VERITAS Cluster Server™ 1.3.0

Release Notes

Solaris

October 2000
30-000035-399

VERITAS

Disclaimer

The information contained in this publication is subject to change without notice. VERITAS Software Corporation makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. VERITAS Software Corporation shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this manual.

Copyright

Printed in the USA, October 2000.

VERITAS Software Corporation
1600 Plymouth St.
Mountain View, CA 94043
Phone 650–335–8000
Fax 650–335–8050
www.veritas.com

Contents

VCS 1.3.0 Release Notes

This document provides important information regarding VERITAS Cluster Server™ (VCS) version 1.3.0 for Solaris 2.5.1, 2.6, 7 and 8. Review this entire document before installing VCS.

Package Contents

If any of the following items are missing from the original package, contact your sales representative for a replacement.

VERITAS CD

- `VRTSllt`, Low Latency Transport
- `VRTSgab`, Group Membership and Atomic Broadcast
- `VRTSvcs`, Cluster Server
- `VRTSperl`, Perl for `VRTSvcs`
- `VRTScscm`, Cluster Server graphical user interface
- `ntgui`, Windows NT version of `VRTScscm`
- `docs`, PostScript (.ps) and Portable Document Format (.pdf) versions of product documentation,
  - `VRTSvcs_ig.pdf` and `.ps`, *Installation Guide*
  - `VRTSvcs_ug.pdf` and `.ps`, *User's Guide*
  - `VRTSvcs_agentdev.pdf` and `.ps`, *Agent Developer's Guide*
  - `VRTSvcs_bundled.pdf` and `.ps`, *Bundled Agents Reference Guide*
  - `VRTSvcs_notes.pdf` and `.ps`, *Release Notes*
  - `VRTSvcs_a5000.pdf` and `.ps`, *Application Note for Sun StorEDGE A5000 Disk Array*

Hard-Copy Documentation Set

- *Installation Guide*
- *User's Guide*
- *Agent Developer's Guide*
- *Bundled Agents Reference Guide*
- *Release Notes*
- *Application Note for Sun StorEDGE A5000 Disk Array*

Technical Support

For assistance or information regarding VERITAS service packages, contact Technical Support at 800.342.0652 (U.S. and Canada). You may also contact Technical Support via email at support@veritas.com.

For Customers Outside U.S. and Canada

From Europe, the Middle East, or Asia, visit the Technical Support website at http://support.veritas.com for a list of each country's contact information.

Supported Hardware

Network Interfaces (Controllers)

- SBus Quad FastEthernet adapter (for private network, SBus architectures)
- PCI Bus Quad FastEthernet adapter (for private network, PCI Bus architectures)
- HME built-in or SunSwift (PCI/SBus)
- Sun Gigabit Ethernet 2.0 adapter

Note You must use external disk controllers for shared disk subsystems, and network independent hubs for each heartbeat link.

Disk Subsystems

- SBus DWIS/S Wide-SCSI host adapter for simple disk arrays
- Sun A1000 and D1000
- Sun StorEDGE A5000 storage subsystem. For information on how to configure the StorEDGE disk array, see the *VCS Application Note for Sun StorEDGE A5000 Disk Array*, included with the VCS documentation set.
- CLARiiON SCSI disk array, series 1000 (dual-bus configurations only). For information on how to configure the CLARiiON disk array for dual-bus operation, see page 27.
- Hitachi Data Systems 5800 fibre storage array
- MTI 3500 (SCSI)

Servers

- Sun SPARC SBus architectures
- Limited set of Sun PCI Bus systems, specifically Ultra30, Ultra60, Ultra80, E250, and E450, with Sun-supported PCI cards.
- VCS clusters are supported on low-end Sun PCI Bus clusters in configurations of Enterprise 5 or Enterprise 10 servers only.
- Sun Netra t 1120 and 1125 servers

For information on additional supported hardware, contact your VERITAS sales representative.

Note VCS cannot access or control data cached by special-purpose hardware, such as Prestoserve NVRAM. If a VCS server fails and data is being cached by other hardware, the data can be lost.

Unsupported Hardware

CLARiiON SCSI disk array SP split-bus configurations are no longer supported.

Supported Software

- Solaris 2.5.1, 2.6, 7, and 8 operating systems (32-bit and 64-bit).
- VERITAS Volume Manager™ (VxVm) 3.0.4 and 3.1.
- VERITAS File System™ (VxFS) 3.3.x.
- Apply Sun-recommended patches to Solaris 2.5.1 to ensure Y2K compliance.

