Veritas Storage Foundation™ for Sybase ASE CE Administrator's Guide

Linux

6.0

November 2011

Veritas Storage Foundation for Sybase ASE CE 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: 6.0

Document version: 6.0.0

Legal Notice

Copyright © 2011 Symantec Corporation. All rights reserved.

Symantec, the Symantec logo, Veritas, Veritas Storage Foundation, CommandCentral, NetBackup, Enterprise Vault, and LiveUpdate 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 support offerings include the following:

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

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

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

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

Contacting Technical Support

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

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

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

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

- Product release level

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

Licensing and registration

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

www.symantec.com/business/support/

Customer service

Customer service information is available at the following URL:

www.symantec.com/business/support/

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

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

Support agreement resources

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

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

Documentation

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

doc_feedback@symantec.com

About Symantec Connect

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

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

Contents

Technical Support 4

Chapter 1 Overview of Veritas Storage Foundation for Sybase ASE CE 11

How SF Sybase CE works (high-level perspective) 11
About SF Sybase CE components 15
Communication infrastructure 15
Cluster interconnect communication channel 17
Low-level communication: port relationship between GAB and processes 19
Cluster Volume Manager (CVM) 20
Cluster File System (CFS) 22
Veritas Cluster Server 25
About I/O fencing in SF Sybase CE environment 28
Sybase ASE CE components 35
About optional features in SF Sybase CE 37
Typical configuration of clusters in secure mode 37
Typical configuration of VOM-managed clusters 38
About SF Sybase CE global cluster setup for disaster recovery 39
How the agent makes Sybase highly available 39
About Veritas Operations Manager 40
Symantec Operations Readiness Tools 40

Chapter 2 Administering SF Sybase CE and its components 43

Administering SF Sybase CE 43
Setting the environment variables 44
Starting or stopping SF Sybase CE on each node 44
Applying operating system updates 48
Adding storage to an SF Sybase CE cluster 49
Recovering from storage failure 50
Enhancing the performance of SF Sybase CE clusters 51
Verifying the nodes in a cluster 51
Administering VCS 52

Viewing available Veritas device drivers 52
Starting and stopping VCS 54
Environment variables to start and stop VCS modules 55
Adding and removing LLT links 57
Configuring aggregated interfaces under LLT 59
Displaying the cluster details and LLT version for LLT links 60
Configuring destination-based load balancing for LLT 61
Enabling and disabling intelligent resource monitoring for agents manually 61
Administering the AMF kernel driver 63
Administering I/O fencing 64
About administering I/O fencing 64
About the vxfentsthdw utility 65
About the vxfenadm utility 73
About the vxfenclearpre utility 78
About the vxfenswap utility 81
Enabling or disabling the preferred fencing policy 91
Administering CVM 93
Establishing CVM cluster membership manually 93
Changing the CVM master manually 94
Importing a shared disk group manually 97
Deporting a shared disk group manually 97
Verifying if CVM is running in an SF Sybase CE cluster 97
Verifying CVM membership state 98
Verifying the state of CVM shared disk groups 98
Verifying the activation mode 99
Administering CFS 99
Adding CFS file systems to VCS configuration 99
Using cfsmount to mount CFS file systems 100
Resizing CFS file systems 100
Verifying the status of CFS file systems 100
Administering the Sybase agent 101
Sybase agent functions 101
Monitoring options for the Sybase agent 104
Using the IPC Cleanup feature for the Sybase agent 104
Configuring the service group for Sybase from Cluster Manager (Java console) 106
Configuring the service group for Sybase using the command line 107
Bringing the Sybase service group online 109
Taking the Sybase service group offline 109
Modifying the Sybase service group configuration 110
Viewing the agent log for Sybase 110

Chapter 3 Troubleshooting SF Sybase CE 111

About troubleshooting SF Sybase CE 111
Gathering information for support analysis 111
Log files 114
About SF Sybase CE kernel and driver messages 116
VCS message logging 116
Troubleshooting tips 122
What to do if you see a licensing reminder 125
Restarting the installer after a failed connection 126
Installer cannot create UUID for the cluster 126
Troubleshooting I/O fencing 126
The vxfentsthdw utility fails when SCSI TEST UNIT READY command fails 126
Node is unable to join cluster while another node is being ejected 127
System panics to prevent potential data corruption 127
Cluster ID on the I/O fencing key of coordinator disk does not match the local cluster's ID 128
Fencing startup reports preexisting split-brain 129
Registered keys are lost on the coordinator disks 131
Replacing defective disks when the cluster is offline 132
Troubleshooting Cluster Volume Manager 134
Restoring communication between host and disks after cable disconnection 134
Shared disk group cannot be imported 134
Error importing shared disk groups 135
Unable to start CVM 135
CVM group is not online after adding a node to the cluster 136
CVMVolDg not online even though CVMCluster is online 136
Shared disks not visible 136
Troubleshooting interconnects 137
Network interfaces change their names after reboot 137
Example entries for mandatory devices 137
Troubleshooting Sybase ASE CE 138
Sybase private networks 138
Sybase instances under VCS control 138
Node does not reboot 138
Sybase instance not starting 138

Chapter 4 Prevention and recovery strategies 139

Prevention and recovery strategies 139
Verification of GAB ports in SF Sybase CE cluster 139

Examining GAB seed membership 140
Manual GAB membership seeding 141
Evaluating VCS I/O fencing ports 142
Verifying normal functioning of VCS I/O fencing 143
Managing SCSI-3 PR keys in SF Sybase CE cluster 143
Identifying a faulty coordinator LUN 145
Starting shared volumes manually 145
Listing all the CVM shared disks 145
I/O Fencing kernel logs 146

Chapter 5 Tunable parameters 147

About SF Sybase CE tunable parameters 147
About GAB tunable parameters 147
About GAB load-time or static tunable parameters 148
About GAB run-time or dynamic tunable parameters 149
About LLT tunable parameters 154
About LLT timer tunable parameters 155
About LLT flow control tunable parameters 159
About VXFEN tunable parameters 162
Configuring the VXFEN module parameters 163

Appendix A Error messages 165

About error messages 165
VxVM error messages 165
VXFEN driver error messages 166
VXFEN driver informational message 166
Node ejection informational messages 167

Index 169

Chapter 1

Overview of Veritas Storage Foundation for Sybase ASE CE

This chapter includes the following topics:

- How SF Sybase CE works (high-level perspective)
- About SF Sybase CE components
- About optional features in SF Sybase CE
- How the agent makes Sybase highly available
- About Veritas Operations Manager
- Symantec Operations Readiness Tools

How SF Sybase CE works (high-level perspective)

Sybase ASE Cluster Edition is a shared disk cluster implementation of Sybase's flagship enterprise database. Sybase ASE is a highly reliable, scalable, and efficient database engine used in mission critical environments such as financial markets, telecommunications networks, and healthcare. Sybase ASE CE allows multiple "instances" of the Sybase ASE database engine running on different hardware "nodes" to simultaneously access and manage a common set of databases on disks. The primary goal of such a system is to provide exceptional availability with the added benefit of some scalability for certain use cases.

In traditional environments, only one instance accesses a database at a specific time. SF Sybase CE enables all nodes to concurrently run Sybase adaptive servers

and execute transactions against the same database. This software coordinates access to the shared data for each node to provide consistency and integrity. Each node adds its processing power to the cluster as a whole and can increase overall throughput or performance.

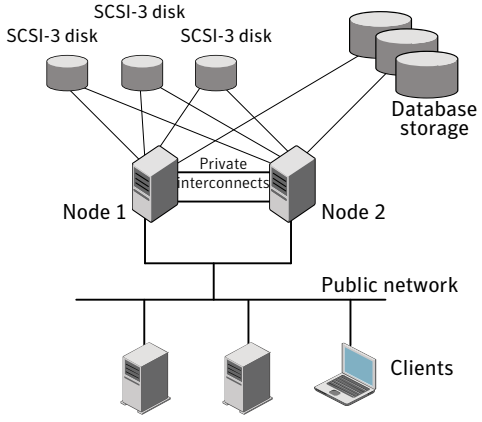
At a conceptual level, SF Sybase CE is a cluster that manages applications (instances), networking, and storage components using resources contained in service groups. SF Sybase CE clusters have the following properties:

- Each node runs its own operating system.
- A cluster interconnect enables cluster communications.
- A public network connects each node to a LAN for client access.
- Shared storage is accessible by each node that needs to run the application.

Figure 1-1 below displays the basic layout and individual components required for a SF Sybase CE installation. This basic layout includes the following components:

- Nodes that form an application cluster and are connected to both the coordinator disks and databases
- Databases for storage and backup
- SCSI-3 Coordinator disks used for I/O fencing

Figure 1-1 SF Sybase CE basic layout and components

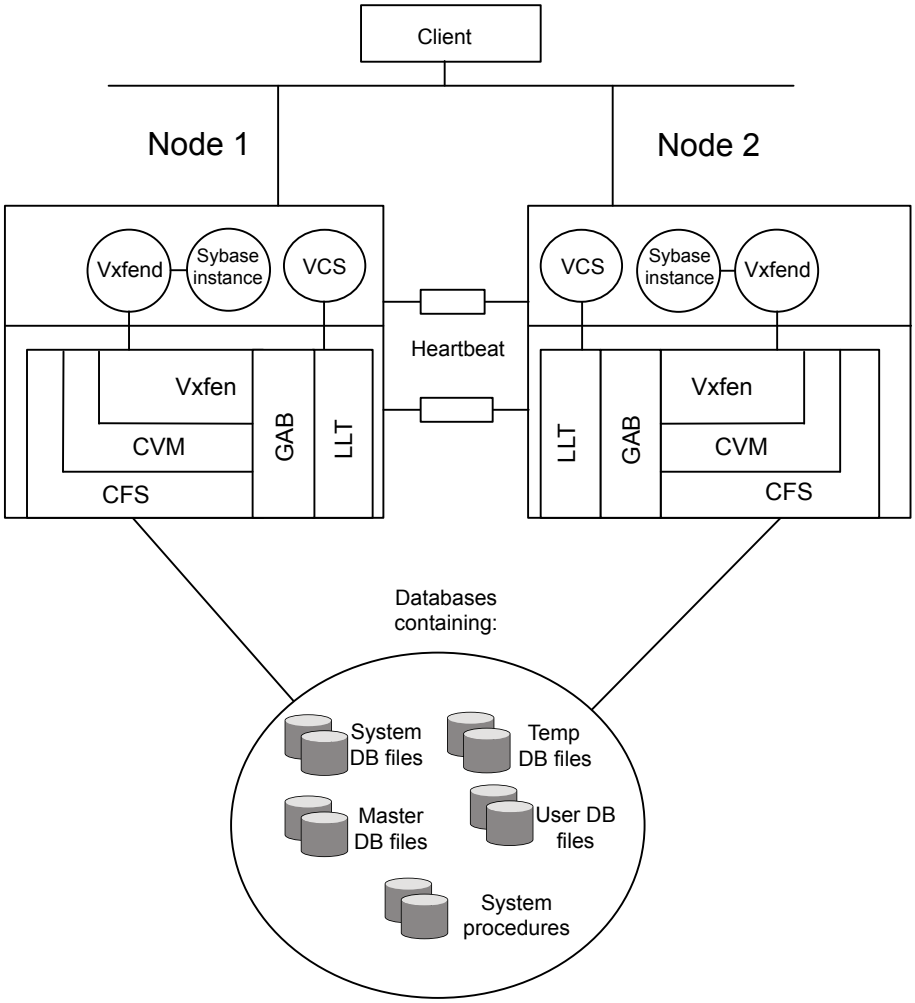


SF Sybase CE adds the following technologies to a cluster environment, which are engineered specifically to improve performance, availability, and manageability of Sybase ASE CE environments:

- Cluster File System (CFS) and Cluster Volume Manager (CVM) technologies to manage multi-instance database access to shared storage.
- VCS for cluster management
- I/O fencing to prevent split brain and protect data integrity
- DMP to provide increased availability and performance
- Veritas Cluster Membership Plug-in (VCMP) to provide interface between Sybase ASE CE cluster and the SF Sybase components
- The qrmutil interface to report the Sybase CE instance status

Figure 1-2 displays the technologies that make up the SF Sybase CE internal architecture.

Figure 1-2 SF Sybase CE architecture



SF Sybase CE provides an environment that can tolerate failures with minimal downtime and interruption to users. If a node fails as clients access the same database on multiple nodes, clients attached to the failed node can reconnect to a surviving node and resume access. Recovery after failure in the SF Sybase CE environment is far quicker than recovery for a failover database because another Sybase instance is already up and running.

About SF Sybase CE components

SF Sybase CE manages database instances running in parallel on multiple nodes using the following architecture and communication mechanisms to provide the infrastructure for Sybase ASE CE.

Table 1-1 SF Sybase CE component products

Component product	Description
Cluster Volume Manager (CVM)	Enables simultaneous access to shared volumes based on technology from Veritas Volume Manager (VxVM). See "Cluster Volume Manager (CVM)" on page 20.
Cluster File System (CFS)	Enables simultaneous access to shared file systems based on technology from Veritas File System (VxFS). See "Cluster File System (CFS)" on page 22.
Cluster Server (VCS)	Uses technology from Veritas Cluster Server to manage Sybase ASE CE databases and infrastructure components. See "Veritas Cluster Server" on page 25.
VXFEN	The VCS module prevents cluster corruption through the use of SCSI3 I/O fencing.
VXFEND	The VXFEN daemon communicates directly with VCMP and relays membership modification messages.
VCMP	VCMP provides interface between Sybase cluster and the SF Sybase CE components.
QRMUTIL	QRMUTIL provides Sybase instance status.
Sybase agent	The VCS agent is responsible for bringing Sybase ASE online, taking it offline, and monitoring it.. It obtains status by checking for processes, performing SQL queries on a running database, and interacting with QRMUTIL.

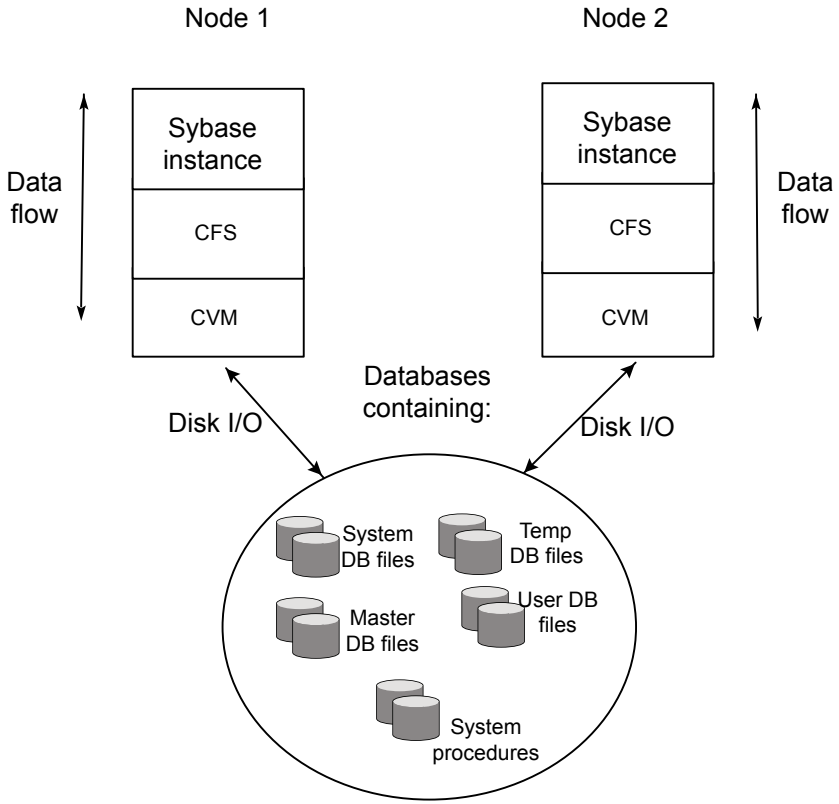
Communication infrastructure

To understand the communication infrastructure, review the data flow and communication requirements.

Data flow

The CVM, CFS, and Sybase elements reflect the overall data flow, or data stack, from an instance running on a server to the shared storage. The various Sybase processes composing an instance read and write data to the storage through the I/O stack. Sybase writes and reads to CFS, which in turn accesses the storage through CVM.

Figure 1-3 Data stack



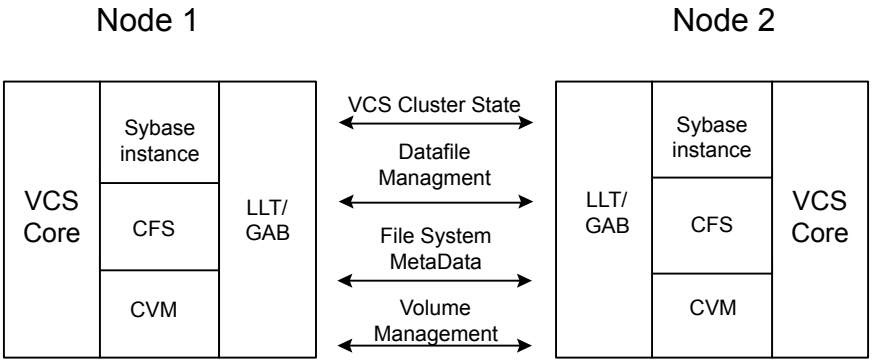
Communication requirements

End-users on a client system are unaware that they are accessing a database hosted by multiple instances. The key to performing I/O to a database accessed by multiple instances is communication between the processes. Each layer or component in the data stack must reliably communicate with its peer on other nodes to function properly. Sybase instances must communicate to coordinate

protection of data blocks in the database. SF Sybase CE processes must communicate to coordinate data file protection and access across the cluster. CFS coordinates metadata and data updates for file systems, while CVM coordinates the status of logical volumes and maps.

Figure 1-4 represents the communication stack.

Figure 1-4 Communication stack



Cluster interconnect communication channel

The cluster interconnect provides an additional communication channel for all system-to-system communication, separate from the one-node communication between modules. Low Latency Transport (LLT) and Group Membership Services/Atomic Broadcast (GAB) make up the VCS communications package central to the operation of SF Sybase CE.

Low Latency Transport

LLT provides fast, kernel-to-kernel communications and monitors network connections. LLT functions as a high performance replacement for the IP stack and runs directly on top of the Data Link Protocol Interface (DLPI) layer. The use of LLT rather than IP removes latency and overhead associated with the IP stack.

The major functions of LLT are traffic distribution, heartbeats:

- Traffic distribution
  LLT distributes (load-balances) internode communication across all available cluster interconnect links. All cluster communications are evenly distributed across as many as eight network links for performance and fault resilience. If a link fails, LLT redirects traffic to the remaining links.
- Heartbeats

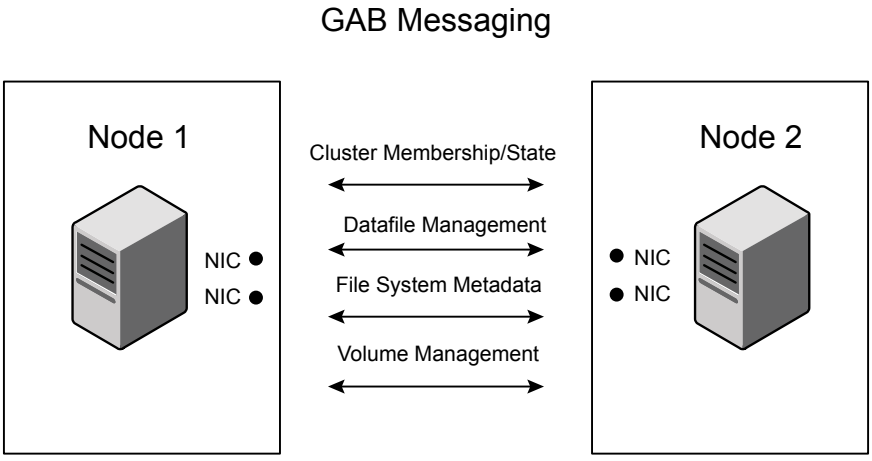
LLT is responsible for sending and receiving heartbeat traffic over network links. The Group Membership Services function of GAB uses heartbeats to determine cluster membership.

Group Membership Services/Atomic Broadcast

The GAB protocol is responsible for cluster membership and cluster communications.

Figure 1-5 shows the cluster communication using GAB messaging.

