers. See event triggers

troubleshooting

back up and restore files 613

license keys 618

logging 586

notification 613

resources 597

service groups 595

VCS startup 594

TRSE attribute 686

TypeDependencies attribute 678

TypeLimit attribute 695

types.cf 52

U

umask, setting for VCS files 65

unable_to_restart_had trigger 461, 462

UpDownState attribute 687

UseFence attribute 695

user credentials, viewing 111

user privileges

about 72

assigning from command line 208

changing from Java Console 119

Cluster Administrator 72

Cluster Guest 73

Cluster Operator 73

for specific commands 624

Group Administrator 73

Group Operator 73

in global clusters 75

removing from command line 208, 209

UserInt attribute 687

UserIntGlobal attribute 679

UserIntLocal attribute 679

UserNames attribute 696

users

adding from Java Console 117

deleting from command line 209

deleting from Java Console 118

displaying from command line 209

UserStrGlobal attribute 679

UserStrLocal attribute 679

utilities

hacf 198

hanotify 434

vxlicinst 185

V

VCS

accessibility 713

additional considerations for stopping 194

assistive technology support 715

event triggers 446

logging 586

logging off of 196, 197

logging on to 196, 197

notification 430

querying from command line 210

SNMP and SMTP 430

starting from command line 191

starting on single node 191

stopping from command line 192

stopping with other options 193

stopping without -force 193

troubleshooting resources 597

troubleshooting service groups 595

VCS agent statistics 579

VCS attributes 57

VCS Simulator

administering from Java Console 267

bringing systems online 271

clearing cluster faults from Java Console 273

creating power outages 272

description of 244

faulting resources 272

installing 266

saving offline configurations 272

simulating cluster faults from command
line 277

simulating cluster faults from Java
Console 269

starting from command line 267

VCSFeatures attribute

for clusters 696

for systems 687

VCSMode attribute 696

vector attribute dimension 58

version information, retrieving 233

violation event trigger 462

virtual fire drill

about 263

running 111

VXFEN

tunable parameters 582

vxfen. See fencing module

vxfenadm command 336

- vx fence clear pre command 340, 604
- vx fence sth dw utility 328
- vx keyless command 186
- vx lic inst utility 185

W

- wac 468
- WACPort attribute 696
- wide-area connector 468
- wide-area failover 49
- Wide-Area Heartbeat agent 469
- wizards
 - Notifier Resource Configuration 170
 - Remote Cluster Configuration 509

Z

- zones
 - creating a root on local disk 705
 - creating a root on shared disk 706
 - prerequisites 703
 - resource attributes 701
 - using custom agents 703