We also recommend applying the following Solaris kernel and networking patches:

- Solaris 7: 106541-11
- Solaris 2.6: 105795-05, 105181-21, 105541-05
- Solaris 2.5.1: 104212-13, 103640-32, 105532-04

Requirements for the VCS Graphical User Interface

The minimum requirements on Solaris are Ultra 5 or greater, 128MB RAM, and 1280x1024 display resolution. The color depth of the monitor must be at least 8-bit, though 24-bit is recommended.

Required Patches for Cluster Manager and Configuration Editor

The following patches are required to run the VCS graphical user interface "Cluster Manager," and the automated offline configuration tool "Configuration Editor." Patches are available from http://java.sun.com/products/jdk/1.2/install-solaris-patches.html.

- Solaris 2.6: 105568-02 or later
- Solaris 2.5.1: 103566-08 or later, and 103640-08 or later

New Features

Installation Script

VCS now includes an interactive installation script, "InstallVCS." Using information you supply, it installs VCS on each system in the cluster and sets up the VCS communication services, LLT and GAB. You can use the script to remotely install VCS to all systems in the cluster. VCS also includes an uninstallation script to assist you in removing VCS from the cluster. For details, see the *VERITAS Cluster Server Installation Guide*.

Scheduling Class and Configuration Support

You can now configure the scheduling class and priority of the VCS engine, the processes created by the engine, VCS agents, and the script processes created by the agent. For more information see the chapter on advanced topics *VERITAS Cluster Server User's Guide* or the chapter on setting parameters in the *VERITAS Cluster Server Agent Developer's Guide*.

Error Messages

VCS error messages are now formatted according to the following convention:

```
TAG | Date | Time | Product | Error Code | Message
```

For example, a typical log resembles:

```
TAG_B 2000/12/03 18:54:58 VCS:11022:VCS engine (had) started
```

A complete list of VCS error messages is now available as a technical note on the VERITAS Technical Support website. To access the site, enter the words “error messages” in the keyword search field under Cluster Server for UNIX.

The list of VCS error messages is also available in file /opt/VRTSvcs/src/messages. (Note that the list included in /opt directory does not contain LLT or GAB messages.)

Revised Naming Convention for Log Files

Change in Behavior: The naming convention for engine log files and agent log files has been standardized. The format for engine log files is now `engine_A.log`; for agent log files, *agent*`_A.log`.

Event Notification Triggers

VCS now provides a method for notifying the administrator of important events such as a resource or system fault. This method is known as event notification, or *event triggers*. This feature also enables the administrator to take specific actions in response to particular events. For more information, see the chapter on advanced topics in the *VERITAS Cluster Server User’s Guide*.

Change in Behavior: The location of the PreOnline trigger has changed. See the chapter on advanced topics in the *VERITAS Cluster Server User’s Guide* for details.

Detecting Concurrency Violations

When a concurrency violation occurs, the Violation event trigger takes the failover service group offline on the system that caused the violation. For details, see the Introduction in the *VERITAS Cluster Server User’s Guide.*

AutoRestart

Setting the AutoRestart attribute enables a service group to be brought back online without manual intervention when a persistent resource goes online after fault and service group is no longer faulted. For more information on the AutoRestart attribute, see the chapter on VCS attributes in the *VERITAS Cluster Server User’s Guide*.

AutoStartList

Change in Behavior: Beginning with 1.3.0, you must define the service group attribute SystemList prior to setting the attribute AutoStartList. For more information, see the chapter on VCS attributes in the *VERITAS Cluster Server User's Guide*.

Type Dependencies

VCS provides a method for establishing dependencies between resources according to the resource type. Resource-type dependencies are specified by the user on a per-group basis in the main.cf file. Rather than specifying the resource dependencies one-by-one, you can specify that all resources of a particular type depend on all resources of another type within the service group. For more information, see Chapter 1 in the *VERITAS Cluster Server User's Guide*.

Service Group Dependencies

You may now specify "soft" or "firm" group dependencies regardless of local, global, or remote restrictions. For details, see the chapter on service group dependencies in the *VERITAS Cluster Server User's Guide*.

Change in Behavior: Previously, local dependencies were firm, global and remote were soft. By default, all are now firm. For instructions on configuring soft dependencies, see the chapter on service group dependencies in the *VERITAS Cluster Server User's Guide*.

AutoDisabling

Change in Behavior: When VCS does not know the status of a service group on a particular system, it now autodisables the service group on that system. Autodisabling occurs under the following conditions:

- When GAB is running on the system but the VCS engine is not.
- When all resources within the service group are not probed on the system.
- When a particular system is visible through disk heartbeat only.

For more information, see the chapter on troubleshooting in the *VERITAS Cluster Server User's Guide* and the manual page for gabconfig.

Additional Considerations for Stopping VCS

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. For details, see the chapter on administering VCS from the command line in the *VERITAS Cluster Server User's Guide*.