Figure 1-5 Cluster communication



Review the following information on cluster membership and cluster communication:

- Cluster membership
  At a high level, all nodes configured by the installer can operate as a cluster; these nodes form a cluster membership. In SF Sybase CE, a cluster membership specifically refers to all systems configured with the same cluster ID communicating by way of a redundant cluster interconnect.
  All nodes in a distributed system, such as SF Sybase CE, must remain constantly alert to the nodes currently participating in the cluster. Nodes can leave or join the cluster at any time because of shutting down, starting up, rebooting, powering off, or faulting processes. SF Sybase CE uses its cluster membership capability to dynamically track the overall cluster topology.

  SF Sybase CE uses LLT heartbeats to determine cluster membership:

- When systems no longer receive heartbeat messages from a peer for a predetermined interval, a protocol excludes the peer from the current membership.
- GAB informs processes on the remaining nodes that the cluster membership has changed; this action initiates recovery actions specific to each module. For example, CVM must initiate volume recovery and CFS must perform a fast parallel file system check.
- When systems start receiving heartbeats from a peer outside of the current membership, a protocol enables the peer to join the membership.

- Cluster communications

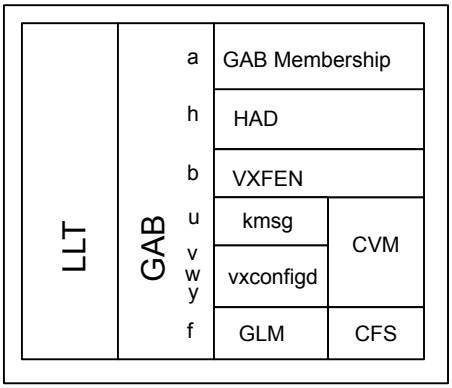
  GAB provides reliable cluster communication between SF Sybase CE modules. GAB provides guaranteed delivery of point-to-point messages and broadcast messages to all nodes. Point-to-point messaging involves sending and acknowledging the message. Atomic-broadcast messaging ensures all systems within the cluster receive all messages. If a failure occurs while transmitting a broadcast message, GAB ensures all systems have the same information after recovery.

Low-level communication: port relationship between GAB and processes

All components in SF Sybase CE use GAB for communication. Each process that wants to communicate with a peer process on other nodes registers with GAB on a specific port. This registration enables communication and notification of membership changes. For example, the VCS engine (HAD) registers on port h. HAD receives messages from peer HAD processes on port h. HAD also receives notification when a node fails or when a peer process on port h unregisters.

Some modules use multiple ports for specific communications requirements. For example, CVM uses multiple ports to allow communications by kernel and user-level functions in CVM independently.

Figure 1-6 Low-level communication



Cluster Volume Manager (CVM)

CVM is an extension of Veritas Volume Manager, the industry-standard storage virtualization platform. CVM extends the concepts of VxVM across multiple nodes. Each node recognizes the same logical volume layout, and more importantly, the same state of all volume resources.

CVM supports performance-enhancing capabilities, such as striping, mirroring, and mirror break-off (snapshot) for off-host backup. You can use standard VxVM commands from one node in the cluster to manage all storage. All other nodes immediately recognize any changes in disk group and volume configuration with no user interaction.

For detailed information, see the *Veritas Storage Foundation Cluster File System High Availability Administrator's Guide.*

CVM architecture

CVM is designed with a "master and slave" architecture. One node in the cluster acts as the configuration master for logical volume management, and all other nodes are slaves. Any node can take over as master if the existing master fails. The CVM master exists on a per-cluster basis and uses GAB and LLT to transport its configuration data.

Just as with VxVM, the Volume Manager configuration daemon, `vxconfigd`, maintains the configuration of logical volumes. This daemon handles changes to the volumes by updating the operating system at the kernel level. For example, if a mirror of a volume fails, the mirror detaches from the volume and `vxconfigd` determines the proper course of action, updates the new volume layout, and

informs the kernel of a new volume layout. CVM extends this behavior across multiple nodes and propagates volume changes to the master `vxconfigd`.

Note: You must perform operator-initiated changes on the master node.

The `vxconfigd` process on the master pushes these changes out to slave `vxconfigd` processes, each of which updates the local kernel. The kernel module for CVM is kmsg.

CVM does not impose any write locking between nodes. Each node is free to update any area of the storage. All data integrity is the responsibility of the upper application. From an application perspective, standalone systems access logical volumes in the same way as CVM systems.

CVM imposes a "Uniform Shared Storage" model. All nodes must connect to the same disk sets for a given disk group. Any node unable to detect the entire set of physical disks for a given disk group cannot import the group. If a node loses contact with a specific disk, CVM excludes the node from participating in the use of that disk.

CVM communication

CVM communication involves various GAB ports for different types of communication. For an illustration of these ports:

CVM communication involves the following GAB ports:

- Port w
  Most CVM communication uses port w for vxconfigd communications. During any change in volume configuration, such as volume creation, plex attachment or detachment, and volume resizing, vxconfigd on the master node uses port w to share this information with slave nodes.
  When all slaves use port w to acknowledge the new configuration as the next active configuration, the master updates this record to the disk headers in the VxVM private region for the disk group as the next configuration.
- Port v
  CVM uses port v for kernel-to-kernel communication. During specific configuration events, certain actions require coordination across all nodes. An example of synchronizing events is a resize operation. CVM must ensure all nodes see the new or old size, but never a mix of size among members.
  CVM also uses this port to obtain cluster membership from GAB and determine the status of other CVM members in the cluster.
- Port u

CVM uses the group atomic broadcast (GAB) port u to ship the commands from the slave node to the master node.

- Port y
  CVM uses port y for kernel-to-kernel communication required while shipping I/Os from nodes that might have lost local access to storage to other nodes in the cluster.

CVM recovery

When a node leaves a cluster, the new membership is delivered by GAB, to CVM on existing cluster nodes. The fencing driver (VXFEN) ensures that split-brain scenarios are taken care of before CVM is notified. CVM then initiates recovery of mirrors of shared volumes that might have been in an inconsistent state following the exit of the node.

Configuration differences with VxVM

CVM configuration differs from VxVM configuration in the following areas:

- Configuration commands occur on the master node.
- Disk groups are created (could be private) and imported as shared disk groups.
- Disk groups are activated per node.
- Shared disk groups are automatically imported when CVM starts.

Cluster File System (CFS)

CFS enables you to simultaneously mount the same file system on multiple nodes and is an extension of the industry-standard Veritas File System. Unlike other file systems which send data through another node to the storage, CFS is a true SAN file system. All data traffic takes place over the storage area network (SAN), and only the metadata traverses the cluster interconnect.

In addition to using the SAN fabric for reading and writing data, CFS offers Storage Checkpoint and rollback for backup and recovery.

Access to cluster storage in typical SF Sybase CE configurations use CFS. Raw access to CVM volumes is also possible but not part of a common configuration.

For detailed information, see the *Veritas Storage Foundation Cluster File System High Availability Administrator's Guide*.

CFS architecture

SF Sybase CE uses CFS to manage a file system in a large database environment. Since CFS is an extension of VxFS, it operates in a similar fashion and caches metadata and data in memory (typically called buffer cache or vnode cache). CFS uses a distributed locking mechanism called Global Lock Manager (GLM) to ensure all nodes have a consistent view of the file system. GLM provides metadata and cache coherency across multiple nodes by coordinating access to file system metadata, such as inodes and free lists. The role of GLM is set on a per-file system basis to enable load balancing.

CFS involves a primary/secondary architecture. One of the nodes in the cluster is the primary node for a file system. Though any node can initiate an operation to create, delete, or resize data, the GLM master node carries out the actual operation. After creating a file, the GLM master node grants locks for data coherency across nodes. For example, if a node tries to modify a block in a file, it must obtain an exclusive lock to ensure other nodes that may have the same file cached have this cached copy invalidated.

CFS uses port f for GLM lock and metadata communication.

CFS communication

CFS uses port f for GLM lock and metadata communication. SF Sybase CE configurations minimize the use of GLM locking except when metadata for a file changes.

CFS file system benefits

CFS adds such features as high availability, consistency and scalability, and centralized management to VxFS. Using CFS in an SF Sybase CE environment provides the following benefits:

- Increased manageability, including easy creation and expansion of files without a file system, you must provide Sybase with fixed-size partitions. With CFS, you can grow file systems dynamically to meet future requirements. Use the `vxresize` command from CVM master and CFS primary to dynamically change the size of a CFS filesystem. For more information on vxresize, refer to the vxresize(1), fsadm_vxfs(1) and chfs(1) manual pages.
- Less prone to user error
  Raw partitions are not visible and administrators can compromise them by mistakenly putting file systems over the partitions.
- Data center consistency

If you have raw partitions, you are limited to a Sybase ASE CE-specific backup strategy. CFS enables you to implement your backup strategy across the data center.

CFS configuration differences

The first node to mount a CFS file system as shared becomes the primary node for that file system. All other nodes are "secondaries" for that file system.

Mount the cluster file system individually from each node. The `-o` cluster option of the `mount` command mounts the file system in shared mode, which means you can mount the file system simultaneously on mount points on multiple nodes.

When using the `fsadm` utility for online administration functions on VxFS file systems, including file system resizing, defragmentation, directory reorganization, and querying or changing the largefiles flag, run `fsadm` from any node.

CFS recovery

The vxfsckd daemon is responsible for ensuring file system consistency when a node crashes that was a primary node for a shared file system. If the local node is a secondary node for a given file system and a reconfiguration occurs in which this node becomes the primary node, the kernel requests `vxfsckd` on the new primary node to initiate a replay of the intent log of the underlying volume. The vxfsckd daemon forks a special call to fsck that ignores the volume reservation protection normally respected by fsck and other VxFS utilities. The vxfsckd can check several volumes at once if the node takes on the primary role for multiple file systems.

After a secondary node crash, no action is required to recover file system integrity. As with any crash on a file system, internal consistency of application data for applications running at the time of the crash is the responsibility of the applications.

Comparing raw volumes and CFS for data files

Keep these points in mind about raw volumes and CFS for data files:

- If you use file-system-based data files, the file systems containing these files must be located on shared disks. Create the same file system mount point on each node.
- If you use raw devices, such as VxVM volumes, set the permissions for the volumes to be owned permanently by the database account.
  For example, type:

```
# vxedit -g dgname set group=sybase owner=sybase mode=660 \
volume_name
```

VxVM sets volume permissions on import. The VxVM volume, and any file system that is created in it, must be owned by the Sybase database user.

Veritas Cluster Server

Veritas Cluster Server (VCS) directs SF Sybase CE operations by controlling the startup and shutdown of components layers and providing monitoring and notification for failures.

In a typical SF Sybase CE configuration, the Sybase ASE CE service groups for VCS run as "parallel" service groups rather than "failover" service groups; in the event of a failure, VCS does not attempt to migrate a failed service group. Instead, the software enables you to configure the group to restart on failure.

VCS architecture

The High Availability Daemon (HAD) is the main VCS daemon running on each node. HAD tracks changes in the cluster configuration and monitors resource status by communicating over GAB and LLT. HAD manages all application services using agents, which are installed programs to manage resources (specific hardware or software entities).

The VCS architecture is modular for extensibility and efficiency. HAD does not need to know how to start up Sybase or any other application under VCS control. Instead, you can add agents to manage different resources with no effect on the engine (HAD). Agents only communicate with HAD on the local node and HAD communicates status with HAD processes on other nodes. Because agents do not need to communicate across systems, VCS is able to minimize traffic on the cluster interconnect.

SF Sybase CE provides specific agents for VCS to manage CVM, CFS, and Sybase components.

VCS communication

VCS uses port h for HAD communication. Agents communicate with HAD on the local node about resources, and HAD distributes its view of resources on that node to other nodes through GAB port h. HAD also receives information from other cluster members to update its own view of the cluster.

About the IMF notification module

The Intelligent Monitoring Framework (IMF) notification module hooks into system calls and other kernel interfaces of the operating system to get notifications on various events such as when a process starts or dies, or when a block device gets mounted or unmounted from a mount point. VCS process-based and mount-based agents use Asynchronous Monitoring Framework (AMF) kernel driver that provides asynchronous event notifications to the agents that are enabled for intelligent resource monitoring.

See "About resource monitoring" on page 26.

About resource monitoring

VCS agents poll the resources periodically based on the monitor interval (in seconds) value that is defined in the MonitorInterval or in the OfflineMonitorInterval resource type attributes. After each monitor interval, VCS invokes the monitor agent function for that resource. For example, for process offline monitoring, the process agent's monitor agent function corresponding to each process resource scans the process table in each monitor interval to check whether the process has come online. For process online monitoring, the monitor agent function queries the operating system for the status of the process id that it is monitoring. In case of the mount agent, the monitor agent function corresponding to each mount resource checks if the block device is mounted on the mount point or not. In order to determine this, the monitor function does operations such as mount table scans or runs `statfs` equivalents.

With intelligent monitoring framework (IMF), VCS supports intelligent resource monitoring in addition to poll-based monitoring. Process-based and Mount-based agents are IMF-aware. IMF is an extension to the VCS agent framework. You can enable or disable the intelligent monitoring functionality of the VCS agents that are IMF-aware.

See "How intelligent resource monitoring works" on page 27.

See "Enabling and disabling intelligent resource monitoring for agents manually" on page 61.

Poll-based monitoring can consume a fairly large percentage of system resources such as CPU and memory on systems with a huge number of resources. This not only affects the performance of running applications, but also places a limit on how many resources an agent can monitor efficiently.

However, with IMF-based monitoring you can either eliminate poll-based monitoring completely or reduce its frequency. For example, for process offline and online monitoring, you can completely avoid the need for poll-based monitoring with IMF-based monitoring enabled for processes. Similarly for vxfs

mounts, you can eliminate the poll-based monitoring with IMF monitoring enabled. Such reduction in monitor footprint will make more system resources available for other applications to consume.

Note: Intelligent Monitoring Framework for mounts is supported only for the VxFS, CFS, and NFS mount types.

With IMF-enabled agents, VCS will be able to effectively monitor larger number of resources.

Thus, intelligent monitoring has the following benefits over poll-based monitoring:

- Provides faster notification of resource state changes
- Reduces VCS system utilization due to reduced monitor function footprint
- Enables VCS to effectively monitor a large number of resources

Consider enabling IMF for an agent in the following cases:

- You have a large number of process resources or mount resources under VCS control.
- You have any of the agents that are IMF-aware.
  See the *Veritas Cluster Server Bundled Agents Reference Guide* for details on whether your bundled agent is IMF-aware.
  See the *Veritas Cluster Server Agent for Sybase Installation and Configuration Guide* for IMF-aware agents for Sybase.
  See the *Veritas Storage Foundation Cluster File System High Availability Installation Guide* for IMF-aware agents in CFS environments.

How intelligent resource monitoring works

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

See “About the IMF notification module” on page 26.

Cluster configuration files

VCS uses two configuration files in a default configuration:

- The main.cf file defines the entire cluster, including the cluster name, systems in the cluster, and definitions of service groups and resources, in addition to service group and resource dependencies.
- The types.cf file defines the resource types. Each resource in a cluster is identified by a unique name and classified according to its type. VCS includes a set of pre-defined resource types for storage, networking, and application services.

About I/O fencing in SF Sybase CE environment

I/O fencing is a mechanism to prevent uncoordinated access to the shared storage. This feature works even in the case of faulty cluster communications causing a split-brain condition. Symantec provides a technology called I/O fencing to remove the risk associated with split-brain. I/O fencing allows write access for members of the active cluster and blocks access to storage from non-members; even a node that is alive is unable to cause damage.

SCSI-3 Persistent Reservations (SCSI-3 PR) are required for I/O fencing and resolve the issues of using SCSI reservations in a clustered SAN environment. SCSI-3 PR enables access for multiple nodes to a device and simultaneously blocks access for other nodes.

Fencing involves coordinator disks and data disks. Each component has a unique purpose and uses different physical disk devices. The fencing driver, known as vxfen, directs CVM as necessary to carry out actual fencing operations at the disk group level. Fencing uses GAB port b for its communication.

In addition to providing I/O fencing capabilities, the I/O fencing module VxFEN is also used to notify Sybase ASE of membership changes on the VCS cluster. When a node is booting, VxFEN will come up after LLT and GAB, process membership information, and reach regular running state. When VxFEN later launches vxfend, the I/O fencing daemon that is used for communication, this daemon first opens a UNIX socket and registers with VCMP, a thread of Sybase ASE. The vxfend daemon is responsible for the communication between VxFEN and VCMP. If the handshake between vxfend and VCMP is successful, vxfend calls an ioctl into the VxFEN kernel module and awaits instructions. VxFEN proceeds to send the current cluster view from VCS perspective to Sybase ASE. When a connection between VxFEN and Sybase ASE has already been established, cluster membership change notification messages are delivered as soon as VxFEN completes any necessary actions (for example, after fencing out departing nodes or lost nodes).

About preferred fencing

The I/O fencing driver uses coordination points to prevent split-brain in a VCS cluster. By default, the fencing driver favors the subcluster with maximum number of nodes during the race for coordination points. With the preferred fencing feature, you can specify how the fencing driver must determine the surviving subcluster.

You can configure the preferred fencing policy using the cluster-level attribute PreferredFencingPolicy as follows:

- Enable system-based preferred fencing policy to give preference to high capacity systems.
- Enable group-based preferred fencing policy to give preference to service groups for high priority applications.
- Disable preferred fencing policy to use the default node count-based race policy.

See “Enabling or disabling the preferred fencing policy” on page 91.

About preventing data corruption with I/O fencing

I/O fencing is a feature that prevents data corruption in the event of a communication breakdown in a cluster.

To provide high availability, the cluster must be capable of taking corrective action when a node fails. In this situation, SF Sybase CE configures its components to reflect the altered membership.

Problems arise when the mechanism that detects the failure breaks down because symptoms appear identical to those of a failed node. For example, if a system in a two-node cluster fails, the system stops sending heartbeats over the private interconnects. The remaining node then takes corrective action. The failure of the private interconnects, instead of the actual nodes, presents identical symptoms and causes each node to determine its peer has departed. This situation typically results in data corruption because both nodes try to take control of data storage in an uncoordinated manner.

In addition to a broken set of private networks, other scenarios can generate this situation. If a system is so busy that it appears to stop responding or "hang," the other nodes could declare it as dead. This declaration may also occur for the nodes that use the hardware that supports a "break" and "resume" function. When a node drops to PROM level with a break and subsequently resumes operations, the other nodes may declare the system dead. They can declare it dead even if the system later returns and begins write operations.

SF Sybase CE uses I/O fencing to remove the risk that is associated with split-brain. I/O fencing allows write access for members of the active cluster. It blocks access to storage from non-members.

About SCSI-3 Persistent Reservations

SCSI-3 Persistent Reservations (SCSI-3 PR) are required for I/O fencing and resolve the issues of using SCSI reservations in a clustered SAN environment. SCSI-3 PR enables access for multiple nodes to a device and simultaneously blocks access for other nodes.

SCSI-3 reservations are persistent across SCSI bus resets and support multiple paths from a host to a disk. In contrast, only one host can use SCSI-2 reservations with one path. If the need arises to block access to a device because of data integrity concerns, only one host and one path remain active. The requirements for larger clusters, with multiple nodes reading and writing to storage in a controlled manner, make SCSI-2 reservations obsolete.

SCSI-3 PR uses a concept of registration and reservation. Each system registers its own "key" with a SCSI-3 device. Multiple systems registering keys form a membership and establish a reservation, typically set to "Write Exclusive Registrants Only." The WERO setting enables only registered systems to perform write operations. For a given disk, only one reservation can exist amidst numerous registrations.

With SCSI-3 PR technology, blocking write access is as easy as removing a registration from a device. Only registered members can "eject" the registration of another member. A member wishing to eject another member issues a "preempt and abort" command. Ejecting a node is final and atomic; an ejected node cannot eject another node. In SF Sybase CE, a node registers the same key for all paths to the device. A single preempt and abort command ejects a node from all paths to the storage device.

About I/O fencing operations

I/O fencing, provided by the kernel-based fencing module (vxfen), performs identically on node failures and communications failures. When the fencing module on a node is informed of a change in cluster membership by the GAB module, it immediately begins the fencing operation. The node tries to eject the key for departed nodes from the coordinator disks using the preempt and abort command. When the node successfully ejects the departed nodes from the coordinator disks, it also ejects the departed nodes from the data disks. In a split-brain scenario, both sides of the split would race for control of the coordinator disks. The side winning the majority of the coordinator disks wins the race and fences the loser. The loser then panics and restarts the system.

About I/O fencing components

The shared storage for SF Sybase CE must support SCSI-3 persistent reservations to enable I/O fencing. SF Sybase CE involves two types of shared storage:

- Data disks—Store shared data
  See "About data disks" on page 31.
- Coordination points—Act as a global lock during membership changes
  See "About coordination points" on page 31.