New Attributes

The following attributes have been added for enhanced for this release of VCS. For details, see the chapter on attributes in the *VERITAS Cluster Server User's Guide*.

- OfflineMonitorInterval
- GUIIPAddr
- ResourceOwner
- SysInfo

New Bundled Agents

The following agents are included with this release of VCS. For details, refer to the *VCS Bundled Agents Reference Guide*.

- Application
- DiskReservation

New Script for Updating Custom Agents

You may now update custom agents using a script provided in $VCS_HOME/bin/vcs_upg.pl. For details, see the accompanying ReadMe file $VCS_HOME/bin/vcs_upg.readme.

New GUI Features

Several new features have been introduced into the VCS graphical user interface, including modifying user preferences including audio notifications, importing resource type information, User Manager, and a template-based configuration wizard. For details, see the chapter on Cluster Manager in the *VERITAS Cluster Server User's Guide*.

Revised Clean Entry Point

When implemented, the `clean` entry point is now called automatically when a resource faults. The `offline` entry point is no longer used for fault cleanup. If a resources faults unexpectedly and `clean` is not implemented, the fault may not be cleaned up properly.

Note The preceding information applies *only* to custom agents. For details, see the Appendix in the *VERITAS Cluster Server Agent Developer's Guide*.

Increased Default Value for FaultOnMonitorTimeouts Attribute

The default value of the FaultOnMonitorTimeouts static attribute (agent parameter) has been increased from 2 to 4. This helps reduce the risk of transient "monitor hangs," which cause unwanted group failovers.

Command Option for gabconfig Changed

The option `-r` for the command `gabconfig` has been replaced by other options, as described in the GAB manual pages and in the chapter on troubleshooting in the *VERITAS Cluster Server User's Guide*.

Manual Pages

The manual pages for the `VRTSllt`, `VRTSgab`, and `VRTSvcs` are installed in `/opt/VRTS/man`. Set the `MANPATH` environment variable so the `man`(1) command can point to the VCS manual pages.

For Bourne or Korn shell (`sh` or `ksh`), type:

```
# MANPATH=$MANPATH:/opt/VRTS/man
# export MANPATH
```

For C shell (`csh` or `tcsh`), type:

```
# setenv MANPATH ${MANPATH}:/opt/VRTS/man
```

For more information, refer to the `man`(1) manual page.

Bundled Agents

The following agents are included with VCS. For information on any of the agents listed below, refer to the *VCS Bundled Agents Reference Guide*.

- Application
- Disk
- DiskGroup
- DiskReservation
- ElifNone
- FileNone
- FileOnOff
- FileOnOnly
- IP
- IPMultiNIC
- Mount
- MultiNICA
- NFS
- NIC
- Phantom
- Process
- Proxy
- ServiceGroupHB
- Share
- Volume

Enterprise Agents

The following agents (enterprise and storage) are sold separately. Contact your VERITAS sales representative for details on these agents, additional agents under development, and agents available as part of an Edition or through VERITAS Consulting Services.

- FireWall-1
- Informix
- Oracle
- SIMS
- PC NetLink
- Sybase
- SuiteSpot
- NetBackup

Storage Agents

- NetApp

Note To configure any of the above enterprise or storage agents with VCS, you must first verify that you are running the correct version of the agent. Only VCS agents version 1.3.0 are compatible with VCS 1.3.0.

Using Agents in NIS

Programs using networked services (for example, NIS, NFS, RPC or a TCP socket connection to a remote host) can hang if the host is disconnected from the network. Note that if such a program is used as an agent entry point, a network disconnect can cause the entry point to hang and possibly timeout. For example, if the host is configured to use NIS maps as a client, basic commands such as `ps -ef` can hang if there is network disconnect.

Compiling Custom Agents

Custom agents must be developed using SUNWspro 4.0, 4.2, or 5.0 C++ compilers. (Note that you must specify the `-compat` flag when compiling and linking with the 5.0 compiler.) Other C++ compilers are not supported.

Note Existing custom agents *must* be recompiled for use with VCS 1.3.0.

Upgrading Custom Agents

See the *VERITAS Cluster Server Agent's Developer's Guide* for instructions on how to upgrade custom agents built prior to this release.

Note Before using custom agents with VCS 1.3.0, you must first upgrade the agents as instructed in the *Agent Developer's Guide.*

Migrating from FirstWatch or VERITAS-HA to VCS

Migrating from supported versions of FirstWatch or VERITAS-HA (formerly AXXiON-HA) requires Consulting Services. VERITAS consultants have the tools and documentation to expedite migration from either of these products. Contact your sales representative for more information.

Installing Binary Packages

Install the packages from the CD onto Sun systems running Solaris 2.5.1, 2.6, 7, or 8. Commands for verifying LLT, GAB, and cluster operation are located in the `/sbin` or `/usr/sbin` directories. Add these directories to your `PATH` environment variable.