About data disks

Data disks are standard disk devices for data storage and are either physical disks or RAID Logical Units (LUNs).

These disks must support SCSI-3 PR and must be part of standard VxVM or CVM disk groups. CVM is responsible for fencing data disks on a disk group basis. Disks that are added to a disk group and new paths that are discovered for a device are automatically fenced.

About 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.

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 SF Sybase CE configuration.

You can configure coordinator disks to use Veritas Volume Manager Dynamic Multi-pathing (DMP) feature. Dynamic Multi-pathing (DMP) allows coordinator disks to take advantage of the path failover and the dynamic adding and removal capabilities of DMP. So, you can configure I/O fencing to use either DMP devices or the underlying raw character devices. I/O fencing uses SCSI-3 disk policy that is either raw or dmp based on the disk device that you use. The disk policy is raw by default. Symantec recommends using the DMP disk policy.

See the *Veritas Storage Foundation Administrator's Guide*.

How I/O fencing works in different event scenarios

Table 1-2 describes how I/O fencing works to prevent data corruption in different failure event scenarios. For each event, review the corrective operator actions.

Table 1-2 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 coordination points. If Node A wins race for coordination points, Node A ejects Node B from the shared disks and continues.	Node B races for majority of coordination points. If Node B loses the race for the coordination points, Node B panics and 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.
Node A hangs.	Node A is extremely busy for some reason or is in the kernel debugger. 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 panics and removes itself from the cluster.	Node B loses heartbeats with Node A, and races for a majority of coordination points. Node B wins race for coordination points and ejects Node A from shared data disks.	Repair or debug the node that hangs and reboot the node to rejoin the cluster.

Table 1-2 I/O fencing scenarios *(continued)*

Event	Node A: What happens?	Node B: What happens?	Operator action
Nodes A and B and private networks lose power. Coordination points and data disks retain power. Power returns to nodes and they restart, but private networks still have no power.	Node A restarts and I/O fencing driver (vxfen) detects Node B is registered with coordination points. 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: `Potentially a preexisting split brain. Dropping out of the cluster. Refer to the user documentation for steps required to clear preexisting split brain.`	Node B restarts and I/O fencing driver (vxfen) detects Node A is registered with coordination points. 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: `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 127.

Table 1-2 I/O fencing scenarios *(continued)*

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 coordination points. 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 127.
The disk array containing two of the three coordination points is powered off. No node leaves the cluster membership	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.	Power on the failed disk array so that subsequent network partition does not cause cluster shutdown, or replace coordination points. See "Replacing I/O fencing coordinator disks when the cluster is online" on page 81.

Table 1-2 I/O fencing scenarios *(continued)*

Event	Node A: What happens?	Node B: What happens?	Operator action
The disk array containing two of the three coordination points is powered off. Node B gracefully leaves the cluster and the disk array is still powered off. Leaving gracefully implies a clean shutdown so that vxfen is properly unconfigured.	Node A continues to operate in the cluster.	Node B has left the cluster.	Power on the failed disk array so that subsequent network partition does not cause cluster shutdown, or replace coordination points. See "Replacing I/O fencing coordinator disks when the cluster is online" on page 81.
The disk array containing two of the three coordination points is powered off. Node B abruptly crashes or a network partition occurs between node A and node B, and the disk array is still powered off.	Node A races for a majority of coordination points. Node A fails because only one of the three coordination points is available. Node A panics and removes itself from the cluster.	Node B has left cluster due to crash or network partition.	Power on the failed disk array and restart I/O fencing driver to enable Node A to register with all coordination points, or replace coordination points. See "Replacing defective disks when the cluster is offline" on page 132.

Sybase ASE CE components

Sybase ASE consists of a single monolithic, user space process named dataserver. A single ASE instance may consist of multiple dataserver processes, each representing an 'engine' in a single instance. The engines communicate via shared memory. ASE's internal threads run across these engines, allowing a single instance to scale to tens of thousands of concurrent users and dozens of processors on an SMP system.

Sybase ASE CE has various clustering components and a failure detection mechanism to enable multiple instances of the same database to simultaneously access it while providing protection against failures at various levels.

The following components are part of Sybase ASE CE:

- CMS (Cluster Membership Service)

Membership management is provided by CMS which is built into the dataserver binary. ASE only handles application level membership management. It is only concerned about applications, namely dataserver, running on the cluster nodes. ASE does not differentiate between a software level failure and a physical node failure.

- Quorum Device

  ASE utilizes a single quorum device to assist with membership management. Quorum device serves as a membership voting area, but also acts as a configuration repository and a semaphore for numerous operations. All access to the quorum device is through a quorum management library which exposes a common API. The cluster definition is stored in the configuration section of the quorum device. This definition includes the instances in the cluster, the nodes they run on, interconnect address, etc. This is essential information to bootstrap each instance. The quorum API provides a disk based distributed locking mechanism. This distributed lock is implemented entirely in software and requires no network communication.

  Quorum locks currently have three primary uses:

  - Race prevention at boot time
  - Configuration changes
  - Split brain prevention

  The quorum API also provides a mechanism to query the state of each instance without needing to connect to the database server.

- CIPC

  Sybase has a built-in layer known as CIPC (Cluster Inter Process Communication) to provide message passing capabilities to the various subsystems within the dataserver. Cluster instances communicate via connection oriented UDP/IP, with CIPC providing reliability on top of UDP. Sybase recommends two private networks for the cluster interconnect.

The following mechanisms are used within ASE CE:

- Heart-beating among instances

  ASE instances exchange periodic heartbeats over the cluster interconnect to signify instance health. The default period is 5 seconds, and this is dynamically configurable. There is also a dynamically configurable number of retries before which missing heartbeats translate into membership failure. Although heartbeat messages are sent explicitly, "proxy heartbeating" is also supported where any message exchange between instances during the heartbeat period can serve as a proxy for the true heartbeat message. This has improved reliability in stress situations.

The heartbeat interval can be bypassed for software failures - failures where the underlying hardware is intact. Sybase CE instances use UDP, the UDP driver on the remote node provides notification when the ASE process exists. This allows the remaining instances to immediately go into membership failure. In this situation the time from process exit to formation of the new cluster view may be under one second.

- Monitoring the health of private interconnects
  A separate mechanism called linkswitch is used to monitor the health of the two interconnect links. Linkswitch is part of the larger CIPC module. When multiple links are configured, linkswitch will detect the loss of one of the links and provide traffic switching. It also detects when a down link comes back online.

Note: The above mechanism of cluster heart-beating, linkswitch, and connected UDP allow CMS to detect the failure of the ASE process, individual interconnects, and the overall physical node (although it is not always clear which of these failures has occurred).

- Monitoring the accessibility to the disk sub-system
  A quorum heartbeat mechanism is used to determine when an instance has lost the ability to write to the disk subsystem. ASE periodically writes a heartbeat value to the quorum device. If this write fails ASE assumes that is has lost access to the disk subsystem and the instance terminates. The frequency of the heartbeat writes and the number of retries are both configurable. Note that this scheme assumes that the access to the quorum device utilizes the same fabric / SAN as the database devices.

About optional features in SF Sybase CE

SF Sybase CE supports the following activities using optional product features:

- Typical configuration of clusters in secure mode
- Typical configuration of VOM-managed clusters
- About SF Sybase CE global cluster setup for disaster recovery

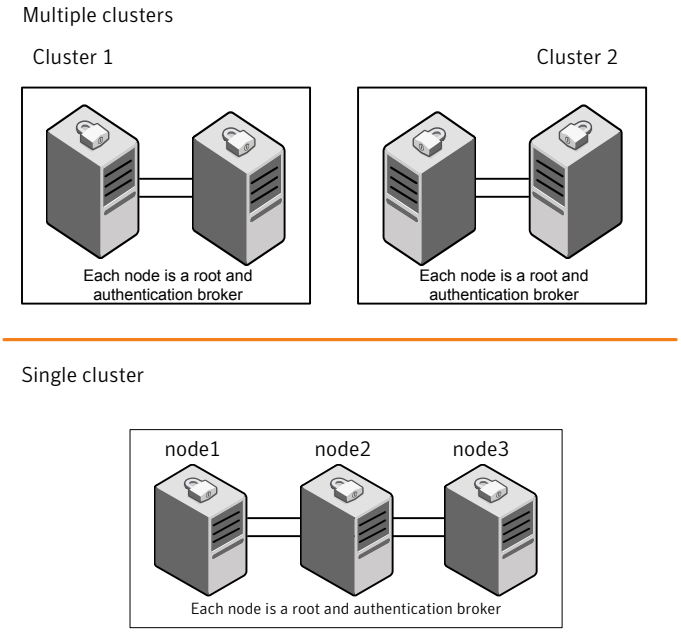
Typical configuration of clusters in secure mode

Enabling secure mode for SF Sybase CE guarantees that all inter-system communication is encrypted and that security credentials of users are verified.

Figure 1-7 illustrates typical configuration of SF Sybase CE clusters in secure mode.

For information about how to configure secure clusters, see your product installation guide.

Figure 1-7 Typical configuration of SF Sybase CE clusters in secure mode



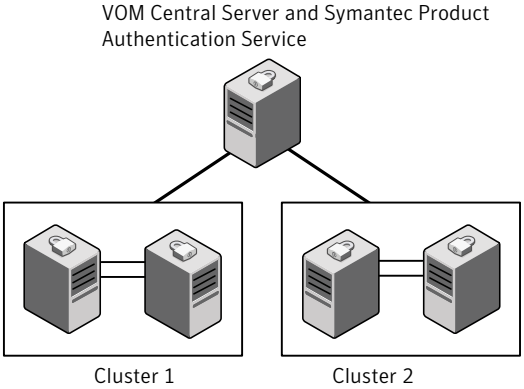
Typical configuration of VOM-managed clusters

Veritas Operations Manager (VOM) provides a centralized management console for Veritas Storage Foundation and High Availability products.

See "About Veritas Operations Manager" on page 40.

Figure 1-8 illustrates a typical setup of clusters that are centrally managed using Veritas Operations Manager.

Figure 1-8 Typical configuration of VOM-managed clusters



About SF Sybase CE global cluster setup for disaster recovery

SF Sybase CE leverages the global clustering feature of VCS to enable high availability and disaster recovery (HA/DR) for businesses that span wide geographical areas. Global clusters provide protection against outages caused by large-scale disasters such as major floods, hurricanes, and earthquakes. An entire cluster can be affected by such disasters. This type of clustering involves migrating applications between clusters over a considerable distance.

To understand how global clusters work, review the example of an Sybase ASE CE database configured using global clustering. Sybase ASE CE is installed and configured in cluster A and cluster B. Sybase database is located on shared disks within each cluster and is replicated across clusters to ensure data concurrency. The VCS service groups for Sybase are online on cluster A and are configured to fail over to cluster B.

SF Sybase CE supports host-based replication using Veritas Volume Replicator (VVR). VVR replicates data to remote sites over any standard IP network. The host at the source location on which the application is running is known as the primary host. The host at the target location is known as the secondary host.

Hardware-based replication technologies are not supported at the time of publication.

How the agent makes Sybase highly available

The Veritas Cluster Server agent for Sybase continuously monitors the Sybase database processes to verify they function properly.

The agent for Sybase can perform different levels of monitoring and different actions which you can configure.

- Primary or Basic monitoring: In the basic monitoring mode, the agent detects an application failure if a configured Sybase server process is not running.
- Detail monitoring: In the optional detail monitoring mode, the agent detects application failure if it cannot perform a transaction in the test table in the Sybase database server.
- Intelligent monitoring framework (IMF): The Sybase agent also supports intelligent monitoring framework (IMF) in the process check mode of basic monitoring. The Sybase agent is IMF-aware and uses asynchronous monitoring framework (AMF) kernel driver for resource state change notifications.

For Sybase cluster edition, the agent uses the Sybase-provided utility, `qrmutil`, to know if the status of the instance is up or down. If `qrmutil` reports the status as failure pending, the agent reboots the node and the instance is automatically started again.

When the Sybase agent detects that the configured Sybase server is not running on a system, based on the value of OnlineRetryLimit attribute of the Sybase server group, the service group is restarted on the same system on which the group faulted.

About Veritas Operations Manager

Symantec recommends use of Veritas Operations Manager to manage Storage Foundation and Cluster Server environments.

Veritas Operations Manager provides a centralized management console for Veritas Storage Foundation and High Availability products. You can use Veritas Operations Manager to monitor, visualize, and manage storage resources and generate reports.

You can download Veritas Operations Manager at no charge at http://go.symantec.com/vom.

Refer to the Veritas Operations Manager documentation for installation, upgrade, and configuration instructions.

Symantec Operations Readiness Tools

Symantec Operations Readiness Tools (SORT) is a Web site that automates and simplifies some of the most time-consuming administrative tasks. SORT helps

you manage your datacenter more efficiently and get the most out of your Symantec products.

Among its broad set of features, SORT lets you do the following:

- Generate server-specific reports that describe how to prepare your servers for installation or upgrade of Symantec enterprise products.
- Access a single site with the latest production information, including patches, agents, and documentation.
- Create automatic email notifications for changes in patches, documentation, and array-specific modules.

To access SORT, go to:

https://sort.symantec.com

Chapter 2

Administering SF Sybase CE and its components

This chapter includes the following topics:

- Administering SF Sybase CE
- Administering VCS
- Administering I/O fencing
- Administering CVM
- Administering CFS
- Administering the Sybase agent

Administering SF Sybase CE

This section provides instructions for the following SF Sybase CE administration tasks:

- Setting the environment variables
  See "Setting the environment variables" on page 44.
- Starting or stopping SF Sybase CE on each node
  See "Starting or stopping SF Sybase CE on each node" on page 44.
- Applying Sybase patches
- Installing Veritas Volume Manager or Veritas File System patches
- Applying operating system updates
  See "Applying operating system updates" on page 48.
- Adding storage to an SF Sybase CE cluster

See "Adding storage to an SF Sybase CE cluster" on page 49.

- Recovering from storage failure
  See "Recovering from storage failure" on page 50.
- Enhancing the performance of SF Sybase CE clusters
  See "Enhancing the performance of SF Sybase CE clusters" on page 51.
- Verifying the nodes in a cluster
  See "Verifying the nodes in a cluster" on page 51.

If you encounter issues while administering SF Sybase CE, refer to the troubleshooting section for assistance.

Setting the environment variables

Set the MANPATH variable in the .profile file (or other appropriate shell setup file for your system) to enable viewing of manual pages.

Based on the shell you use, type one of the following:

For sh, ksh, or bash
```
# export MANPATH=$MANPATH:\
/opt/VRTS/man
```

For csh
```
# setenv MANPATH $MANPATH:/opt/VRTS/man
```

Some terminal programs may display garbage characters when you view the man pages. Set the environment variable LC_ALL=C to resolve this issue.

Set the PATH environment variable in the .profile file (or other appropriate shell setup file for your system) on each system to include installation and other commands.

Based on the shell you use, type one of the following:

For sh, ksh, or bash
```
# PATH=/usr/sbin:/sbin:/usr/bin:\
/opt/VRTS/bin\
$PATH; export PATH
```

Starting or stopping SF Sybase CE on each node

You can start or stop SF Sybase CE on each node in the cluster using the SF Sybase CE installer or manually.

To start SF Sybase CE	Using installer: See "Starting SF Sybase CE using the SF Sybase CE installer" on page 45. Manual: See "Starting SF Sybase CE manually on each node" on page 45.
To stop SF Sybase CE	Using installer: See "Stopping SF Sybase CE using the SF Sybase CE installer" on page 46. Manual: See "Stopping SF Sybase CE manually on each node" on page 47.

Starting SF Sybase CE using the SF Sybase CE installer

Run the SF Sybase CE installer with the `-start` option to start SF Sybase CE on each node.

To start SF Sybase CE using the SF Sybase CE installer

1. Log into one of the nodes in the cluster as the root user.
2. Start SF Sybase CE:

```
# /opt/VRTS/install/installsfsybasece -start system1 system2
```

Starting SF Sybase CE manually on each node

Perform the steps in the following procedures to start SF Sybase CE manually on each node.

To start SF Sybase CE manually on each node

1. Log into each node as the root user.
2. Start LLT:

```
# /etc/init.d/llt start
```

3. Start GAB:

```
# /etc/init.d/gab start
```

4 Start fencing:

```
# /etc/init.d/vxfen start
```

5 Start GLM:

```
# /etc/init.d/vxglm start
```

6 Start VCS, CVM, and CFS:

```
# hastart
```

7 Verify that all GAB ports are up and running:

```
# gabconfig -a
GAB Port Memberships
===============================================================
Port a gen   564004 membership 01
Port b gen   564008 membership 01
Port f gen   564024 membership 01
Port h gen   56401a membership 01
Port u gen   564021 membership 01
Port v gen   56401d membership 01
Port w gen   56401f membership 01
Port y gen   56401c membership 01
```

Stopping SF Sybase CE using the SF Sybase CE installer

Run the SF Sybase CE installer with the `-stop` option to stop SF Sybase CE on each node.

To stop SF Sybase CE using the SF Sybase CE installer

1 Log into one of the nodes in the cluster as the root user.

2 Unmount the VxFS file system, which is not under VCS control.

```
# mount |grep vxfs
# fuser -m /mount_point
# umount /mount_point
```

Make sure that no processes are running which make use of mounted shared file system or shared volumes.

```
# fuser -cu /mount_point
```

3 Stop VCS:

```
# hastop -all
```

4 Stop SF Sybase CE:

```
# /opt/VRTS/install/installsfsybasece -stop system1 system2
```

Stopping SF Sybase CE manually on each node

Perform the steps in the following procedures to stop SF Sybase CE manually on each node.

To stop SF Sybase CE manually on each node

1 Stop the Sybase database.

If the Sybase ASE CE instance is managed by VCS, log in as the root user and take the service group offline:

```
# hagrp -offline sybase_group -sys node_name
```

2 Stop all applications that are not configured under VCS but dependent on Sybase ASE CE or resources controlled by VCS. Use native application commands to stop the application.

3 Unmount the VxFS file systems that are not managed by VCS.

Make sure that no processes are running, which make use of mounted shared file system or shared volumes:

```
# mount -v | grep vxfs
# fuser -cu /mount_point
```

Unmount the VxFS file system:

```
# umount /mount_point
```

4 Take the VCS service groups offline:

```
# hagrp -offline group_name -sys node_name
```

Verify that the VCS service groups are offline:

```
# hagrp -state group_name
```

When Sybase group under VCS control is offline, the vxfend daemon also stops.

5 Stop VCS, CVM and CFS:

```
# hastop -local
```

Verify that the ports 'f', 'u', 'v', 'w', 'y', and 'h' are closed:

```
# gabconfig -a
GAB Port Memberships
===============================================================
Port a gen   761f03 membership 01
Port b gen   761f08 membership 01
```

6 Stop GLM:

```
# /etc/init.d/vxglm stop
```

7 Stop fencing:

```
# /etc/init.d/vxfen stop
```

8 Stop GAB:

```
# /etc/init.d/gab stop
```

9 Stop LLT:

```
# /etc/init.d/llt stop
```

Applying operating system updates

If you need to apply updates to the base version of the operating system, perform the steps in this section on each node of the cluster, one node at a time.

To apply operating system updates

1 Log in to the node as the root user and change to `/opt/VRTS/install` directory:

```
# cd /opt/VRTS/install
```

2 Take the VCS service groups offline:

```
# hagrp -offline grp_name -sys node_name
```

3 Stop SF Sybase CE:

```
# ./installsfsybasece -stop
```

4 Upgrade the operating system.

For instructions, see the operating system documentation.

5 If the node is not rebooted after the operating system upgrade, reboot the node:

```
# shutdown -r now
```

6 Repeat all the steps on each node in the cluster.

Adding storage to an SF Sybase CE cluster

You can add storage to an SF Sybase CE cluster in the following ways:

Extend the volume space on a disk group	Use the `vxresize` command to change the length of a volume containing a file system. It automatically locates available disk space on the specified volume and frees up unused space to the disk group for later use. See the `vxresize` (1M) manual page for information on various options. See "To add storage to an SF Sybase CE cluster by extending the volume space on a disk group" on page 50.
Add a disk to a disk group	Use the `vxdg` command to add a disk to a disk group. See the `vxdg` (1M) manual page for information on various options. See "To add storage to an SF Sybase CE cluster by adding a disk to a disk group" on page 50.

To add storage to an SF Sybase CE cluster by extending the volume space on a disk group

1 Determine the length by which you can increase an existing volume.

```
# vxassist [-g diskgroup] maxsize
```

For example, to determine the maximum size the volume `sybvol` in the disk group `sybdata_101` can grow, given its attributes and free storage available:

```
# vxassist -g sybdata_101 maxsize
```

2 Extend the volume, as required. You can extend an existing volume to a certain length by specifying the new size of the volume (the new size must include the additional space you plan to add). You can also extend a volume by a certain length by specifying the additional amount of space you want to add to the existing volume.

To extend a volume to a certain length	For example, to extend the volume `sybvol` of size 10 GB in the disk group `sybdata_101` to 30 GB:
	`# vxresize -g sybdata_101 \`
	`sybvol 30g`

To extend a volume by a certain length	For example, to extend the volume `sybvol` of size 10 GB in the disk group `sybdata_101` by 10 GB:
	`# vxresize -g sybdata_101 \`
	`sybvol +10g`

To add storage to an SF Sybase CE cluster by adding a disk to a disk group

◆ Add a disk to the disk group:

```
# vxdg -g dg_name adddisk disk_name
```

Recovering from storage failure

Veritas Volume Manager (VxVM) protects systems from disk and other hardware failures and helps you to recover from such events. Recovery procedures help you prevent loss of data or system access due to disk and other hardware failures.

For information on various failure and recovery scenarios, see the *Veritas Volume Manager Troubleshooting Guide*.

Enhancing the performance of SF Sybase CE clusters

The main components of clustering that impact the performance of an SF Sybase CE cluster are:

- Kernel components, specifically LLT and GAB
- VCS engine (had)
- VCS agents

Each VCS 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 functionality that is particular to an agent. For various options provided by the clustering components to monitor and enhance performance, see the chapter "VCS performance considerations" in the *Veritas Cluster Server Administrator's Guide*.

Veritas Volume Manager can improve system performance by optimizing the layout of data storage on the available hardware. For more information on tuning Veritas Volume Manager for better performance, see the chapter "Performance monitoring and tuning" in the *Veritas Storage Foundation Administrator's Guide*.

Veritas Volume Replicator Advisor (VRAdvisor) is a planning tool that helps you determine an optimum Veritas Volume Replicator (VVR) configuration. For installing VRAdvisor and evaluating various parameters using the data collection and data analysis process, see the *Veritas Storage Foundation and High Availability Solutions Replication Administrator's Guide*.

Verifying the nodes in a cluster

Table 2-1 lists the various options that you can use to periodically verify the nodes in your cluster.

Table 2-1 Options for verifying the nodes in a cluster

Type of check	Description
Symantec Operations Readiness Tools (SORT)	Use the Symantec Operations Readiness Tools (SORT) to evaluate your systems before and after any installation, configuration, upgrade, patch updates, or other routine administrative activities. The utility performs a number of compatibility and operational checks on the cluster that enable you to diagnose and troubleshoot issues in the cluster. The utility is periodically updated with new features and enhancements. For more information and to download the utility, visit http://sort.symantec.com.

Administering VCS

This section provides instructions for the following VCS administration tasks:

- Viewing available Veritas devices and drivers
  See "Viewing available Veritas device drivers" on page 52.
- Starting and stopping VCS
  See "Starting and stopping VCS" on page 54.
- Environment variables to start and stop VCS modules
  See "Environment variables to start and stop VCS modules" on page 55.
- Adding and removing LLT links
  See "Adding and removing LLT links" on page 57.
- Displaying the cluster details and LLT version for LLT links
  See "Displaying the cluster details and LLT version for LLT links" on page 60.
- Configuring aggregated interfaces under LLT
  See "Configuring aggregated interfaces under LLT" on page 59.
- Configuring destination-based load balancing for LLT
  See "Configuring destination-based load balancing for LLT" on page 61.
- Enabling and disabling intelligent resource monitoring
  See "Enabling and disabling intelligent resource monitoring for agents manually" on page 61.
- Administering the AMF kernel driver
  See "Administering the AMF kernel driver" on page 63.

If you encounter issues while administering VCS, see the troubleshooting section in the *Veritas Cluster Server Administrator's Guide* for assistance.

Viewing available Veritas device drivers

To view the available Veritas devices:

```
# cat /proc/devices
Character devices:
  1 mem
  2 pty
  3 ttyp
  4 /dev/vc/0
  4 tty
  4 ttyS
  5 /dev/tty
```

```
  5 /dev/console
  5 /dev/ptmx
  7 vcs
 10 misc
 13 input
 21 sg
 29 fb
128 ptm
136 pts
180 usb
189 usb_device
199 VxVM
200 VxSPEC
250 vxfen
251 vxglm
252 gab
253 llt
254 amf

Block devices:
  1 ramdisk
  2 fd
  3 ide0
  7 loop
  8 sd
  9 md
 65 sd
 66 sd
 67 sd
 68 sd
 69 sd
 70 sd
 71 sd
128 sd
129 sd
130 sd
131 sd
132 sd
133 sd
134 sd
135 sd
199 VxVM
201 VxDMP
```

```
253 device-mapper
254 mdp
```

To view the drivers that are loaded in memory, run the `lsmod` command as shown in the following examples.

For example:

If you want to view whether or not the driver 'gab' is loaded in memory:

```
# lsmod |grep "^gab"
gab      246420  11 vxfen,vcsmm

gab                   214912  8
```

If you want to view whether or not the 'vx' drivers are loaded in memory:

```
# lsmod |grep "^vx"
vxfen                 260888  1
vxodm                 159820  1
vxglm                 267400  2
vxgms                 285692  2
vxspec                  4360  4
vxio                 3090616  3 vxspec
vxdmp                 341064  92 vxspec,vxio
vxportal                7820  2
vxfs                 2326872  11 vxportal,fdd

vxfen                 224260  1
vxglm                 250264  1
vxspec                 10840  12
vxio                 5917172  3 [unsafe]
vxdmp                 278696  34
vxportal                6656  2
vxfs                 2188336  4 vxportal,fdd
scsi_mod              146321  9 vxdmp,ib_srp,libata,qla2xxx,\
scsi_transport_fc,mptsas,mptspi,mptscsi,sd_mod
```

Starting and stopping VCS

To start VCS on each node:

```
# hastart
```

To stop VCS on each node:

```
# hastop -local
```

You can also use the command `hastop -all`; however, make sure that you wait for port 'h' to close before restarting VCS.

Environment variables to start and stop VCS modules

The start and stop environment variables for AMF, 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 AMF, LLT, GAB, VxFEN, and VCS engine are inter-dependent. For a clean startup or shutdown of SF Sybase CE, you must either enable or disable the startup and shutdown modes for all these modules.

Table 2-2 describes the start and stop variables for VCS.

Table 2-2 Start and stop environment variables for VCS

Environment variable	Definition and default value
AMF_START	Startup mode for the AMF driver. By default, the AMF driver is enabled to start up after a system reboot. This environment variable is defined in the following file: `/etc/sysconfig/amf` Default: 1
AMF_STOP	Shutdown mode for the AMF driver. By default, the AMF driver is enabled to stop during a system shutdown. This environment variable is defined in the following file: `/etc/sysconfig/amf` Default: 1
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: `/etc/sysconfig/llt` Default: 1

Table 2-2 Start and stop environment variables for VCS *(continued)*

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

Table 2-2 Start and stop environment variables for VCS *(continued)*

Environment variable	Definition and default value
VCS_STOP	Shutdown mode for VCS engine. By default, VCS engine is enabled to stop during a system shutdown. This environment variable is defined in the following file: `/etc/sysconfig/vcs` Default: 1

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.

Note: When you add or remove LLT links, you need not shut down GAB or the high availability daemon, `had`. Your changes take effect immediately, but are lost on the next restart. For changes to persist, you must also update the `/etc/llttab` file.

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:

    ```
    # lltconfig -t devtag -d device
    -b udp6 [-s port] [-m mtu]
    -I IPaddr -B mcast
    ```

Where:

devtag	Tag to identify the link
device	Device name of the interface. For link type ether, you can specify the device name as an interface name. For example, eth0. Preferably, specify the device name as eth-macaddress. For example, eth- xx:xx:xx:xx:xx:xx. For link types udp and udp6, the device is the udp and udp6 device name 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:

- For ether link type:

```
# lltconfig -t eth4 -d eth4 -s 0xcafe -m 1500
```

- For UDP link type:

```
# lltconfig -t link1 -d udp -b udp
-I 192.1.2.255 -B 192.1.2.255
```

- For UDP6 link type:

```
# lltconfig -t link1 -d udp6 -b udp6 -I 2000::1
```

To remove an LLT link

◆ Run the following command to remove a network link that is configured under LLT:

```
# lltconfig -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 `lltconfig` 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, stop LLT after you stop the other dependent modules.

   ```
   # /etc/init.d/llt stop
   ```

   See "Starting or stopping SF Sybase CE on each node" on page 44.

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
   ```

tag	Tag to identify the link
device_name	Device name of the aggregated interface.
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. Restart the other dependent modules that you stopped in step 1.

   ```
   # /etc/init.d/llt start
   ```

   See "Starting or stopping SF Sybase CE on each node" on page 44.

To configure aggregated interfaces under LLT using the lltconfig command

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

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

devtag	Tag to identify the link
device	Device name of the aggregated interface.
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 59.

Displaying the cluster details and LLT version for LLT links

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

See the `lltdump`(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 eth2 is connected to galaxy, then the command displays a list of all cluster IDs and node IDs present on the network link eth2.

  ```
  # /opt/VRTSllt/lltdump -D -f eth2

  lltdump : Configuration:

  device : eth2

  sap : 0xcafe
  promisc sap : 0
  promisc mac : 0
  cidsnoop : 1
  === Listening for LLT packets ===
  cid nid vmaj vmin
  3456 1 5 0
  3456 3 5 0
  83 0 4 0
  27 1 3 7
  3456 2 5 0
  ```

Configuring destination-based load balancing for LLT

Destination-based load balancing for LLT is turned off by default. Symantec recommends destination-based load balancing when the cluster setup has more than two nodes and more active LLT ports.

To configure destination-based load balancing for LLT

- Run the following command to configure destination-based load balancing:

  ```
  lltconfig -F linkburst:0
  ```

Enabling and disabling intelligent resource monitoring for agents manually

Review the following procedures to enable or disable intelligent resource monitoring manually. The intelligent resource monitoring feature is enabled by

default. The IMF resource type attribute determines whether an IMF-aware agent must perform intelligent resource monitoring.

See "About resource monitoring" on page 26.

To enable intelligent resource monitoring

1. Make the VCS configuration writable.

```
# haconf -makerw
```

2. Run the following command to enable intelligent resource monitoring.

   - To enable intelligent monitoring of offline resources:

```
# hatype -modify resource_type IMF -update Mode 1
```

   - To enable intelligent monitoring of online resources:

```
# hatype -modify resource_type IMF -update Mode 2
```

   - To enable intelligent monitoring of both online and offline resources:

```
# hatype -modify resource_type IMF -update Mode 3
```

3. Change the values of the MonitorFreq key and the RegisterRetryLimit key of the IMF attribute.

4. Save the VCS configuration.

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

5. Make sure that the AMF kernel driver is configured on all nodes in the cluster.

```
/etc/init.d/amf status
```

   If the AMF kernel driver is configured, the output resembles:

```
AMF: Module loaded and configured
```

   Configure the AMF driver if the command output returns that the AMF driver is not loaded or not configured.

   See "Administering the AMF kernel driver" on page 63.

6. Restart the agent. Run the following commands on each node.

```
# haagent -stop agent_name -force -sys sys_name
# haagent -start agent_name -sys sys_name
```

To disable intelligent resource monitoring

1 Make the VCS configuration writable.

```
# haconf -makerw
```

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

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

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

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

4 Save the VCS configuration.

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

Note: VCS provides haimfconfig script to enable or disable the IMF functionality for agents. You can use the script with VCS in running or stopped state. Use the script to enable or disable IMF for the IMF-aware bundled agents, enterprise agents, and custom agents. For more information, refer to the "Enabling and disabling IMF for agents by using script" section in the *Veritas™ Cluster Server Administrator's Guide*.

Administering the AMF kernel driver

Review the following procedures to start or stop the AMF kernel driver.

See "About the IMF notification module" on page 26.

See "Environment variables to start and stop VCS modules" on page 55.

To start the AMF kernel driver

1 Set the value of the AMF_START variable to 1 in the following file, if the value is not already 1:

```
/etc/sysconfig/amf
```

2 Start the AMF kernel driver. Run the following command:

```
/etc/init.d/amf start
```

To stop the AMF kernel driver

1 Set the value of the AMF_START variable to 1 in the following file, if the value is not already 1:

```
/etc/sysconfig/amf
```

2 Stop the AMF kernel driver. Run the following command:

```
/etc/init.d/amf stop
```

Administering I/O fencing

This section describes I/O fencing and provides instructions for common I/O fencing administration tasks.

- About administering I/O fencing
  See "About administering I/O fencing" on page 64.
- About vxfentsthdw utility
  See "About the vxfentsthdw utility" on page 65.
- About vxfenadm utility
  See "About the vxfenadm utility" on page 73.
- About vxfenclearpre utility
  See "About the vxfenclearpre utility" on page 78.
- About vxfenswap utility
  See "About the vxfenswap utility" on page 81.

If you encounter issues while administering I/O fencing, refer to the troubleshooting section for assistance.

See "Troubleshooting I/O fencing" on page 126.

About administering I/O fencing

The I/O fencing feature provides the following utilities that are available through the VRTSvxfen rpm:

vxfentsthdw	Tests SCSI-3 functionality of the disks for I/O fencing See "About the vxfentsthdw utility" on page 65.
vxfenconfig	Configures and unconfigures I/O fencing Lists the coordination points 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 73.
vxfenclearpre	Removes SCSI-3 registrations and reservations from disks See "About the vxfenclearpre utility" on page 78.
vxfenswap	Replaces coordination points without stopping I/O fencing See "About the vxfenswap utility" on page 81.
vxfendisk	Generates the list of paths of disks in the diskgroup. This utility requires that Veritas Volume Manager is installed and configured.

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 vxfentsthdw utility

You can use the vxfentsthdw 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 Storage Foundation for Sybase ASE CE Installation and Configuration 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 `vxfentsthdw`(1M) manual page.

About general guidelines for using vxfentsthdw utility

Review the following guidelines to use the vxfentsthdw utility:

- 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 vxfentsthdw 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 that the nodes see the same disk" on page 77.

- 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/sdx is ready to be configured for
  I/O Fencing on node system1
  ```

  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.

About the vxfentsthdw command options

Table 2-3 describes the methods that the utility provides to test storage devices.

Table 2-3 vxfentsthdw options

vxfentsthdw 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 69.

Table 2-3 vxfentsthdw options *(continued)*

vxfentsthdw option	Description	When to use
-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 disk group. If you want the script to use the raw paths for disks in the disk group, 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 disk group. 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 vxfentsthdw -c option" on page 68.
-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 vxfentsthdw -m option" on page 70.
-f *filename*	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 vxfentsthdw -f option" on page 72.

Table 2-3 vxfentsthdw options *(continued)*

vxfentsthdw option	Description	When to use
-g *disk_group*	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 disk group using the vxfentsthdw -g option" on page 72.

Testing the coordinator disk group using vxfentsthdw -c option

Use the vxfentsthdw utility to verify disks are configured to support I/O fencing. In this procedure, the vxfentsthdw utility tests the three disks one disk at a time from each node.

The procedure in this section uses the following disks for example:

- From the node system1, the disks are seen as /dev/sdg, /dev/sdh, and /dev/sdi.
- From the node system2, the same disks are seen as /dev/sdx, /dev/sdy, and /dev/sdz.

Note: To test the coordinator disk group using the vxfentsthdw utility, the utility requires that the coordinator disk group, vxfencoorddg, be accessible from two nodes.

To test the coordinator disk group using vxfentsthdw -c

1. Use the vxfentsthdw command with the -c option. For example:

```
# vxfentsthdw -c vxfencoorddg
```

2. Enter the nodes you are using to test the coordinator disks:

```
Enter the first node of the cluster: system1
Enter the second node of the cluster: system2
```

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/sdg have PASSED.
The disk is now ready to be configured for I/O Fencing on node
system1 as a COORDINATOR DISK.

ALL tests on the disk /dev/sdx have PASSED.
The disk is now ready to be configured for I/O Fencing on node
system2 as a COORDINATOR DISK.
```

4 After you test all disks in the disk group, the vxfencoorddg 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 vxfencoorddg 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 Storage Foundation Administrator's Guide*.

2 Add a new disk to the node, initialize it, and add it to the coordinator disk group.

See the *Veritas Storage Foundation for Sybase ASE CE Installation and Configuration Guide* for instructions to initialize disks for I/O fencing and to set up coordinator disk groups.

If necessary, start the disk group.

See the *Veritas Storage Foundation Administrator's Guide* for instructions to start the disk group.

3 Retest the disk group.

See "Testing the coordinator disk group using vxfentsthdw -c option" on page 68.

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:

  ```
  # vxfentsthdw -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 using the vxfentsthdw -m option

Review the procedure to test the shared disks. By default, the utility uses the -m option.

This procedure uses the /dev/sdx 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 vxfentsthdw utility indicates a disk can be used for I/O fencing with a message resembling:

```
The disk /dev/sdx is ready to be configured for
I/O Fencing on node system1
```

Note: For A/P arrays, run the `vxfentsthdw` command only on active enabled paths.

To test disks using vxfentsthdw script

1. Make sure system-to-system communication is functioning properly.
2. From one node, start the utility.

   ```
   # vxfentsthdw [-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: system1
Enter the second node of the cluster: system2
```

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
system1 in the format:
        for dmp: /dev/vx/rdmp/sdx
        for raw: /dev/sdx
Make sure it's the same disk as seen by nodes system1 and system2
/dev/sdr

Enter the disk name to be checked for SCSI-3 PGR on node
system2 in the format:
        for dmp: /dev/vx/rdmp/sdx
        for raw: /dev/sdx
Make sure it's the same disk as seen by nodes system1 and system2
/dev/sdr
```

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/sdx have PASSED
The disk is now ready to be configured for I/O Fencing on node
system1
...
Removing test keys and temporary files, if any ...
.
.
```

7 Run the vxfentsthdw utility for each disk you intend to verify.

Testing the shared disks listed in a file using the vxfentsthdw -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 system1 and system2 that might resemble:

   ```
   system1 /dev/sdz system2 /dev/sdy
   system1 /dev/sdu system2 /dev/sdw
   ```

   Where the first disk is listed in the first line and is seen by system1 as /dev/sdz and by system2 as /dev/sdy. The other disk, in the second line, is seen as /dev/sdu from system1 and /dev/sdw from system2. Typically, the list of disks could be extensive.

2. To test the disks, enter the following command:

   ```
   # vxfentsthdw -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 disk group using the vxfentsthdw -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 with the `vxdg import` command.

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:

   ```
   # vxfentsthdw -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 consist 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 (consumes four bytes).

By default, CVM uses unique fencing key for each disk group. However, some arrays have a restriction on the total number of unique keys that can be registered. In such cases, you can use the `same_key_for_alldgs` tunable parameter to change the default behavior. The default value of the parameter is off. If your configuration

hits the storage array limit on total number of unique keys, you can turn the value on using the `vxdefault` command as follows:

```
# vxdefault set same_key_for_alldgs on
# vxdefault list
KEYWORD              CURRENT-VALUE   DEFAULT-VALUE
...
same_key_for_alldgs  on              off
...
```

If the tunable is changed to 'on', all subsequent keys that the CVM generates on disk group imports or creates have '0000' as their last four bytes (DGcount is 0). You must deport and re-import all the disk groups that are already imported for the changed value of the `same_key_for_alldgs` tunable to take effect.

Displaying the I/O fencing registration keys

You can display the keys that are currently assigned to the disks using the vxfenadm command.

The variables such as *disk_7*, *disk_8*, and *disk_9* in the following procedure represent the disk names in your setup.

To display the I/O fencing registration keys

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/sdx from the system with node ID 1, enter the following command:

```
# vxfenadm -s /dev/sdx
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: system1
```

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]:  66,80,71,82,48,48,48,49
  [Character Format]: BPGR0001
  [Node Format]: Cluster ID: unknown  Node ID: 1 Node Name: system2
```

- 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: system1
```

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: system1
```

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 look up 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: system2
```

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 that 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/sdr path on node A and the /dev/sdt 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 SF Sybase CE.
2. From node A, enter the following command:

```
# vxfenadm -i /dev/sdr

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/sdt path.