For Bourne or Korn shell (`sh` or `ksh`), type:

```
# PATH=/sbin:/usr/sbin:$PATH export PATH
```

For C shell (`csh` or `tcsh`), type:

```
% setenv PATH /sbin:/usr/sbin:${PATH}
```

For more information, refer to the *VERITAS Cluster Server Installation Guide*.

Upgrading to VCS 1.3.0 from Previous Versions

Perform the following procedures in the order they are listed to upgrade to VCS 1.3.0 from previous versions. (To upgrade to Solaris 8 while upgrading to VCS 1.3.0, see "Upgrading to Solaris 8 from Solaris 7" on page 20.)

Note To ensure a successful upgrade, verify that all systems in the cluster are running VCS.

- ✔ Shut down VCS.
- ✔ Remove previous version of VCS.
- ✔ Install VCS 1.3.0.
- ✔ Start VCS 1.3.0.
- ✔ Upgrade Cluster Manager.

▼ To shut down VCS

1. Make the VCS configuration writable. On any system, type:

   ```
   # haconf -makerw
   ```

2. List the groups in your configuration. On any system, type:

   ```
   # hagrp -list
   ```

3. Freeze all service groups. On any system, type the following command for each group name displayed in the output from the step 2.

   ```
   # hagrp -freeze group_name -persistent
   ```

4. Save the configuration file (`main.cf`) with the groups frozen. On any system, type:

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

5. Shut down VCS. On any system, type:

   ```
   # hastop -all -force
   ```


6. Confirm that VCS has shut down. On each system, type:

```
# gabconfig -a
```

Output resembles:

```
GAB Port Memberships
===================================
Port a gen 23dc0001 membership 01
```

Note that the output shows no membership for port h.

7. Confirm that GAB is not running on any disks:

```
# gabdiskhb -l
# gabdiskx -l
```

If it is, remove it from the disks:

```
# gabdiskhb -d
# gabdiskx -d
```

8. Shut down GAB. On each system, type:

```
# gabconfig -U
```

9. Confirm that GAB has shut down. On each system, type:

```
# gabconfig -a
```

Output resembles:

```
GAB Port Memberships
===================================
```

Note that the output shows no membership for port a.

10. Unload GAB. On each system, perform the following steps:

    a. Identify the GAB kernel module:

```
# modinfo | grep gab
149 50cc6000  2b451 112   1  gab (GAB device)
```

The module number appears at the far left of the output (in bold, above).

b. Unload GAB using the module number:

```
# modunload -i 149
```

The following message is displayed on the console:

```
GAB unavailable
```

11. Shut down LLT. On each system, type:

```
# lltconfig -U
```

The following message is displayed on the console:

```
lltconfig: this will attempt to stop and reset LLT.
Confirm (y/n)?
```

Note This message does *not* display if you are upgrading from version 1.0.2.

12. Type `y` on each system in response to the message.

13. Unload LLT. On each system, perform the following steps:

a. Identify the LLT kernel module:

```
# modinfo | grep llt
147 50ca4000 d6bc 110 1 llt (Low Latency Transport device)
```

The module number appears at the far left of the output (in bold, above).

b. Unload LLT using the module number:

```
# modunload -i 147
```

The following message is displayed on the console:

```
LLT Protocol unavailable
```


14. Compare your `types.cf` file with the original `types.cf` file. Note that if any changes have been made to the default `types.cf` file, or any user-defined types have been added to the original `types.cf` file, the files will be different.

    On any system, type:

    ```
    # diff -w /etc/VRTSvcs/conf/config/types.cf
          /etc/VRTSvcs/conf/types.cf
    ```

    - If the files are the same, proceed to the next procedure, "To remove previous version of VCS" on page 17.
    - If the files are different, make a copy of the modified file. You will use the copy in step 9 of "To install VCS 1.3.0" on page 17. For example:

      ```
      # cp /etc/VRTSvcs/conf/config/types.cf
             /etc/VRTSvcs/conf/types.save
      ```

    The following example output from the `diff` command shows the minor differences created when VCS dumped the configuration. These differences can be safely ignored during the upgrade procedure.