On a disk from another manufacturer, Hitachi Data Systems, the output is different and may resemble:

```
# vxfenadm -i /dev/sdt

Vendor id       : HITACHI
Product id      : OPEN-3
Revision        : 0117
Serial Number   : 0401EB6F0002
```

Refer to the `vxfenadm`(1M) manual page for more information.

About the vxfenclearpre utility

You can use the vxfenclearpre utility to remove SCSI-3 registrations and reservations on the disks.

See "Removing preexisting keys" on page 78.

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 on all nodes.

   ```
   # hastop -all
   ```

2. Make sure that the port h is closed on all the nodes. Run the following command on each node to verify that the port h is closed:

   ```
   # gabconfig -a
   ```

   Port h must not appear in the output.

3. Stop I/O fencing on all nodes. Enter the following command on each node:

   ```
   # /etc/init.d/vxfen stop
   ```

4. 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.

5. Start the vxfenclearpre script:

   ```
   # vxfenclearpre
   ```

6 Read the script's introduction and warning. Then, you can choose to let the script run.

```
Do you still want to continue: [y/n] (default : n) y
```

In some cases, informational messages resembling the following may appear on the console of one of the nodes in the cluster when a node is ejected from a disk/LUN. You can ignore these informational messages.

```
<date> <system name> scsi: WARNING: /sbus@3,0/lpfs@0,0/
sd@0,1(sd91):
<date> <system name> Error for Command: <undecoded
cmd 0x5f> Error Level: Informational
<date> <system name> scsi: Requested Block: 0 Error Block 0
<date> <system name> scsi: Vendor: <vendor> Serial Number:
0400759B006E
<date> <system name> scsi: Sense Key: Unit Attention
<date> <system name> scsi: ASC: 0x2a (<vendor unique code
0x2a>), ASCQ: 0x4, FRU: 0x0
```

The script cleans up the disks and displays the following status messages.

```
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
reboot the system.
```

7 Start the fencing module.

```
# /etc/init.d/vxfen start
```

8 Start VCS on all nodes.

```
# hastart
```

About the vxfenswap utility

The vxfenswap utility allows you to add, remove, and replace coordinator points 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.

Refer to the `vxfenswap`(1M) manual page.

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 81.
  See "Replacing the coordinator diskgroup in a cluster that is online" on page 84.
  If you want to replace the coordinator disks when the cluster is offline, you cannot use the vxfenswap 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 132.
- 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 86.
- The keys that are registered on the coordinator disks are lost.
  In such a case, the cluster might panic when a network partition 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 89.

If the vxfenswap operation is unsuccessful, then you can use the `-a cancel` 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.

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 a coordinator diskgroup when the cluster is online

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: SYBASE
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
  * 0 (system1)
  1 (system2)
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, vxfencoorddg) 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. If your setup uses VRTSvxvm version 5.1 RP2 (or later) or 5.1 SP1 (or later), then skip to step 5. You need not set `coordinator=off` to add or remove disks. For other VxVM versions, perform this step:

   Turn off the coordinator attribute value for the coordinator disk group.

```
# vxdg -g vxfencoorddg set coordinator=off
```

5. To remove disks from the coordinator 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 the new disks to the coordinator disk group and set the coordinator attribute value as "on" for the coordinator disk group.

See the *Veritas Storage Foundation for Sybase ASE CE Installation and Configuration Guide* for detailed instructions.

Note that though the disk group content changes, the I/O fencing remains in the same state.

7 From one node, start the vxfenswap utility. You must specify the diskgroup to the utility.

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.
- Reads the diskgroup you specified in the vxfenswap 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.

8 If the disk verification passes, the utility reports success and asks if you want to commit the new set of coordinator disks.

9 Confirm whether you want to clear the keys on the coordination points and proceed with the vxfenswap operation.

```
Do you want to clear the keys on the coordination points
and proceed with the vxfenswap operation? [y/n] (default: n) y
```

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 11.

```
Do you wish to commit this change? [y/n] (default: n) y
```

If the utility successfully commits, the utility 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 vxfendg.

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: SYBASE
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
  * 0 (system1)
  1 (system2)
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
sda    auto:cdsdisk    -  (vxfendg)    online
sdb    auto:cdsdisk    -  (vxfendg)    online
sdc    auto:cdsdisk    -  (vxfendg)    online
sdx    auto:cdsdisk    -  (vxfencoorddg)    online
sdy    auto:cdsdisk    -  (vxfencoorddg)    online
sdz    auto:cdsdisk    -  (vxfencoorddg)    online
```

5 Validate the new disk group for I/O fencing compliance. Run the following command:

```
# vxfentsthdw -c vxfendg
```

See "Testing the coordinator disk group using vxfentsthdw -c option" on page 68.

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 Storage Foundation for Sybase ASE CE Installation and Configuration Guide* for more information.

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.
- Reads the diskgroup you specified in the vxfenswap 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 utility reports success and asks if you want to replace the coordinator disk group.

10 Confirm whether you want to clear the keys on the coordination points and proceed with the vxfenswap operation.

```
Do you want to clear the keys on the coordination points
and proceed with the vxfenswap operation? [y/n] (default: n) y
```

11 Review the message that the utility displays and confirm that you want to replace the coordinator disk group. Else skip to step 14.

```
Do you wish to commit this change? [y/n] (default: n) y
```

If the utility successfully commits, the utility moves the /etc/vxfentab.test file to the /etc/vxfentab file.

The utility also updates the /etc/vxfendg file with this new diskgroup.

12 Set the coordinator attribute value as "on" for the new coordinator disk group.

```
# vxdg -g vxfendg set coordinator=on
```

Set the coordinator attribute value as "off" for the old disk group.

```
# vxdg -g vxfencoorddg set coordinator=off
```

13 Verify that the coordinator disk group has changed.

```
# cat /etc/vxfendg
vxfendg
```

The swap operation for the coordinator disk group is complete now.

14 If you do not want to replace the coordinator disk group, answer n at the prompt.

The vxfenswap utility rolls back any changes to the coordinator diskgroup.

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 dmp to raw 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.

```
# vxfenadm -d


I/O Fencing Cluster Information:
================================
Fencing Protocol Version: 201
Fencing Mode: SYBASE
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
  * 0 (system1)
  1 (system2)
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=sybase
scsi3_disk_policy=raw
```

4. From any node, start the vxfenswap utility:

```
# vxfenswap -g vxfencoordg [-n]
```

5. Verify the change in the disk policy.

```
# vxfenadm -d


I/O Fencing Cluster Information:
================================
Fencing Protocol Version: 201
Fencing Mode: SYBASE
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.

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: SYBASE
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
  * 0 (system1)
  1 (system2)
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
sdx    auto:cdsdisk    -  (vxfencoorddg)    online
sdy    auto    -  -    offline
sdz    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/sdx       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 vxfencoorddg [-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/sdx       32  48  R450 00013154 0312     dmp
/dev/vx/rdmp/sdy       32  32  R450 00013154 0313     dmp
/dev/vx/rdmp/sdz       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 network partition 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: SYBASE
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
  * 0 (system1)
  1 (system2)
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/sdx
Total Number of Keys: 0
No keys...
...
```

4. On any node, run the following command to start the vxfenswap utility:

```
# vxfenswap -g vxfencoorddg [-n]
```

5. Verify that the keys are atomically placed on the coordinator disks.

```
# vxfenadm -s all -f /etc/vxfentab


Device Name: /dev/vx/rdmp/sdx
Total Number of Keys: 4
...
```

Enabling or disabling the preferred fencing policy

You can enable or disable the preferred fencing feature for your I/O fencing configuration.

You can enable preferred fencing to use system-based race policy or group-based race policy. If you disable preferred fencing, the I/O fencing configuration uses the default count-based race policy.

See "About preferred fencing" on page 29.

To enable preferred fencing for the I/O fencing configuration

1. Make sure that the cluster is running with I/O fencing set up.

   ```
   # vxfenadm -d
   ```

2. Make sure that the cluster-level attribute UseFence has the value set to SCSI3.

   ```
   # haclus -value UseFence
   ```

3. To enable system-based race policy, perform the following steps:

   - Make the VCS configuration writable.

     ```
     # haconf -makerw
     ```

   - Set the value of the cluster-level attribute PreferredFencingPolicy as System.

     ```
     # haclus -modify PreferredFencingPolicy System
     ```

   - Set the value of the system-level attribute FencingWeight for each node in the cluster.
     For example, in a two-node cluster, where you want to assign system1 five times more weight compared to system2, run the following commands:

     ```
     # hasys -modify system1 FencingWeight 50
     # hasys -modify system2 FencingWeight 10
     ```

   - Save the VCS configuration.

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

4. To enable group-based race policy, perform the following steps:

   - Make the VCS configuration writable.

```
# haconf -makerw
```

- Set the value of the cluster-level attribute PreferredFencingPolicy as Group.

```
# haclus -modify PreferredFencingPolicy Group
```

- Set the value of the group-level attribute Priority for each service group. For example, run the following command:

```
# hagrp -modify service_group Priority 1
```

  Make sure that you assign a parent service group an equal or lower priority than its child service group. In case the parent and the child service groups are hosted in different subclusters, then the subcluster that hosts the child service group gets higher preference.

- Save the VCS configuration.

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

5 To view the fencing node weights that are currently set in the fencing driver, run the following command:

```
# vxfenconfig -a
```

To disable preferred fencing for the I/O fencing configuration

1 Make sure that the cluster is running with I/O fencing set up.

```
# vxfenadm -d
```

2 Make sure that the cluster-level attribute UseFence has the value set to SCSI3.

```
# haclus -value UseFence
```

3 To disable preferred fencing and use the default race policy, set the value of the cluster-level attribute PreferredFencingPolicy as Disabled.

```
# haconf -makerw
# haclus -modify PreferredFencingPolicy Disabled
# haconf -dump -makero
```

Administering CVM

This section provides instructions for the following CVM administration tasks:

- Listing all the CVM shared disks
  See "Listing all the CVM shared disks" on page 145.
- Establishing CVM cluster membership manually
  See "Establishing CVM cluster membership manually" on page 93.
- Changing CVM master manually
  See "Changing the CVM master manually" on page 94.
- Importing a shared disk group manually
  See "Importing a shared disk group manually" on page 97.
- Deporting a shared disk group manually
  See "Deporting a shared disk group manually " on page 97.
- Starting shared volumes manually
  See "Starting shared volumes manually" on page 145.
- Verifying if CVM is running in an SF Sybase CE cluster
  See "Verifying if CVM is running in an SF Sybase CE cluster" on page 97.
- Verifying CVM membership state
  See "Verifying CVM membership state" on page 98.
- Verifying the state of CVM shared disk groups
  See "Verifying the state of CVM shared disk groups" on page 98.
- Verifying the activation mode
  See "Verifying the activation mode" on page 99.

If you encounter issues while administering CVM, refer to the troubleshooting section for assistance.

See "Troubleshooting Cluster Volume Manager" on page 134.

Establishing CVM cluster membership manually

In most cases you do not have to start CVM manually; it normally starts when VCS is started.

Run the following command to start CVM manually:

```
# vxclustadm -m vcs -t gab startnode

vxclustadm: initialization completed
```

Note that `vxclustadm` reads main.cf for cluster configuration information and is therefore not dependent upon VCS to be running. You do not need to run the `vxclustadm startnode` command as normally the `hastart` (VCS start) command starts CVM automatically.

To verify whether CVM is started properly:

```
# vxclustadm nidmap
Name          CVM Nid     CM Nid      State
system1        0           0         Joined: Master
system2        1           1         Joined: Slave
```

Changing the CVM master manually

You can change the CVM master manually from one node in the cluster to another node, while the cluster is online. CVM migrates the master node, and reconfigures the cluster.

Symantec recommends that you switch the master when the cluster is not handling VxVM configuration changes or cluster reconfiguration operations. In most cases, CVM aborts the operation to change the master, if CVM detects that any configuration changes are occurring in the VxVM or the cluster. After the master change operation starts reconfiguring the cluster, other commands that require configuration changes will fail until the master switch completes.

See "Errors during CVM master switching" on page 96.

To change the master online, the cluster must be cluster protocol version 100 or greater.

To change the CVM master manually

1 To view the current master, use one of the following commands:

```
# vxclustadm nidmap
Name              CVM Nid    CM Nid    State
system1            0          0         Joined: Slave
system2            1          1         Joined: Master
```

```
# vxdctl -c mode
mode: enabled: cluster active - MASTER
master: system2
```

In this example, the CVM master is system2.

2 From any node on the cluster, run the following command to change the CVM master:

```
# vxclustadm setmaster nodename
```

where *nodename* specifies the name of the new CVM master.

The following example shows changing the master on a cluster from system2 to system1:

```
# vxclustadm setmaster system1
```

3 To monitor the master switching, use the following command:

```
# vxclustadm -v nodestate
 state: cluster member
        nodeId=0
        masterId=0
        neighborId=1
        members[0]=0xf
        joiners[0]=0x0
        leavers[0]=0x0
        members[1]=0x0
        joiners[1]=0x0
        leavers[1]=0x0
        reconfig_seqnum=0x9f9767
        vxfen=off
state: master switching in progress
reconfig: vxconfigd in join
```

In this example, the state indicates that master is being changed.

4 To verify whether the master has successfully changed, use one of the following commands:

```
# vxclustadm nidmap
Name              CVM Nid    CM Nid    State
system1            0          0         Joined: Master
system2            1          1         Joined: Slave
```

```
# vxdctl -c mode
mode: enabled: cluster active - MASTER
master: system1
```

Errors during CVM master switching

Symantec recommends that you switch the master when the cluster is not handling VxVM or cluster configuration changes.

In most cases, CVM aborts the operation to change the master, if CVM detects any configuration changes in progress. CVM logs the reason for the failure into the system logs. In some cases, the failure is displayed in the `vxclustadm setmaster` output as follows:

```
# vxclustadm setmaster system1
VxVM vxclustadm ERROR V-5-1-0 Master switching, a reconfiguration or
```

```
a transaction is in progress.
Try again
```

In some cases, if the master switching operation is interrupted with another reconfiguration operation, the master change fails. In this case, the existing master remains the master of the cluster. After the reconfiguration is complete, reissue the `vxclustadm setmaster` command to change the master.

If the master switching operation has started the reconfiguration, any command that initiates a configuration change fails with the following error:

```
Node processing a master-switch request. Retry operation.
```

If you see this message, retry the command after the master switching has completed.

Importing a shared disk group manually

You can use the following command to manually import a shared disk group:

```
# vxdg -s import dg_name
```

Deporting a shared disk group manually

You can use the following command to manually deport a shared disk group:

```
# vxdg deport dg_name
```

Note that the deport of a shared disk group removes the SCSI-3 PGR keys on the disks.

Verifying if CVM is running in an SF Sybase CE cluster

You can use the following options to verify whether CVM is up or not in an SF Sybase CE cluster.

The following output is displayed on a node that is not a member of the cluster:

```
# vxdctl -c mode
mode: enabled: cluster inactive
# vxclustadm -v nodestate
state: out of cluster
```

On the master node, the following output is displayed:

```
# vxdctl -c mode
```

```
mode: enabled: cluster active - MASTER
master: system1
```

On the slave nodes, the following output is displayed:

```
# vxdctl -c mode
```

```
mode: enabled: cluster active - SLAVE
master: system2
```

The following command lets you view all the CVM nodes at the same time:

```
# vxclustadm nidmap
```

```
Name       CVM Nid    CM Nid     State
system1          0          0     Joined: Master
system2          1          1     Joined: Slave
```

Verifying CVM membership state

The state of CVM can be verified as follows:

```
# vxclustadm -v nodestate
```

```
state: joining
        nodeId=0
        masterId=0
        neighborId=0
        members=0x1
        joiners=0x0
        leavers=0x0
        reconfig_seqnum=0x0
        reconfig: vxconfigd in join
```

The state indicates that CVM has completed its kernel level join and is in the middle of vxconfigd level join.

The `vxdctl -c mode` command indicates whether a node is a CVM master or CVM slave.

Verifying the state of CVM shared disk groups

You can use the following command to list the shared disk groups currently imported in the SF Sybase CE cluster:

```
# vxdg list |grep shared
```

```
sybbindg_101 enabled,shared 1052685125.1485.csha3
```

Verifying the activation mode

In an SF Sybase CE cluster, the activation of shared disk group should be set to “shared-write” on each of the cluster nodes.

To verify whether the “shared-write”activation is set:

```
# vxdg list dg_name |grep activation
```

```
local-activation: shared-write
```

If "shared-write" activation is not set, run the following command:

```
# vxdg -g dg_name set activation=sw
```

Administering CFS

This section describes some of the major aspects of cluster file system administration.

This section provides instructions for the following CFS administration tasks:

- Adding CFS file systems to VCS configuration
  See “Adding CFS file systems to VCS configuration” on page 99.
- Resizing CFS file systems
  See “Resizing CFS file systems” on page 100.
- Verifying the status of CFS file systems
  See “Verifying the status of CFS file systems” on page 100.

If you encounter issues while administering CFS, refer to the troubleshooting section for assistance.

Adding CFS file systems to VCS configuration

Run the following command to add a CFS file system to the VCS main.cf file without using an editor.

For example:

```
# cfsmntadm add quorum_101 quorumvol /quorum sybasece \
all=suid,rw
```

```
Mount Point is being added...
  /quorum added to the cluster-configuration
```

Using cfsmount to mount CFS file systems

To mount a CFS file system using cfsmount:

```
# cfsmntadm add sdg vol1 /quorum all=
# cfsmount /quorum
Mounting...
[/dev/vx/rdsk/quorum_101/quorum
mounted successfully at /quorumvol on system1
[/dev/vx/rdsk/quorum_101/quorumvol]
mounted successfully at /quorum on system2
```

Resizing CFS file systems

If you see a message on the console indicating that a CFS file system is full, you may want to resize the file system. The `vxresize` command lets you resize a CFS file system. It extends the file system and the underlying volume.

See the `vxresize` (1M) manual page for information on various options.

The following command resizes an Sybase binary CFS file system (the Sybase binary volume is CFS mounted):

```
# vxresize -g sybbindg sybbinvol +2G
```

where sybbindg is the CVM disk group,sybbinvol is the volume, and +2G indicates the increase in volume size by 2 Gigabytes.

Verifying the status of CFS file systems

Run the "cfscluster status" command to see the status of the nodes and their mount points:

```
# cfscluster status

Node              :  system2
  Cluster Manager  :  not-running
  CVM state        :  not-running
  MOUNT POINT    SHARED VOLUME  DISK GROUP         STATUS

  /quorum       quorumvol       quorum_101         NOT MOUNTED
  /sybase        sybbinvol      sybbindg            NOT MOUNTED
  /sybdata       sybvol         sybdata_101         NOT MOUNTED

  Node              :  system1
  Cluster Manager  :  running
```

```
CVM state        :  running
MOUNT POINT    SHARED VOLUME  DISK GROUP        STATUS