    ```
    60c60
    <       int IfconfigTwice
    ---
    >       int IfconfigTwice = 0
    70c70
    <       int IfconfigTwice
    ---
    >       int IfconfigTwice = 0
    94c94
    <       int IfconfigTwice
    ---
    >       int IfconfigTwice = 0
    145c145
    <       int OnlineNFSRestart
    ---
    >       int OnlineNFSRestart = 0
    ```

▼ To remove previous version of VCS

1. On each system, type:

```
# pkgrm VRTScsga VRTSvcs VRTSgab VRTSllt VRTSperl VRTSvcswz
# pkgrm VRTScsga VRTSvcs VRTSgab VRTSllt VRTSperl VRTSvcswz
```

Note If you are upgrading from 1.0.1 or 1.0.2, you must also remove the package `VRTSsnmp`, and any packages containing a `.2` extension, such as `VRTScsga.2`, `VRTSvcs.2`, `VRTSgab.2`, `VRTSllt.2`, `VRTSperl.2`, `VRTSsnmp.2`, and `VRTSvcswz.2`.

2. As the packages are removed, answer `Yes` when prompted.

   If you have installed VCS enterprise agents, you must also remove the agent packages. Package names for enterprise agents include `VRTSvcsix` (Informix), `VRTSvcsnb` (NetBackup), `VRTSvcssor` (Oracle), and `VRTSvcssy` (Sybase). Refer to the *VCS Enterprise Agents Release Notes* for more information on removing agents.

▼ To install VCS 1.3.0

Note The following steps must be performed on each system in the cluster.

1. Verify that you have implemented the `clean` entry point in custom resources for which a fault may occur. (If necessary, review "Revised Clean Entry Point" on page 8.)

2. Insert the VERITAS Cluster Server CD.

3. If you are not running Solaris volume-management software, you must mount the CD manually. For example:

```
# mount -F hsfs -o ro /dev/dsk/c0t6d0s2 /cdrom
# cd cdrom
```

4. Install the software:

```
# pkgadd -d .
```

   Available packages are displayed.

5. When prompted, select `all`.

6. As the packages are added, answer `Yes` when prompted.

7. After all packages are added, type `q`.

8. **If you are upgrading on Solaris 2.5.1**: reboot the system when prompted. Otherwise, proceed to the next step.

9. Copy the saved `types.cf` file (`types.save`) created in step 14 of "To shut down VCS" to `/etc/VRTSvcs/conf/config`. On any system, type:

```
# cd /etc/VRTSvcs/conf/config
# cp ../types.save types.cf
```

About the VCS Log Files

The naming convention for VCS log files has changed from *logname*.log_TAG to *logname*_TAG.log. For example, the format for the engine log is now `engine_A.log`; for agent log files, *agent*_`A.log`.

Location and Syntax of the PreOnline Trigger

In VCS 1.1.2, PreOnline scripts were specified in one of two locations:

- /opt/VRTSvcs/bin/*service_group*/PreOnline
- /opt/VRTSvcs/bin/PreOnline

Starting with 1.3.0, VCS provides an entire suite of triggers that are called upon during an event. Types of triggers include PreOnline, PostOnline, PostOffline, ResFault, SysOffline, InJeopardy, and more. (For a complete list of triggers and their descriptions, see the chapter on advanced topics in the *VERITAS Cluster Server User's Guide*.)

Trigger scripts are stored in /opt/VRTSvcs/bin/triggers directory as /opt/VRTSvcs/bin/triggers/preonline, /opt/VRTSvcs/bin/triggers/ResFault, etc. A sample directory is also provided, /opt/VRTSvcs/bin/triggers/sample_triggers, which includes an example for each of trigger.

If Upgrading from 1.1.2

If you had one PreOnline script per service group configured in VCS 1.1.2, for example:

/opt/VRTSvcs/bin/group_one/preonline

/opt/VRTSvcs/bin/group_two/preonline

/opt/VRTSvcs/bin/group_three/preonline

you now must only maintain one trigger: /opt/VRTSvcs/bin/triggers/preonline.

This trigger is passed three arguments: *system_name group_name* and *whypreonlining*. Based on the *group_name*, you can determine the codepath the trigger will take. For more information, see the sample preonline trigger in the directory /opt/VRTSvcs/bin/sample_triggers/preonline.

▼ To start VCS 1.3.0

1. On each system, type:

```
# /etc/rc2.d/S70llt start
# /etc/rc2.d/S92gab start
# /etc/rc3.d/S99vcs start
```

2. Verify that all the resources have been probed. On any system, type:

```
# hastatus -summary
```

3. Unfreeze all service groups. On any system, type:

```
# haconf -makerw
# hagrp -unfreeze service_group -persistent
# haconf -dump -makero
```

4. Repeat the `hagrp` command for each group displayed in the output of step 2 on page 13 (“To shut down VCS”).

▼ To upgrade the VCS GUI

Note Prior to release 1.3.0, the binary package name for Cluster Manager was `VRTScsga`.

On UNIX (Solaris)

1. Type the following commands:

```
# pkgrm VRTScsga
# pkgadd VRTScscm
```

2. Answer Yes if prompted.

On Windows NT

1. From the Control Panel, double-click Add/Remove Programs.

2. Select VERITAS Cluster Manager.

3. Click Add/Remove.

4. Follow the instructions presented by the uninstall wizard.

For fresh installations, see the *VERITAS Cluster Server Installation Guide* for instructions on how to install Cluster Manager.

Upgrading to Solaris 8 from Solaris 7

1. Make the VCS configuration writable. On the maintenanced server, type:

   # **haconf -makerw**

2. Move all service groups from the maintenanced server to the other system, and keep services from failing over to this server. On the maintenanced server, type:

   # **hasys -freeze -persistent -evacuate** *maintenanced_server*

3. Check if all service groups and resources are OFFLINE on the maintenanced server and ONLINE on the other system. Type:

   # **hastatus**

4. Unload the VCS services on the maintenanced server. On the maintenanced server, type:

   # **hastop -local**

5. Follow the Sun installation guide to upgrade the OS kernel on the maintenanced server to the new version of Solaris. After the system comes up, verify that it provides the same services as before the upgrade.

6. Verify that the VCS services are running on the upgraded server. On the upgraded server, type:

   ```
   # ps -ef | grep ha
   root   576   1  0 16:54:12 ?    0:02 /opt/VRTSvcs/bin/had
   root   578   1  0 16:54:13 ?    0:00 /opt/VRTSvcs/bin/hashadow
   ```

   If they are not running, reload the VCS services. Type:

   ```
   # hastart
   ```

7. Unfreeze the upgraded server and save the configuration. On the upgraded server, type:

   ```
   # hasys -unfreeze -persistent upgraded_server
   # haconf -dump -makero
   ```

8. To upgrade VCS, follow the procedure described in "Upgrading to VCS 1.3.0 from Previous Versions" on page 13.

9. Restore all services to the normal state prior to the upgrade by bringing the service groups online on the upgraded server (if applicable).

Additional Considerations for Upgrading the Sun OS

- The OS upgrade may take hours to finish. We recommend coordinating with your system administrator to plan the outage time of the other system. This helps reduce downtime and ensures availability of services for your customers.
- To upgrade the OS on another system in the cluster, repeat step 1 on page 20 through step 7 and step 9 above.

Software Limitations in VCS 1.3.0

Issuing hastop -local for Groups with OnOnly or Persistent Resources

For service groups made up solely of OnOnly or persistent resources, issuing the command `hastop -local` on the system causes HAd to transition to the LEAVING state. HAd cannot exit the state unless you issue the command `hastop -local -force`.

Workaround: Add a Phantom resource to service group to ensure that the command `hastop -local` shuts down the engine properly. For more information on the Phantom resource, see the *VERITAS Cluster Server Installation Guide*.

Engine May Hang in LEAVING State

When the command `hares -online` is issued for a parent resource when a child resource faults, and is followed by the command `hastop -local` on the same node, the engine transitions to the LEAVING state and hangs. **Workaround:** Issue the command `hastop -local -force`.

GUI Hang

The VCS graphical user interface hangs when it attempts to connect to a system with attribute GUIIPaddr=localhost. Specifically, If this attribute is set to “localhost” or 127.0.0.1, the engine closes the connection.

Also, double-clicking a Cluster Monitor panel may also cause the logon dialog box to hang. **Workaround:** Single- or right-click the panel.

Agent Hang

Under extreme conditions, the volume agent may hang. We observed this behavior under the following circumstances:

- Failover was set to 0 for the JNI fibre-channel driver (JNIfcaPCI). Note this is *not* failover for VCS. There is a variable for the JNI driver called “failover,” which defines the number of seconds after the target is declared offline, and before it is declared failed, and all pending commands are flushed back to the application. This value is set in the file `/kernel/drv/fca-pci.conf`. Setting failover to 0 means that the target is never declared failed.
- Fibre cable was disconnected from the switch (to simulate failure of the fibre drives).

Note When failover for JNI driver was set to a value of 30 seconds, VCS agent behavior was normal.

Erroneous Message in Engine Log File

The following message is seen in the engine log vxfs mount:

```
/dev/vx/dsk/sharedg/vol03 is not a vxfs file system
```

VCS tries to mount the file system. If it returns an error, VCS runs `fsck` on the file system and then on mount. The above message is seen for a vxfs file system when VCS tries to mount the file system for the first time and it cannot because fsck needs to be run.

Problem with LinkMonitoring

If you enable LinkMonitoring and issue any form of the `hastop` command, the HAD process crashes. A crash of VCS is not critical because HAD crashes after offlining or switching service groups. However, because the HAD process quits ungracefully, the hashadow process restarts HAD and VCS remains running. This is timing-dependent and occurs sporadically. **Workaround:** There are two solutions to this problem:

- Do not use LinkMonitoring. This is default behavior. Use the InJeopardy trigger or SNMP trap to get notification when only one network link is remaining.
- If you must use LinkMonitoring, set it to 0 before issuing the command `hastop`. Issue the command `haclus -disable LinkMonitoring` to set the attribute to 0.