  /quorum       quorumvol       quorum_101        MOUNTED
/sybase        sybbinvol      sybbindg           MOUNTED
/sybdata       sybvol         sybdata_101        MOUNTED
```

Administering the Sybase agent

SF Sybase CE includes the VCS Sybase agent. The agent can perform different operations or functions on the database. These functions are online, offline, monitor, and clean.

Sybase agent functions

The agent for Sybase starts a Sybase ASE dataserver, monitors the server processes, and shuts down the server.

The Sybase agent is IMF-aware.

Table 2-4 lists the Sybase agent for SQL server functions.

Table 2-4 Sybase agent for SQL server functions

Agent function	Description
Online	Starts the Sybase ASE dataserver by using the following command. `startserver -f $SYBASE/$SYBASE_ASE/install/RUN_$Server` where $Server is the instance_name, and `$SYBASE/$SYBASE_ASE/install/RUN_$Server` is the default location of the Run server file. If you specify the value of the Run_ServerFile attribute, then the value that you specify is used instead of the default location. If the WaitForRecovery attribute is enabled, the agent waits either till recovery has been completed and all databases that can be made online are brought online. The agent queries the recovery status by connecting to the `isql` session. The OnlineTimeout attribute must be set to a sufficiently large value so that the recovery completes before the OnlineTimeout is reached. By default, the WaitForRecovery attribute is not enabled. If the interfaces file location is specified using the interfaces_File attribute, agent uses [-I interfaces file] option while connecting to the isql session. When DelayAfterOnline attribute is set, the monitor function is invoked after completion of online function, and after the number of seconds specified in DelayAfterOnline attribute have elapsed.
Monitor	In the basic monitoring mode, agent scans the process table for the dataserver process. In the detail monitoring mode, agent runs the script that is specified in MonScript as an option. The agent uses the Sybase provided utility, `qrmutil`, to know if the status of the instance is up or down. If qrmutil reports the status as failure pending, the agent reboots the node and the instance is automatically started again. See "Monitoring options for the Sybase agent" on page 104.

Table 2-4 Sybase agent for SQL server functions *(continued)*

Agent function	Description
Offline	Stops the Sybase SQL server by using the `isql` command in the following manner. If interfaces file location is specified using the interfaces_File attribute, agent uses the specified file while connecting to isql session. The agent first executes the `shutdown with wait` command. Sybase agent uses the `timeout` option during shutdown of Sybase dataserver if this option is supported. For Sybase ASE Cluster edition the timeout option for shutdown command is supported from versions 15.5 ESD #1 onwards. If the timeout option is not supported for Sybase ASE Cluster edition, the offline script waits in a loop till the dataserver completely stops. The agent waits for up to the OfflineTimeout duration. If the process is still running, the offline script kills it. When DelayAfterOffline attribute is set, the monitor function is invoked after completion of offline function and after the number of seconds specified in DelayAfterOffline attribute have elapsed.
sybase_imf_init	Initializes the agent to interface with the AMF kernel driver, which is the IMF notification module for Sybase agent. This function runs when the agent starts up.
sybase_imf_register	Registers or unregisters resource entities with the AMF kernel module. This function runs for each resource after the resource goes into steady state (online or offline).
sybase_imf_getnotification	Gets notification about resource state changes. This function runs after the agent initializes with the AMF kernel module. This function continuously waits for notification and takes action on the resource upon notification.
Clean	Forcefully stops the Sybase SQL server by using the `isql` command in the following manner. The agent first executes the `shutdown with wait` command. For Sybase ASE Cluster edition, if the `shutdown with wait` command does not stop the dataserver, the agent directly proceeds to kill the dataserver process.

Monitoring options for the Sybase agent

The Veritas agent for Sybase provides two levels of application monitoring: basic and detail.

In the basic monitoring mode, the agent for Sybase monitors the Sybase daemon processes to verify whether they are running.

For Sybase cluster edition, the agent uses `qrmutil` utility that Sybase provides to get the status of the Sybase instance. If the state returned by `qrmutil` utility is 'failure pending', the agent panics the node. When the Sybase agent detects that the configured Sybase server is not running on a system, based on the value of the OnlineRetryLimit attribute of the Sybase service group, the service group is restarted on the same system on which the group faulted.

For example:

```
# qrmutil --quorum_dev=/quorum/quorum.dat --monitor=ase1
Executing 'monitor' command for instance 'ase1'
Instance 'ase1' has a failure pending.
# echo $?
99
```

In this example instance ‘ase1’ has a failure pending state. The agent will panic the node running the instance ‘ase1’. The node will automatically rejoin the cluster after reboot.

In the detail monitoring mode, the agent performs a transaction on a test table in the database to ensure that Sybase functions properly. The agent uses this test table for internal purposes. Symantec recommends that you do not perform any other transaction on the test table.

Using the IPC Cleanup feature for the Sybase agent

When the Adaptive Server starts, it creates shared memory files in $SYBASE to store information about the shared memory segments that it uses. Adaptive Server start-up parameter -M can be used to change the location of directory that stores shared memory files. The start-up parameter -M should be updated in RUN_$Server file.

If the Sybase home directory is unmounted, the Sybase clean script cannot access the shared memory files and does not clean the IPC resources that are allocated by the Sybase processes. Hence, the agent requires shared memory files to be present in the following directory on local system `/var/tmp/sybase_shm/$Server`.

In the $SYBASE/$SYBASE_ASE/install directory, edit the RUN_$Server file. Change the location of the directory that stores shared memory files to /var/tmp/sybase_shm/$Server using the -M option.

For example, the file RUN_Sybase_Server resembles the following before the change:

```
/home/sybase/ASE-15_0/bin/dataserver \

-sSybase_Server \

-d/home/sybase/data/master.dat \

-e/home/sybase/ASE-15_0/install/Sybase_Server.log \

-c/home/sybase/ASE-15_0/Sybase_Server.cfg \

-M/home/sybase/ASE-15_0 \

--quorum_dev=/qrmmnt/qfile
```

After the replacement, the file resembles:

```
/home/sybase/ASE-15_0/bin/dataserver \

-sSybase_Server \

-d/home/sybase/data/master.dat \

-e/home/sybase/ASE-15_0/install/Sybase_Server.log \

-c/home/sybase/ASE-15_0/Sybase_Server.cfg \

-M/var/tmp/sybase_shm/Sybase_Server \
```

Here Sybase_Server is the name of the Adaptive server.

Note: Make sure you create the /var/tmp/sybase_shm/Sybase_Server directory with proper permissions.

Configuring the service group for Sybase from Cluster Manager (Java console)

A template for the Sybase resource groups is automatically installed with the Veritas agent for Sybase. Using the VCS cluster Manager (Java console), you can view the template, which displays the Sybase service group, its resources and their attributes. You can dynamically modify the attributes' values as necessary for your configuration.

Note: Before you start the following procedure, for more instructions on Java GUI, refer to the *Veritas Cluster Server 6.0 Installation Guide*.

To configure a service group from the Java console

1. Make sure that the Sybase type definition file SybaseTypes.cf is imported in your configuration.
2. Launch the Cluster Configuration wizard using any of the following ways:
   - From the Cluster Explorer menu, select **Tools > Configuration Wizard**.
   - If no service groups exist on the system, Cluster Explorer prompts you to launch the Cluster Configuration wizard. Click **Yes** when prompted.

   The Loading Templates Information window appears, and launches the wizard.
3. Review the information in the Welcome dialog box and click **Next**.
4. Specify the name of the service group and the target systems on which the service group is configured.
   - Enter the name of the service group.
   - From the **Available Systems** box, select the systems on which to configure the service group.
   - Click the right arrow to move the selected systems to the **Systems for Service Group** box. To remove a system from the box, select the system and click the left arrow.
   - Specify system priority for the service group. System priority is numbered sequentially, with the lowest assigned number denoting the highest priority.
   - Select the **Service Group Type** as Parallel and click **Next**.
5. On the **Would you like to use a template to configure the service group?** dialog box, click **Next** to configure the service group using a template.

6 Select the **SybaseGroup** template to configure a Sybase service group.

If applicable, a window opens to notify that names of some resources within the new service group are already in use. Resolve the name clashes, if any and click **Next**.

7 Click **Next** to create the service group that is based on the selected template.

A progress indicator displays the percentage of the commands that are executed to create the service group. The actual commands are displayed at the top of the indicator.

8 After the service group is created, click **Next** to edit the attributes of the resources.

The left pane in the dialog box lists all the resources for the Sybase service group. Select a resource from the left pane to list the attributes on the right pane. The attributes in bold denote mandatory attributes. You can modify the attribute values as given in the procedure that follows.

9 Click **Finish** to accept the default values and complete the configuration.

To edit the Sybase service group resource attributes

1 Select the resource from the list on the left pane. The resource attributes appear in the right pane.

2 Select the attribute to be modified and click the edit icon in the **Edit** column.

3 In the Edit Attribute dialog box, enter the attribute values. To modify the scope of the attribute, choose the **Global** or **Local** option.

4 Click **OK**.

5 Repeat the procedure for each resource and click **Finish**. Edit the attributes for all the resources according to your configuration.

6 Follow the wizard instructions to complete the configuration. Click **Finish** to quit the wizard.

Caution: For added security, you must always provide a secure value for passwords.

Configuring the service group for Sybase using the command line

The Veritas agent for Sybase contains a sample configuration file that can be used as reference to directly modify your present configuration file. This method requires you to restart VCS before the configuration takes effect.

To configure a service group for Sybase from the command line

1. Log in to a cluster system as superuser.
2. Make sure the Sybase type definition is imported into VCS engine.
3. Edit the main.cf file at `/etc/VRTSvcs/conf/config/main.cf`. For reference, use the sample files at `/etc/VRTSagents/ha/conf/Sybase`.
   - Create binmnt group to configure CFS mounts for Sybase binaries.
   - Create sybasece service group.
   - Create Sybase resources.
   - Edit the default attributes to match the parameters in your configuration. For added security, you must always provide a secure value for passwords.
   - Assign dependencies to the newly created resources. Refer to the sample file at `/etc/VRTSagents/ha/conf/Sybase/`.
     See the *Veritas Cluster Server User's Guide* for more information on assigning dependencies.
4. Save and close the file.
5. Verify the syntax of the file `/etc/VRTSvcs/conf/config/main.cf`

   ```
   # cd /etc/VRTSvcs/conf/config
   # hacf -verify .
   ```

6. Start VCS on the local node.

   ```
   # hastart
   ```

7. Start VCS on the other nodes.
8. If the system is listed in AutoStartList attribute of the Sybase service group, verify that all Sybase service group resources are brought online.

   ```
   # hagrp -state
   ```

9. Take the service group offline and verify that all the resources are stopped.

   ```
   # hagrp -offline service_group -sys system_name

   # hagrp -state
   ```

10 Bring the service group online again and verify that all the resources are available.

```
# hagrp -online service_group -sys system_name

# hagrp -state
```

11 On all systems, look at the following log files for any errors or status.

```
/var/VRTSvcs/log/engine_A.log
/var/VRTSvcs/log/Sybase_A.log
```

Bringing the Sybase service group online

Perform the following steps to bring a service group online. Note that in the initial few cycles of bringing a service group online, the memory usage by the agent can spike.

To bring a service group online

1 From Cluster Explorer, click the **Service Groups** tab in the configuration tree.

2 Right-click the service group and click **Enable Resources** to enable all the resources in this group.

3 Right-click the service group, hover over **Enable**, and select either the node or all the nodes where you want to enable the service group.

4 Save and close the configuration. Click **File > Save Configuration**, then **Close Configuration**.

5 Right-click the service group, pause over Online, and select the system where you want to bring the service group online.

Taking the Sybase service group offline

Perform the following procedure from Cluster Manager (Java Console) to take the service group offline. Note that in the initial few cycles of taking a service group offline, the agent's memory usage can spike.

To take a service group offline

1 In the Cluster Explorer configuration tree with the Service Groups tab selected, right-click the service group that you want to take offline.

2 Choose **Offline**, and select the appropriate system from the pop-up menu.

Modifying the Sybase service group configuration

You can dynamically modify the Sybase agent using several methods, including the Cluster Manager (Java Console), Veritas Operations Manager, and the command line.

See the *Veritas Cluster Server User's Guide* for more information.

Viewing the agent log for Sybase

The Veritas agent for Sybase logs messages to the following files:

/var/VRTSvcs/log/engine_A.log

/var/VRTSvcs/log/Sybase_A.log

Chapter 3

Troubleshooting SF Sybase CE

This chapter includes the following topics:

- About troubleshooting SF Sybase CE
- What to do if you see a licensing reminder
- Restarting the installer after a failed connection
- Installer cannot create UUID for the cluster
- Troubleshooting I/O fencing
- Troubleshooting Cluster Volume Manager
- Troubleshooting interconnects
- Troubleshooting Sybase ASE CE

About troubleshooting SF Sybase CE

Use the information in this chapter to diagnose setup or configuration problems that you might encounter. For issues that arise from the component products, it may be necessary to refer to the appropriate documentation to resolve it.

Gathering information for support analysis

Use troubleshooting scripts to gather information about the configuration and status of your cluster and its modules. The scripts identify rpm information, debugging messages, console messages, and information about disk groups and

volumes. Forwarding the output of these scripts to Symantec Tech Support can assist with analyzing and solving any problems.

- Gathering configuration information using VRTSexplorer
  See "Gathering configuration information using SORT Data Collector" on page 112.
- Gathering VCS information for support analysis
  See "Gathering VCS information for support analysis" on page 112.
- Gathering LLT and GAB information for support analysis
  See "Gathering LLT and GAB information for support analysis" on page 113.
- Gathering IMF information for support analysis
  See "Gathering IMF information for support analysis" on page 113.

Gathering configuration information using SORT Data Collector

SORT Data Collector now supersedes the VRTSexplorer utility.

Run the Data Collector with the `VxExplorer` option to gather system and configuration information from a node to diagnose or analyze issues in the cluster.

If you find issues in the cluster that require professional help, run the Data Collector and send the tar file output to Symantec Technical Support to resolve the issue.

Visit the SORT Website and download the UNIX Data Collector appropriate for your operating system:

https://sort.symantec.com

For more information:

https://sort.symantec.com/public/help/wwhelp/wwhimpl/js/html/wwhelp.htm

Gathering VCS information for support analysis

You must run the `hagetcf` command to gather information when you encounter issues with VCS. Symantec Technical Support uses the output of these scripts to assist with analyzing and solving any VCS problems. The `hagetcf` command gathers information about the installed software, cluster configuration, systems, logs, and related information and creates a gzip file.

See the `hagetcf`(1M) manual page for more information.

To gather VCS information for support analysis

◆ Run the following command on each node:

```
# /opt/VRTSvcs/bin/hagetcf
```

The command prompts you to specify an output directory for the gzip file. You may save the gzip file to either the default `/tmp` directory or a different directory.

Troubleshoot and fix the issue.

If the issue cannot be fixed, then contact Symantec technical support with the file that the `hagetcf` command generates.

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. The `getcomms` script also collects core dump and stack traces along with the LLT and GAB information.

To gather LLT and GAB information for support analysis

1 If you had changed the default value of the GAB_FFDC_LOGDIR parameter, you must again export the same variable before you run the `getcomms` script.

See "GAB message logging" on page 117.

2 Run the following command to gather information:

```
# /opt/VRTSgab/getcomms
```

The script uses rsh by default. Make sure that you have configured passwordless rsh. If you have passwordless ssh between the cluster nodes, you can use the `-ssh` option. To gather information on the node that you run the command, use the `-local` option.

Troubleshoot and fix the issue.

If the issue cannot be fixed, then contact Symantec technical support with the file `/tmp/commslog.`*`time_stamp`*`.tar` that the `getcomms` script generates.

Gathering IMF information for support analysis

You must run the `getimf` script to gather information when you encounter issues with IMF (Intelligent Monitoring Framework).

To gather IMF information for support analysis

- Run the following command on each node:

  ```
  # /opt/VRTSamf/bin/getimf
  ```

  Troubleshoot and fix the issue.

  If the issue cannot be fixed, then contact Symantec technical support with the file that the `getimf` script generates.

Log files

Table 3-1 lists the various log files and their location. The log files contain useful information for identifying issues and resolving them.

Table 3-1 List of log files

Log file	Location	Description
VCS engine log file	`/var/VRTSvcs/log/engine_A.log`	Contains all actions performed by the high availability daemon `had`. **Note:** Verify if there are any CVM or PrivNIC errors logged in this file, since they may prove to be critical errors.

CVM log files	`/var/adm/vx/cmdlog`
	`/var/adm/vx/ddl.log`
	`/var/adm/vx/translog`
	`/var/adm/vx/dmpevents.log`
	`/var/VRTSvcs/log/engine_A.log`
		The cmdlog file contains the list of CVM commands.
		For more information on collecting important CVM logs:
		See "Collecting important CVM logs" on page 115.

VCS agent log files	`/var/VRTSvcs/log/`*`agenttype`*`_A.log`
	where *agenttype* is the type of the VCS agent.
	For example, the log files for the CFS agent can be located at:
	`/var/VRTSvcs/log/CFSMount_A.log`
		Contains messages and errors related to the agent functions.
		For more information, see the *Veritas Storage Foundation Administrator's Guide*.

Table 3-1 List of log files *(continued)*

Log file	Location	Description

OS system log	`/var/log/messages`
	
	You can also run the following command to collect the OS system log:
	
	`# errpt -a`
		Contains messages and errors arising from operating system modules and drivers.

I/O fencing kernel logs	`/var/VRTSvcs/log/vxfen/vxfen.log`
	
	Obtain the logs by running the following command:
	
	`# /opt/VRTSvcs/vxfen/bin/\ vxfendebug -p`
		Contains messages, errors, or diagnostic information for I/O fencing.

Vxfend log	`/var/VRTSvcs/log/vxfen/vxfend_A.log`	Contains messages, errors, or diagnostic information for the vxfend process.

Collecting important CVM logs

You need to stop and restart the cluster to collect detailed CVM TIME_JOIN messages.

To collect detailed CVM TIME_JOIN messages

1. On all the nodes in the cluster, perform the following steps.
   - Edit the `/opt/VRTSvcs/bin/CVMCluster/online` script.
     Insert the '-T' option to the following string.
     Original string: `clust_run=`$VXCLUSTADM -m vcs -t $TRANSPORT startnode 2> $CVM_ERR_FILE`
     Modified string: **`clust_run=`$VXCLUSTADM -m vcs -t $TRANSPORT -T startnode 2> $CVM_ERR_FILE`**
2. Stop the cluster.

   `# hastop -all`
3. Start the cluster.

   `# hastart`

   At this point, CVM TIME_JOIN messages display in the `/var/log/messages`file and on the console.

You can also enable vxconfigd daemon logging as follows:

```
# vxdctl debug 9 /var/adm/vx/vxconfigd_debug.out
```

The debug information that is enabled is accumulated in the system console log and in the text file `/var/adm/vx/vxconfigd_debug.out`. '9' represents the level of debugging. '1' represents minimal debugging. '9' represents verbose output.

Caution: Turning on vxconfigd debugging degrades VxVM performance. Use vxconfigd debugging with discretion in a production environment.

To disable vxconfigd debugging:

```
# vxdctl debug 0
```

The CVM kernel message dump can be collected on a live node as follows:

```
# /etc/vx/diag.d/kmsgdump -d 2000 > \
/var/adm/vx/kmsgdump.out
```

About SF Sybase CE kernel and driver messages

SF Sybase CE drivers such as GAB print messages to the console if the kernel and driver messages are configured to be displayed on the console. Make sure that the kernel and driver messages are logged to the console.

For details on how to configure console messages, see the operating system documentation.

VCS message logging

VCS generates two types of 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:

```
2011/07/10 16:08:09 VCS INFO V-16-1-10077 Received new
cluster membership
```

The agent log is located at `/var/VRTSvcs/log/<agent>.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:

```
2011/07/10 10:38:23 VCS WARNING V-16-20018-301
Sybase:ase:monitor:Open for dataserver 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 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.

The default value for variable $VCS_DIAG is `/var/VRTSvcs/`.

If the debug logging is not turned on, these FFDC logs are useful to analyze the issues that require professional support.

GAB message logging

If GAB encounters some problem, then First Failure Data Capture (FFDC) logs are also generated and dumped.

When you have configured GAB, GAB also starts a GAB logging daemon (`/opt/VRTSgab/gablogd`). GAB logging daemon is enabled by default. You can change the value of the GAB tunable parameter `gab_ibuf_count` to disable the GAB logging daemon.

See "About GAB load-time or static tunable parameters" on page 148.

This GAB logging daemon collects the GAB related logs when a critical events such as an iofence or failure of the master of any GAB port occur, and stores the data in a compact binary form. You can use the `gabread_ffdc` utility as follows to read the GAB binary log files:

```
/opt/VRTSgab/gabread_ffdc-kernel_version  binary_logs_files_location
```

You can change the values of the following environment variables that control the GAB binary log files:

- GAB_FFDC_MAX_INDX: Defines the maximum number of GAB binary log files
  The GAB logging daemon collects the defined number of log files each of eight MB size. The default value is 20, and the files are named `gablog.1` through `gablog.20`. At any point in time, the most recent file is the `gablog.1` file.
- GAB_FFDC_LOGDIR: Defines the log directory location for GAB binary log files
  The default location is:
  `/var/log/gab_ffdc`
  Note that the gablog daemon writes its log to the `glgd_A.log` and `glgd_B.log` files in the same directory.