Undocumented Commands and Command Options

VCS contains undocumented commands and command options intended for development use only. The use of any undocumented commands is not supported by VERITAS Customer Support.

NFS Locking

NFS lock failover is not yet supported.

Cluster Manager

Solaris

We recommend not running the VCS graphical user interface “Cluster Manager” for an extended period on a cluster server system. The Solaris version of the Java Virtual Machine has a memory leak that can gradually consume the host system’s swap space. This leak does not appear on Windows NT systems.

Windows NT

Cluster Manager on Windows NT does not display properly on an NCD XTerminal due to different versions of X11 libraries.

Custom Agent Libraries

Releases prior to VCS 1.1 included the libraries `libvcsprimitives.a` and `libvcscommon.a`. Agent developers had to link with the agent framework library (`libvcsagfw.so` or `libvcsagfw.a`) and `libvcsprimitives.a`. (The library `libvcscommon.a` was not required.)

In the current release, the *only* VCS library required for custom agents is the agent framework library. The `VRTSvcs` package does *not* include `libvcsprimitives.a` and `libvcscommon.a`. Agent developers using C++ must change their makefiles to delete any references to these two libraries.

Symbolic Link for LLT in the /dev Directory

If the symbolic link for LLT in the `/dev` directory is accidently removed after the first installation of the `VRTSllt` package, you must re-create it using the following commands:

```
# cd /dev
# ln -s ..devices/pseudo/clone@0:llt
```

Volume and DiskGroup Agents

The VCS Volume and DiskGroup agents do not work properly with VERITAS Volume Manager™ versions 2.5.1, 2.6, and 3.0.1. See the VxVM documentation for more information.

GlobalCounter Attribute not Updated by GUI

To avoid updating various views unnecessarily, the VCS graphical user interface does not update the GlobalCounter attribute of the cluster.

Gabsync

The `gabsync`(1M) utility is intended for testing only. Do not use this utility in a production environment. `Gabsync` enables a sync state in the GAB driver that permits user-level processes monitored by the kernel to close the GAB device without generating user-process death notification or halting the system. Under these conditions, a process failure could be interpreted as a system failure. Also, the normal operating system shutdown could kill the process before the system is fully shut down, causing other systems to take over shared resources before they are free.

The start/stop Script in /etc/*.d Not Linked

The /etc/init.d/vcs script is *not* linked to the corresponding entries in /etc/rc3.d/S99vcs and /etc/rc0.d/K10vcs.

Command hasys -nodeid

The command hasys -nodeid does not work if nodeid is set to 0.

Group Dependency Limitations

Online Local Firm Dependency Violation

If the parent group and the child group are online on node 1, and if the child group faults, VCS begins to take the parent group offline. However, this occurs at the same time the child group is failing over to node 2. If the parent group fails to go completely offline and the child group goes online on node 2, thus resulting in a dependency violation.

Online Remote Firm Dependency Violation

If the parent group is online on node 1 and the child group is online on node 2 and faults, the child group selects node 1 as its failover target, but the parent group fails to go offline on node 1, thus resulting in a dependency violation.

Incorrect Migration by Parent Group

In failover parent-parallel child online local soft dependency, parent incorrectly migrates to the node where the child group is brought online.

No Failover for Some Instances of Parent Group

In service groups in which the group dependency is configured as parallel parent/failover child, online global, remote soft or firm, the parent group may not online on all nodes after a child group faults.

Parent Groups May Not be Taken Offline in Firm Dependencies

For firm dependencies (online local, online global, or online remote), when the attribute AutoFailOver is set to 0 for the child group and the child group faults, the parent groups may not be taken offline.

Engine Doesn't Check for Group Dependency Violation for hares -online

When the command hares -online is issued, the engine doesn't check for group dependency violation. For example, if group G1 requires group G2 online local firm, and group G2 is online on node 1, if `hares -online` is issued for a resource of G1 on node 2, the engine incorrectly allows resource online operation, thus violating group dependency.