You can either define these variables in the following GAB startup file or use the `export` command. You must restart GAB for the changes to take effect.

```
/etc/sysconfig/gab
```

About debug log tags usage

The following table illustrates the use of debug tags:

Entity	Debug logs used
Agent functions	DBG_1 to DBG_21
Agent framework	DBG_AGTRACE
	DBG_AGDEBUG
	DBG_AGINFO
Icmp agent	DBG_HBFW_TRACE
	DBG_HBFW_DEBUG
	DBG_HBFW_INFO

Entity	Debug logs used
HAD	DBG_AGENT (for agent-related debug logs)
	DBG_ALERTS (for alert debug logs)
	DBG_CTEAM (for GCO debug logs)
	DBG_GAB, DBG_GABIO (for GAB debug messages)
	DBG_GC (for displaying global counter with each log message)
	DBG_INTERNAL (for internal messages)
	DBG_IPM (for Inter Process Messaging)
	DBG_JOIN (for Join logic)
	DBG_LIC (for licensing-related messages)
	DBG_NTEVENT (for NT Event logs)
	DBG_POLICY (for engine policy)
	DBG_RSM (for RSM debug messages)
	DBG_TRACE (for trace messages)
	DBG_SECURITY (for security-related messages)
	DBG_LOCK (for debugging lock primitives)
	DBG_THREAD (for debugging thread primitives)
	DBG_HOSTMON (for HostMonitor debug logs)

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:

```
# haconf -makerw
```

2. Enable logging and set the desired log levels. The following example depicts the command for the IPMultiNIC resource type.

```
# hatype -modify IPMultiNIC LogDbg DBG_1 DBG_2 DBG_4 DBG_21
```

   See the description of the LogDbg attribute for more information.

3 For script-based 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
```

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.

Enabling debug logs for the VCS engine

You can enable debug logs for the VCS engine, VCS agents, and HA commands in two ways:

- To enable debug logs at run-time, use the `halog -addtags` command.
- To enable debug logs at startup, use the VCS_DEBUG_LOG_TAGS environment variable. You must set the VCS_DEBUG_LOG_TAGS before you start HAD or before you run HA commands.
  Examples:

```
# export VCS_DEBUG_LOG_TAGS="DBG_TRACE DBG_POLICY"
# hastart

# export VCS_DEBUG_LOG_TAGS="DBG_AGINFO DBG_AGDEBUG DBG_AGTRACE"
# hastart

# export VCS_DEBUG_LOG_TAGS="DBG_IPM"
# hagrp -list
```

Note: Debug log messages are verbose. If you enable debug logs, log files might fill up quickly.

Enabling debug logs for IMF

Run the following commands to enable additional debug logs for Intelligent Monitoring Framework (IMF). The messages get logged in the agent-specific log file `/var/VRTSvcs/log/agentname_A.log`.

To enable additional debug logs

1 For Process, Mount, and Application agents:

```
# hatype -modify agentname LogDbg
DBG_AGDEBUG DBG_AGTRACE DBG_AGINFO DBG_1 DBG_2
DBG_3 DBG_4 DBG_5 DBG_6 DBG_7
```

2 For Sybase agents:

```
# hatype -modify agentname LogDbg
DBG_AGDEBUG DBG_AGTRACE DBG_AGINFO DBG_1 DBG_2
DBG_3 DBG_4 DBG_5 DBG_6 DBG_7
DBG_8 DBG_9 DBG_10
```

3 For CFSMount agent:

```
# hatype -modify agentname LogDbg
DBG_AGDEBUG DBG_AGTRACE DBG_AGINFO DBG_1 DBG_2
DBG_3 DBG_4 DBG_5 DBG_6 DBG_7
DBG_8 DBG_9 DBG_10 DBG_11 DBG_12
DBG_13 DBG_14 DBG_15 DBG_16
DBG_17 DBG_18 DBG_19 DBG_20 DBG_21
```

4 For CVMvxconfigd agent, you do not have to enable any additional debug logs.

5 For AMF driver in-memory trace buffer:

```
# amfconfig -S errlevel all all
```

If you had enabled AMF driver in-memory trace buffer, you can view the additional logs using the `amfconfig -p dbglog` command.

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:

gcoconfig.bmc	gcoconfig messages
VRTSvcsHbfw.bmc	Heartbeat framework messages
VRTSvcsTriggers.bmc	VCS trigger messages
VRTSvcsWac.bmc	Wide-area connector process messages
vxfen*.bmc	Fencing messages
gab.bmc	GAB command-line interface messages
hagetcf.bmc	hagetcf messages
llt.bmc	LLT command-line interface messages
VRTSvcsAgfw.bmc	Agent framework messages
VRTSvcsAlerts.bmc	VCS alert messages
VRTSvcsApi.bmc	VCS API messages
VRTSvcsCommon.bmc	Common modules messages
VRTSvcsHad.bmc	VCS engine (HAD) messages
VRTSvcs*platform*Agent.bmc	VCS bundled agent messages
VRTSvcs*platformagent_name*.bmc	VCS enterprise agent messages

Troubleshooting tips

The following files and command output may be required to determine the source of a problem:

- Sybase installation error log
- Veritas log files
- OS system log
- GAB port membership

Sybase installation error log

This file contains errors that occurred during installation. It clarifies the nature of the error and exactly when it occurred during the installation.

To check the Sybase installation error log

1 Access the following file:

```
$SYBASE/$SYBASE_ASE/install/dataserver_name*.log
```

2 Verify if there are any installation errors logged in this file, since they may prove to be critical errors.

3 If there are any installation problems, send this file to Tech Support. It is required for debugging the issue.

Veritas log files

The Veritas log file contains all actions performed by HAD.

To check the Veritas log files

1 Access the following file:

```
/var/VRTSvcs/log/engine_A.log
```

2 Verify if there are any CVM errors logged in this file, since they may prove to be critical errors.

3 Check the vxconfigd log file at:

```
/var/adm/vx/vxconfigd.log
```

There are additional log files pertaining to the agents for SF Sybase CE components such as CVM and CFS in the /var/VRTSvcs/log directory, which are also helpful in diagnosing issues.

To check the agent log files for CVM:

```
# /var/VRTSvcs/log/CVMVolDg_A.log
```

To check the agent log files for CFS:

```
# /var/VRTSvcs/log/CFSMount_A.log
```

To check the agent log files for Sybase:

```
# /var/VRTSvcs/log/Sybase_A.log
```

OS system log

OS syslog files can provide valuable information for diagnosing problems. The system log can be checked in the following file:

```
/var/log/messages
```

GAB port membership

To check GAB port membership

Enter the following command:

```
# /sbin/gabconfig -a
```

Port b must exist on the local system.

The output resembles this information:

```
GAB Port Memberships
===============================================================
Port a gen 4a1c0001 membership 01
Port b gen ada40d01 membership 01
Port f gen f1990002 membership 01
Port h gen d8850002 membership 01
Port u gen de4f0203 membership 01
Port v gen 1fc60002 membership 01
Port w gen 15ba0002 membership 01
Port y gen 73f449   membership 01
```

Table 3-2 defines each GAB port's function.

For illustration of different GAB ports,

Table 3-2 GAB port function

Port	Function
a	This port is used for GAB internally.
b	This port is used for I/O fencing communications.
f	CFS uses this port for GLM lock and metadata communication.
h	VCS uses this port. VCS communicates the status of resources running on each system to all systems in the cluster.
u	CVM uses this port to ship commands from slave node to master node.
v	CVM uses this port for kernel-to-kernel communication.

Table 3-2 GAB port function *(continued)*

Port	Function
w	`vxconfigd` configuration daemon (module for CVM) uses this port for messaging.
y	CVM uses this port for I/O shipping.

What to do if you see a licensing reminder

In this release, you can install without a license key. In order to comply with the End User License Agreement, you must either install a license key or make the host managed by a Management Server. If you do not comply with these terms within 60 days, the following warning messages result:

```
WARNING V-365-1-1 This host is not entitled to run Veritas Storage
Foundation/Veritas Cluster Server.As set forth in the End User
License Agreement (EULA) you must complete one of the two options
set forth below. To comply with this condition of the EULA and
stop logging of this message, you have <nn> days to either:
- make this host managed by a Management Server (see
  http://go.symantec.com/sfhakeyless for details and free download),
  or
- add a valid license key matching the functionality in use on this host
  using the command 'vxlicinst'
```

To comply with the terms of the EULA, and remove these messages, you must do one of the following within 60 days:

- Install a valid license key corresponding to the functionality in use on the host. After you install the license key, you must validate the license key using the following command:

  ```
  # /opt/VRTS/bin/vxkeyless
  ```

- Continue with keyless licensing by managing the server or cluster with a management server.
  For more information about keyless licensing, see the following URL: http://go.symantec.com/sfhakeyless

Restarting the installer after a failed connection

If an installation is killed because of a failed connection, you can restart the installer to resume the installation. The installer detects the existing installation. The installer prompts you whether you want to resume the installation. If you resume the installation, the installation proceeds from the point where the installation failed.

Installer cannot create UUID for the cluster

The installer displays the following error message if the installer cannot find the uuidconfig.pl script before it configures the UUID for the cluster:

```
Couldn't find uuidconfig.pl for uuid configuration,
please create uuid manually before start vcs
```

You may see the error message during SF Sybase CE configuration, upgrade, or when you add a node to the cluster using the installer.

Workaround: To start SF Sybase CE, you must run the uuidconfig.pl script manually to configure the UUID on each cluster node.

To configure the cluster UUID when you create a cluster manually

- On one node in the cluster, perform the following command to populate the cluster UUID on each node in the cluster.

  ```
  # /opt/VRTSvcs/bin/uuidconfig.pl -clus -configure nodeA
  nodeB ... nodeN
  ```

  Where nodeA, nodeB, through nodeN are the names of the cluster nodes.

Troubleshooting I/O fencing

The following sections discuss troubleshooting the I/O fencing problems. Review the symptoms and recommended solutions.

The vxfentsthdw utility fails when SCSI TEST UNIT READY command fails

While running the vxfentsthdw utility, you may see a message that resembles as follows:

```
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 the Hitachi Data Systems 99XX arrays if bit 186 of the system mode option is not enabled.

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:

```
...VxFEN ERROR V-11-1-25 ... Unable to join running cluster
since cluster is currently fencing
a node out of the cluster.
```

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.

To manually join the node to the cluster when I/O fencing attempts fail

- If the vxfen script fails in the attempts to allow the node to join the cluster, restart vxfen driver with the command:

  ```
  # /etc/init.d/vxfen stop
  # /etc/init.d/vxfen start
  ```

  If the command fails, restart the new node.

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.
```

A node experiences the split-brain condition when it loses the heartbeat with its peer nodes due to failure of all private interconnects or node hang. Review the behavior of I/O fencing under different scenarios and the corrective measures to be taken.

See “How I/O fencing works in different event scenarios” on page 31.

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 system1:

```
system1> # lltstat -C
57069

system1> # cat /etc/vxfentab
/dev/vx/rdmp/disk_7
/dev/vx/rdmp/disk_8
/dev/vx/rdmp/disk_9

system1> # 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]:  86,70,48,49,52,66,48,48
 [Character Format]: VFBEAD00
 [Node Format]: Cluster ID: 48813 Node ID: 0 Node Name: unknown
```

Where *disk_7*, *disk_8*, and *disk_9* represent the disk names in your setup.

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 vxfenclearpre command before you use them as coordinator disks in the local cluster.

See "About the vxfenclearpre utility" on page 78.

Fencing startup reports preexisting split-brain

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 coordination points.

See “Clearing preexisting split-brain condition” on page 129.

Clearing preexisting split-brain condition

Review the information on how the VxFEN driver checks for preexisting split-brain condition.

See “Fencing startup reports preexisting split-brain” on page 129.

Table 3-3 describes how to resolve a preexisting split-brain condition depending on the scenario you have encountered:

Table 3-3 Recommened solution to clear pre-existing split-brain condition

Scenario	Solution
Actual potential split-brain condition—system 2 is up and system 1 is ejected	**1** Determine if system1 is up or not. **2** If system 1 is up and running, shut it down and repair the private network links to remove the split-brain condition. **3** Restart system 1.
Apparent potential split-brain condition—system 2 is down and system 1 is ejected	**1** Physically verify that system 2 is down. Verify the systems currently registered with the coordination points. Use the following command for coordinator disks: `# vxfenadm -s all -f /etc/vxfentab` The output of this command identifies the keys registered with the coordinator disks. **2** Clear the keys on the coordinator disks as well as the data disks in all shared disk groups using the `vxfenclearpre` command. The command removes SCSI-3 registrations and reservations. See "About the vxfenclearpre utility" on page 78. **3** Make any necessary repairs to system 2. **4** Restart system 2.

Table 3-3 Recommened solution to clear pre-existing split-brain condition *(continued)*

Scenario	Solution

Apparent potential split-brain condition—system 2 is down and system 1 is ejected	**1** Physically verify that system 2 is down.
	Verify the systems currently registered with the coordination points.
	Use the following command for CP servers:
	`# 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 VCS name for the SF Sybase CE cluster (application cluster).
	The command lists the systems registered with the CP server.
	2 Clear the keys on the CP servers using the `cpsadm` command. The `cpsadm` command clears a registration on a CP server:
	`# cpsadm -s cp_server -a unreg_node -c cluster_name -n nodeid`
	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 , and *nodeid* specifies the node id of node. Ensure that fencing is not already running on a node before clearing its registration on the CP server.
	After removing all stale registrations, the joiner node will be able to join the cluster.
	3 Make any necessary repairs to system 2.
	4 Restart system 2.

Registered keys are lost on the coordinator disks

If the coordinator disks lose the keys that are registered, the cluster might panic when a cluster reconfiguration occurs.

To refresh the missing keys

- 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 89.

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.

In a cluster that is online, you can replace the disks using the vxfenswap utility.

See "About the vxfenswap utility" on page 81.

Review the following information to replace coordinator disk in the coordinator disk group, or to destroy a coordinator disk group.

Note the following about the procedure:

- When you add a disk, add the disk to the disk group vxfencoorddg 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 replace a disk in the coordinator disk group when the cluster is offline

1. Log in as superuser on one of the cluster nodes.
2. If VCS is running, shut it down:

   ```
   # hastop -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 each node:

   ```
   # /etc/init.d/vxfen 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, vxfencoorddg) 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, vxdiskadm.

You may also destroy the existing coordinator disk group. For example:

- Verify whether the coordinator attribute is set to on.

```
# vxdg list vxfencoorddg |  grep flags: | grep coordinator
```

- Destroy the coordinator disk group.

```
# vxdg -o coordinator destroy vxfencoorddg
```

6 Add the new disk to the node and initialize it as a VxVM disk.

Then, add the new disk to the vxfencoorddg disk group:

- If you destroyed the disk group in step 5, then create the disk group again and add the new disk to it.
- If the diskgroup already exists, then add the new disk to it.

```
# vxdg -g vxfencoorddg -o coordinator adddisk disk_name
```

7 Test the recreated disk group for SCSI-3 persistent reservations compliance.

See "Testing the coordinator disk group using vxfentsthdw -c option" on page 68.

8 After replacing disks in a coordinator disk group, deport the disk group:

```
# vxdg deport `cat /etc/vxfendg`
```

9 On each node, start the I/O fencing driver:

```
# /etc/init.d/vxfen 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.

```
# vxfenadm -d
```

Make sure that I/O fencing mode is not disabled in the output.

11 If necessary, restart VCS on each node:

```
# hastart
```

Troubleshooting Cluster Volume Manager

This section discusses troubleshooting CVM problems.

Restoring communication between host and disks after cable disconnection

If a fiber cable is inadvertently disconnected between the host and a disk, you can restore communication between the host and the disk without restarting.

To restore lost cable communication between host and disk

1 Reconnect the cable.

2 On all nodes, use the `fdisk -l` command to scan for new disks.

It may take a few minutes before the host is capable of seeing the disk.

3 On all nodes, issue the following command to rescan the disks:

```
# vxdisk scandisks
```

4 On the master node, reattach the disks to the disk group they were in and retain the same media name:

```
# vxreattach
```

This may take some time. For more details, see `vxreattach` (1M) manual page.

Shared disk group cannot be imported

If you see a message resembling:

```
vxvm:vxconfigd:ERROR:vold_pgr_register(/dev/vx/rdmp/disk_name):
local_node_id<0
Please make sure that CVM and vxfen are configured
and operating correctly
```

First, make sure that CVM is running. You can see the CVM nodes in the cluster by running the vxclustadm nidmap command.

```
# vxclustadm nidmap
Name          CVM Nid     CM Nid      State
system1         1           0           Joined: Master
system2         0           1           Joined: Slave
```

This above ouput shows that CVM is healthy, with system system1 as the CVM master. If CVM is functioning correctly, then the output above is displayed when CVM cannot retrieve the node ID of the local system from the `vxfen` driver. This usually happens when port b is not configured.

To verify vxfen driver is configured

◆ Check the GAB ports with the command:

```
# gabconfig -a
```

Port b must exist on the local system.

Error importing shared disk groups

The following message may appear when importing shared disk group:

```
VxVM vxdg ERROR V-5-1-587 Disk group disk group name: import
failed: No valid disk found containing disk group
```

You may need to remove keys written to the disk.

For information about removing keys written to the disk:

See "Removing preexisting keys" on page 78.

Unable to start CVM

If you cannot start CVM, check the consistency between the `/etc/llthosts` and `main.cf` files for node IDs.

You may need to remove keys written to the disk.

For information about removing keys written to the disk:

See "Removing preexisting keys" on page 78.

CVM group is not online after adding a node to the cluster

The possible causes for the CVM group being offline after adding a node to the cluster are as follows:

- Other resources configured in the cvm group as critical resources are not online.

To resolve the issue if other resources in the group are not online

1. Log onto one of the nodes in the existing cluster as the root user.
2. Bring the resource online:

```
# hares -online resource_name -sys system_name
```

3. Verify the status of the resource:

```
# hastatus -resource  resource_name
```

4. If the resource is not online, configure the resource as a non-critical resource:

```
# haconf -makerw
# hares -modify resource_name Critical 0
# haconf -dump -makero
```

CVMVolDg not online even though CVMCluster is online

When the CVMCluster resource goes online, then all shared disk groups that have the auto-import flag set are automatically imported. If the disk group import fails for some reason, the CVMVolDg resources fault. Clearing and taking the CVMVolDg type resources offline does not resolve the problem.

To resolve the resource issue

1. Fix the problem causing the import of the shared disk group to fail.
2. Offline the cvm group containing the resource of type CVMVolDg as well as the service group containing the CVMCluster resource type.
3. Bring the cvm group containing the CVMCluster resource online.
4. Bring the cvm group containing the CVMVolDg resource online.

Shared disks not visible

If the shared disks in `/dev` are not visible, perform the following tasks:

Make sure that all shared LUNs are discovered by the HBA and SCSI layer. This can be verified by running the `ls -ltr` command on any of the disks under `/dev/*`.

For example:

```
# ls -ltr /dev/disk_name
```

If all LUNs are not discovered by SCSI, the problem might be corrected by specifying `dev_flags` or `default_dev_flags` and *max_luns* parameters for the SCSI driver.

If the LUNs are not visible in `/dev/*` files, it may indicate a problem with SAN configuration or zoning.

Troubleshooting interconnects

This section discusses troubleshooting interconnect problems.

Network interfaces change their names after reboot

On SUSE systems, network interfaces change their names after reboot even with `HOTPLUG_PCI_QUEUE_NIC_EVENTS=yes` and `MANDATORY_DEVICES="..."` set.

Workaround: Use `PERSISTENT_NAME= ethX` where X is the interface number for all interfaces.

Example entries for mandatory devices

If you are using eth2 and eth3 for interconnectivity, use the following procedure examples to set mandatory devices.

To set mandatory devices entry in the /etc/sysconfig/network/config

Enter:

```
MANDATORY_DEVICES="eth2-00:04:23:AD:4A:4C
eth3-00:04:23:AD:4A:4D"
```

To set a persistent name entry in an interface file

In file: /etc/sysconfig/network/ifcfg-eth-id-00:09:3d:00:cd:22 (Name of the eth0 Interface file), enter:

```
BOOTPROTO='static'
BROADCAST='10.212.255.255'
IPADDR='10.212.88.22'
```

```
MTU=' '
NETMASK='255.255.254.0'
NETWORK='10.212.88.0'
REMOTE_IP=' '
STARTMODE='onboot'
UNIQUE='RFE1.bBSepP2NetB'
_nm_name='bus-pci-0000:06:07.0'
PERSISTENT_NAME=eth0
```

Troubleshooting Sybase ASE CE

This section discusses troubleshooting Syabase ASE CE.

Sybase private networks

Sybase private networks should be on LLT links.

Sybase instances under VCS control

Sybase instances should be configured under VCS control.

Node does not reboot

Cluster membership mode should be set to "vcs".

Problem: a node does not reboot after a dataserver is killed.

Resolution: check if membership-mode is set to vcs in qrmutil.

```
# qrmutil
--quorum-dev=/quorum/quorum.data --display=config |grep mode
```

Sybase instance not starting

Problem: a Sybase instance that is not starting and is stuck in "VCMP is waiting for vxfend message."