Software Fixes and Enhancements

VCS 1.3.0	Incident Number	Description
	30153	Executing the command `haconf -dump` on a system whose /tmp directory is full caused the engine process to crash on that system.
	31917	Quick unregister and re-register of GAB no longer causes system halt.
	32238	Mount agent clean script now defines `$df`.
	33635	Persistent non-critical resources are treated like other non-critical resources (for example, they do not necessarily fault the service group at resource fault).
	37557	VCS Sybase agent started the database but did not report it as online if a very long path is used.
	37734	Clean script for the VCS enterprise agent for Oracle now functions properly. Previously, the "if" block for string comparison used a "!=" instead of "ne". As a result, the `if` block was not executed in any condition. Now if the result of the `if` block execution is false, the oracle instance is shutdown.
	38693	VCS now sends an snmp trap when a heartbeat is disconnected.
	43105	The command to take a child group offline was not rejected if parent group was online.
	45301	Deleting a Type attribute caused the corresponding agent to hang.
	45310	Agents no longer create and remove a temp file for each script execution.
	45321	Agents monitor offline resources less frequently.

VCS 1.3.0	Incident Number	Description
	45548	Stale data in lock file no longer prevents VCS from terminating a hung agent.
	46071	Added logic to handle the interval during which a group is taken offline at the same time a resource that was not waiting to go online times out (for example, after faulting and restarting).
	46177	User could not modify any Assoc type attribute from Attribute View panel.
	46492	Occasionally, the command `halog -a` caused halog to core dump.
	46544	The event trigger ResNotOff was fired inconsistently.
	47663	Agent scripts no longer get SIGPIPE when writing to stdout or stderr.

VCS 1.1.2	Incident Number	Description
	34778	Problem with false concurrency violations resolved with GAB package.

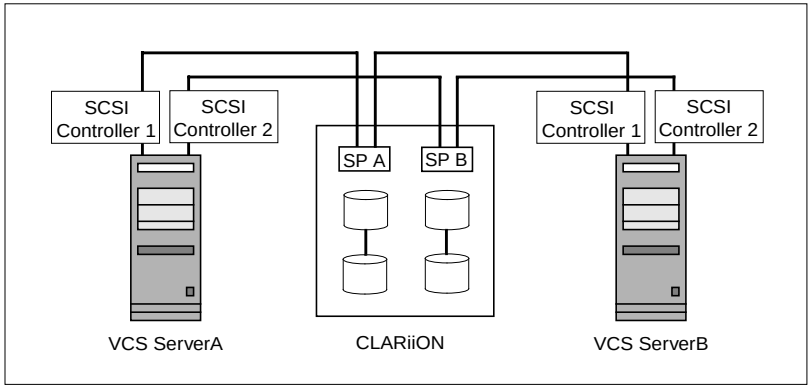
VCS 1.1.1	Incident Number	Description
	28067	Group offline followed by online now faults resources and fails over.
	31881	Underscore (_) is now allowed in system name.
	32914	Quick-Start wizard referred to `rsh` (restricted shell) instead of `remsh` (remote shell).
	33184	Added instrumentation to LLT to enhance heartbeat packet information. Also, increased peerinact timer default to 16 seconds.
	33762	Agent framework now calls clean script when offline is returned and subsequent monitor times out.

Configuring a CLARiiON Dual-Bus Configuration

VCS supports a dual-bus CLARiiON configuration with ATF. The ATF driver provides transparent failover of the I/O path, thus ensuring that data-dependent applications are uninterrupted if a failure occurs on the path. Refer to the CLARiiON documentation for instructions on how to configure the ATF driver.

The ATF driver also monitors the I/O paths to the CLARiiON disk array, which fails over LUNs to the functional SP. Thus, the server does not require failover for client systems to gain data availability.

In the figure below, each VCS server is connected to two CLARiiON SPs. Both servers have access to Logical Unit Numbers (LUNs) bound through the SPs. Note that if you restore a failed SP, you may be required to execute `/usr/sbin/atf_restore` several times before the restore succeeds.



Dual-Bus Configuration

Setting Up the Hardware

In the following steps, the VCS servers are referred to as serverA and serverB.

1. Install the four SCSI adapters, two on each server.
2. Connect `SCSI-A` of the CLARiiON to both servers with SCSI cables.
3. Connect `SCSI-B` of the CLARiiON to both servers with SCSI cables.

4. Connect the two serial ports on the CLARiiON to the servers with serial cables. The serial connections configure the SPs with DASSMGR, software for configuring the CLARiiON disk array. Select one of the following methods to set up the serial connections:

   - To configure the CLARiiON from one server, connect one of the VCS servers to ports A and B of the CLARiiON with serial cables. Then, on the same server, type the following command to configure SP A:

     ```
     # dassmgr ttya
     ```

     Type the following command on the same server to configure SP B:

     ```
     # dassmgr ttyb
     ```

   - To configure the CLARiiON from both servers through DASSMGR, connect serverA to serial port A of the CLARiiON, and serverB to serial port B of the CLARiiON. Then, on serverA, type the following command to configure SP A:

     ```
     serverA# dassmgr ttya
     ```

     On serverB, type the following command to configure SP B:

     ```
     serverB# dassmgr ttya
     ```