Resolution: restart vxfend:

```
# hares -online vxfend -sys system1
```

Chapter 4

Prevention and recovery strategies

This chapter includes the following topics:

- Prevention and recovery strategies

Prevention and recovery strategies

The following topics are useful diagnostic tools and strategies for preventing and recovering from the various problems that can occur in the SF Sybase CE environment.

Verification of GAB ports in SF Sybase CE cluster

The following ports need to be up on all the nodes of SF Sybase CE cluster:

- port a (GAB)
- port b (I/O fencing)
- port f (CFS)
- port h (VCS)
- port v (CVM kernel messaging)
- port w (CVM vxconfigd)
- port u (CVM - to ship commands from slave node to master node)
- port y (CVM - I/O shipping)

The following command can be used to verify the state of GAB ports:

```
# gabconfig -a
```

GAB Port Memberships

```
Port a gen 7e6e7e05 membership 01
Port b gen 58039502 membership 01
Port f gen 1ea84702 membership 01
Port h gen cf430b02 membership 01
Port u gen de4f0203 membership 01
Port v gen db411702 membership 01
Port w gen cf430b02 membership 01
Port y gen 73f449   membership 01
```

The data indicates that all the GAB ports are up on the cluster having nodes 0 and 1.

For more information on the GAB ports in SF Sybase CE cluster, see the *Veritas Storage Foundation for Sybase ASE CE Installation and Configuration Guide.*

Examining GAB seed membership

The number of systems that participate in the cluster is specified as an argument to the `gabconfig` command in /etc/gabtab. In the following example, two nodes are expected to form a cluster:

```
# cat /etc/gabtab
```

```
/sbin/gabconfig -c -n2
```

GAB waits until the specified number of nodes becomes available to automatically create the port “a” membership. Port “a” indicates GAB membership for an SF Sybase CE cluster node. Every GAB reconfiguration, such as a node joining or leaving increments or decrements this seed membership in every cluster member node.

A sample port ‘a’ membership as seen in `gabconfig -a` is shown:

```
Port a gen 7e6e7e01 membership 01
```

In this case, 7e6e7e01 indicates the “membership generation number” and 01 corresponds to the cluster’s “node map”. All nodes present in the node map reflects the same membership ID as seen by the following command:

```
# gabconfig -a | grep "Port a"
```

The semi-colon is used as a placeholder for a node that has left the cluster. In the following example, node 0 has left the cluster:

```
# gabconfig -a | grep "Port a"
```

```
Port a gen 7e6e7e04 membership ;1
```

When the last node exits the port “a” membership, there are no other nodes to increment the membership ID. Thus the port “a” membership ceases to exist on any node in the cluster.

When the last and the final system is brought back up from a complete cluster cold shutdown state, the cluster will seed automatically and form port “a” membership on all systems. Systems can then be brought down and restarted in any combination so long as at least one node remains active at any given time.

The fact that all nodes share the same membership ID and node map certifies that all nodes in the node map participates in the same port “a” membership. This consistency check is used to detect “split-brain” and “pre-existing split-brain” scenarios.

Split-brain occurs when a running cluster is segregated into two or more partitions that have no knowledge of the other partitions. The pre-existing network partition is detected when the "cold" nodes (not previously participating in cluster) start and are allowed to form a membership that might not include all nodes (multiple sub-clusters), thus resulting in a potential split-brain.

Note: Symantec I/O fencing prevents data corruption resulting from any split-brain scenarios.

Manual GAB membership seeding

It is possible that one of the nodes does not come up when all the nodes in the cluster are restarted, due to the "minimum seed requirement" safety that is enforced by GAB. Human intervention is needed to safely determine that the other node is in fact not participating in its own mini-cluster.

The following should be carefully validated before manual seeding, to prevent introducing split-brain and subsequent data corruption:

- Verify that none of the other nodes in the cluster have a port “a” membership
- Verify that none of the other nodes have any shared disk groups imported
- Determine why any node that is still running does not have a port “a” membership

Run the following command to manually seed GAB membership:

```
# gabconfig -cx
```

Refer to gabconfig (1M) for more details.

Evaluating VCS I/O fencing ports

I/O Fencing (VxFEN) uses a dedicated port that GAB provides for communication across nodes in the cluster. You can see this port as port 'b' when `gabconfig -a` runs on any node in the cluster. The entry corresponding to port 'b' in this membership indicates the existing members in the cluster as viewed by I/O Fencing.

GAB uses port "a" for maintaining the cluster membership and must be active for I/O Fencing to start.

To check whether fencing is enabled in a cluster, the '-d' option can be used with vxfenadm (1M) to display the I/O Fencing mode on each cluster node. Port "b" membership should be present in the output of `gabconfig -a` and the output should list all the nodes in the cluster.

If the GAB ports that are needed for I/O fencing are not up, that is, if port "a" is not visible in the output of `gabconfig -a` command, LLT and GAB must be started on the node.

The following commands can be used to start LLT and GAB respectively:

To start LLT on each node:

```
# /etc/init.d/llt start
```

If LLT is configured correctly on each node, the console output displays:

```
LLT INFO V-14-1-10009 LLT Protocol available
```

On a two node cluster, for example system1 and system2, checks you can run to make sure LLT is properly configured:

```
# gabconfig -a |grep 'Port a'
Port a gen   614605 membership 01

# cat /etc/llthosts
0 system1
1 system2
```

Check the llttab on both nodes:

```
# cat /etc/llttab
set-node system1
set-cluster Cluster id
link eth1 eth-00:15:17:48:b4:98 - ether - -
link eth2 eth-00:15:17:48:b4:99 - ether - -
```

To start GAB, on each node:

```
# /etc/init.d/gab start
```

If GAB is configured correctly on each node, the console output displays:

```
GAB INFO V-15-1-20021 GAB available
GAB INFO V-15-1-20026 Port a registration waiting for seed port
membership
```

Check to make sure that GAB is properly configured:

```
# gabconfig -a |grep 'Port b'
Port b gen   614604 membership 01
# cat /etc/gabtab
/sbin/gabconfig -c -n2 <here it 2 as number of nodes are 2)
```

Verifying normal functioning of VCS I/O fencing

It is mandatory to have VCS I/O fencing enabled in SF Sybase CE cluster to protect against split-brain scenarios. VCS I/O fencing can be assumed to be running normally in the following cases:

- Fencing port 'b' enabled on the nodes

  ```
  # gabconfig -a
  ```

  To verify that fencing is enabled on the nodes:

  ```
  # vxfenadm -d
  ```

- Registered keys present on the coordinator disks

  ```
  # vxfenadm -g all -f /etc/vxfentab
  ```

Managing SCSI-3 PR keys in SF Sybase CE cluster

I/O Fencing places the SCSI-3 PR keys on coordinator LUNs. The format of the key follows the naming convention wherein ASCII “A” is prefixed to the LLT ID of the system that is followed by 7 dash characters.

For example:

node 0 uses A-------

node 1 uses B-------

In an SF Sybase CE/SF CFS/SF HA environment, VxVM/CVM registers the keys on data disks, the format of which is ASCII “A” prefixed to the LLT ID of the system

followed by the characters “PGRxxxx” where ‘xxxx’ = i such that the disk group is the ith shared group to be imported.

For example: node 0 uses APGR0001 (for the first imported shared group).

In addition to the registration keys, VCS/CVM also installs a reservation key on the data LUN. There is one reservation key per cluster as only one node can reserve the LUN.

See “About SCSI-3 Persistent Reservations” on page 30.

The following command lists the keys on a data disk group:

```
# vxdg list |grep data
```

```
sybdata_101 enabled,shared,cds 1201715530.28.pushover
```

Select the data disk belonging to sybdata_101:

```
# vxdisk -o alldgs list |grep sybdata_101
```

```
sdh auto:cdsdisk    sdh       sybdata_101  online shared
sdj auto:cdsdisk    sdj       sybdata_101  online shared
sdk auto:cdsdisk    sdk       sybdata_101  online shared
```

The following command lists the PR keys:

```
# vxdisk -o listreserve list sdh
```

.....

.....

Alternatively, the PR keys can be listed using `vxfenadm` command:

```
#echo "/dev/vx/dmp/sdh" > /tmp/disk71
```

```
# vxfenadm -g all -f /tmp/disk71
```

```
Device Name: /dev/vx/dmp/sdh
Total Number Of Keys: 2
key[0]:
        Key Value [Numeric Format]:  66,80,71,82,48,48,48,52
        Key Value [Character Format]: BPGR0004
key[1]:
        Key Value [Numeric Format]:  65,80,71,82,48,48,48,52
        Key Value [Character Format]: APGR0004
```

Evaluating the number of SCSI-3 PR keys on a coordinator LUN, if there are multiple paths to the LUN from the hosts

The utility vxfenadm (1M) can be used to display the keys on the coordinator LUN. The key value identifies the node that corresponds to each key. Each node installs a registration key on all the available paths to the LUN. Thus, the total number of registration keys is the sum of the keys that are installed by each node in the above manner.

See "About the vxfenadm utility" on page 73.

Detecting accidental SCSI-3 PR key removal from coordinator LUNs

The keys currently installed on the coordinator disks can be read using the following command:

```
# vxfenadm -s all -f /etc/vxfentab
```

There should be a key for each node in the operating cluster on each of the coordinator disks for normal cluster operation. There will be two keys for every node if you have a two-path DMP configuration.

Identifying a faulty coordinator LUN

The utility vxfentsthdw (1M) provided with I/O fencing can be used to identify faulty coordinator LUNs. This utility must be run from any node in the cluster. The coordinator LUN, which needs to be checked, should be supplied to the utility.

See "About the vxfentsthdw utility" on page 65.

Starting shared volumes manually

Following a manual CVM shared disk group import, the volumes in the disk group need to be started manually, as follows:

```
# vxvol -g dg_name startall
```

To verify that the volumes are started, run the following command:

```
# vxprint -htrg dg_name | grep ^v
```

Listing all the CVM shared disks

You can use the following command to list all the CVM shared disks:

```
# vxdisk -o alldgs list |grep shared
```

I/O Fencing kernel logs

I/O Fencing kernel logs contain useful information to troubleshoot intricate I/O fencing issues. The logs can be collected using the following command:

```
# /opt/VRTSvcs/vxfen/bin/vxfendebug -p
```

Chapter 5

Tunable parameters

This chapter includes the following topics:

- About SF Sybase CE tunable parameters
- About GAB tunable parameters
- About LLT tunable parameters
- About VXFEN tunable parameters

About SF Sybase CE tunable parameters

Tunable parameters can be configured to enhance the performance of specific SF Sybase CE features. This chapter discusses how to configure the following SF Sybase CE tunables:

- GAB
- LLT
- VXFEN

Symantec recommends that you do not change the tunable kernel parameters without assistance from Symantec support personnel. Several of the tunable parameters preallocate memory for critical data structures, and a change in their values could increase memory use or degrade performance.

About GAB tunable parameters

GAB provides various configuration and tunable parameters to modify and control the behavior of the GAB module.

These tunable parameters not only provide control of the configurations like maximum possible number of nodes in the cluster, but also provide control on

how GAB behaves when it encounters a fault or a failure condition. Some of these tunable parameters are needed when the GAB module is loaded into the system. Any changes to these load-time tunable parameters require either unload followed by reload of GAB module or system reboot. Other tunable parameters (run-time) can be changed while GAB module is loaded, configured, and cluster is running. Any changes to such a tunable parameter will have immediate effect on the tunable parameter values and GAB behavior.

These tunable parameters are defined in the following file:

`/etc/sysconfig/gab`

See "About GAB load-time or static tunable parameters" on page 148.

See "About GAB run-time or dynamic tunable parameters" on page 149.

About GAB load-time or static tunable parameters

Table 5-1 lists the static tunable parameters in GAB that are used during module load time. Use the `gabconfig -e` command to list all such GAB tunable parameters.

You can modify these tunable parameters only by adding new values in the GAB configuration file. The changes take effect only on reboot or on reload of the GAB module.

Table 5-1 GAB static tunable parameters

GAB parameter	Description	Values (default and range)
gab_numnids	Maximum number of nodes in the cluster	Default: 64 Range: 1-64
gab_numports	Maximum number of ports in the cluster	Default: 32 Range: 1-32
gab_flowctrl	Number of pending messages in GAB queues (send or receive) before GAB hits flow control. This can be overwritten while cluster is up and running with the `gabconfig -Q` option. Use the `gabconfig` command to control value of this tunable.	Default: 128 Range: 1-1024
gab_logbufsize	GAB internal log buffer size in bytes	Default: 48100 Range: 8100-65400

Table 5-1 GAB static tunable parameters *(continued)*

GAB parameter	Description	Values (default and range)
gab_msglogsize	Maximum messages in internal message log	Default: 256 Range: 128-4096
gab_isolate_time	Maximum time to wait for isolated client Can be overridden at runtime See "About GAB run-time or dynamic tunable parameters" on page 149.	Default: 120000 msec (2 minutes) Range: 160000-240000 (in msec)
gab_kill_ntries	Number of times to attempt to kill client Can be overridden at runtime See "About GAB run-time or dynamic tunable parameters" on page 149.	Default: 5 Range: 3-10
gab_conn_wait	Maximum number of wait periods (as defined in the stable timeout parameter) before GAB disconnects the node from the cluster during cluster reconfiguration	Default: 12 Range: 1-256
gab_ibuf_count	Determines whether the GAB logging daemon is enabled or disabled The GAB logging daemon is enabled by default. To disable, change the value of gab_ibuf_count to 0. This can be overridden while cluster is up and running with the `gabconfig -K` option. Use the `gabconfig` command to control value of this tunable.	Default: 8 Range: 0-32
gab_kstat_size	Number of system statistics to maintain in GAB	Default: 60 Range: 0 - 240

About GAB run-time or dynamic tunable parameters

You can change the GAB dynamic tunable parameters while GAB is configured and while the cluster is running. The changes take effect immediately on running

the `gabconfig` command. Note that some of these parameters also control how GAB behaves when it encounters a fault or a failure condition. Some of these conditions can trigger a PANIC which is aimed at preventing data corruption.

You can display the default values using the `gabconfig -l` command. To make changes to these values persistent across reboots, you can append the appropriate command options to the `/etc/gabtab` file along with any existing options. For example, you can add the `-k` option to an existing `/etc/gabtab` file that might read as follows:

```
gabconfig -c -n4
```

After adding the option, the `/etc/gabtab` file looks similar to the following:

```
gabconfig -c -n4 -k
```

Table 5-2 describes the GAB dynamic tunable parameters as seen with the `gabconfig -l` command, and specifies the command to modify them.

Table 5-2 GAB dynamic tunable parameters

GAB parameter	Description and command

Control port seed	This option defines the minimum number of nodes that can form the cluster. This option controls the forming of the cluster. If the number of nodes in the cluster is less than the number specified in the `gabtab` file, then the cluster will not form. For example: if you type `gabconfig -c -n4`, then the cluster will not form until all four nodes join the cluster. If this option is enabled using the `gabconfig -x` command then the node will join the cluster even if the other nodes in the cluster are not yet part of the membership.
	
	Use the following command to set the number of nodes that can form the cluster:
	
	`gabconfig -n count`
	
	Use the following command to enable control port seed. Node can form the cluster without waiting for other nodes for membership:
	
	`gabconfig -x`

Table 5-2 GAB dynamic tunable parameters *(continued)*

GAB parameter	Description and command

Halt on process death	Default: Disabled
	This option controls GAB's ability to halt (panic) the system on user process death. If _had and _hashadow are killed using `kill -9`, the system can potentially lose high availability. If you enable this option, then the GAB will PANIC the system on detecting the death of the client process. The default behavior is to disable this option.
	Use the following command to enable halt system on process death:
	`gabconfig -p`
	Use the following command to disable halt system on process death:
	`gabconfig -P`

Missed heartbeat halt	Default: Disabled
	If this option is enabled then the system will panic on missing the first heartbeat from the VCS engine or the `vxconfigd` daemon in a CVM environment. The default option is to disable the immediate panic.
	This GAB option controls whether GAB can panic the node or not when the VCS engine or the vxconfigd daemon miss to heartbeat with GAB. If the VCS engine experiences a hang and is unable to heartbeat with GAB, then GAB will NOT PANIC the system immediately. GAB will first try to abort the process by sending SIGABRT (kill_ntries - default value 5 times) times after an interval of "iofence_timeout" (default value 15 seconds). If this fails, then GAB will wait for the "isolate timeout" period which is controlled by a global tunable called isolate_time (default value 2 minutes). If the process is still alive, then GAB will PANIC the system.
	If this option is enabled GAB will immediately HALT the system in case of missed heartbeat from client.
	Use the following command to enable system halt when process heartbeat fails:
	`gabconfig -b`
	Use the following command to disable system halt when process heartbeat fails:
	`gabconfig -B`

Table 5-2 GAB dynamic tunable parameters *(continued)*

GAB parameter	Description and command

Halt on rejoin	Default: Disabled
	
	This option allows the user to configure the behavior of the VCS engine or any other user process when one or more nodes rejoin a cluster after a network partition. By default GAB will not PANIC the node running the VCS engine. GAB kills the userland process (the VCS engine or the vxconfigd process). This recycles the user port (port h in case of the VCS engine) and clears up messages with the old generation number programmatically. Restart of the process, if required, must be handled outside of GAB control, e.g., for hashadow process restarts _had.
	
	When GAB has kernel clients (such as fencing, VxVM, or VxFS), then the node will always PANIC when it rejoins the cluster after a network partition. The PANIC is mandatory since this is the only way GAB can clear ports and remove old messages.
	
	Use the following command to enable system halt on rejoin:
	
	`gabconfig -j`
	
	Use the following command to disable system halt on rejoin:
	
	`gabconfig -J`

Keep on killing	Default: Disabled If this option is enabled, then GAB prevents the system from PANICKING when the VCS engine or the vxconfigd process fail to heartbeat with GAB and GAB fails to kill the VCS engine or the vxconfigd process. GAB will try to continuously kill the VCS engine and will not panic if the kill fails. Repeat attempts to kill process if it does not die `gabconfig -k`

Table 5-2 GAB dynamic tunable parameters *(continued)*

GAB parameter	Description and command

Quorum flag	Default: Disabled
	This is an option in GAB which allows a node to IOFENCE (resulting in a PANIC) if the new membership set is < 50% of the old membership set. This option is typically disabled and is used when integrating with other products
	Enable iofence quorum
	`gabconfig -q`
	Disable iofence quorum
	`gabconfig -d`

GAB queue limit	Default: Send queue limit: 128
	Default: Recv queue limit: 128
	GAB queue limit option controls the number of pending message before which GAB sets flow. Send queue limit controls the number of pending message in GAB send queue. Once GAB reaches this limit it will set flow control for the sender process of the GAB client. GAB receive queue limit controls the number of pending message in GAB receive queue before GAB send flow control for the receive side.
	Set the send queue limit to specified value
	`gabconfig -Q sendq:value`
	Set the receive queue limit to specified value
	`gabconfig -Q recvq:value`

IOFENCE timeout	Default: 15000(ms) This parameter specifies the timeout (in milliseconds) for which GAB will wait for the clients to respond to an IOFENCE message before taking next action. Based on the value of kill_ntries , GAB will attempt to kill client process by sending SIGABRT signal. If the client process is still registered after GAB attempted to kill client process for the value of kill_ntries times, GAB will halt the system after waiting for additional isolate_timeout value. Set the iofence timeout value to specified value in milliseconds. `gabconfig -f value`

Table 5-2 GAB dynamic tunable parameters *(continued)*

GAB parameter	Description and command
Stable timeout	Default: 5000(ms) Specifies the time GAB waits to reconfigure membership after the last report from LLT of a change in the state of local node connections for a given port. Any change in the state of connections will restart GAB waiting period. Set the stable timeout to specified value `gabconfig -t stable`
Isolate timeout	Default: 120000(ms) This tunable specifies the timeout value for which GAB will wait for client process to unregister in response to GAB sending SIGKILL signal. If the process still exists after isolate timeout GAB will halt the system `gabconfig -S isolate_time:value`
Kill_ntries	Default: 5 This tunable specifies the number of attempts GAB will make to kill the process by sending SIGABRT signal. `gabconfig -S kill_ntries:value`
Driver state	This parameter shows whether GAB is configured. GAB may not have seeded and formed any membership yet.
Partition arbitration	This parameter shows whether GAB is asked to specifically ignore jeopardy. See the `gabconfig` (1M) manual page for details on the `-s` flag.

About LLT tunable parameters

LLT provides various configuration and tunable parameters to modify and control the behavior of the LLT module. This section describes some of the LLT tunable parameters that can be changed at run-time and at LLT start-time.

The tunable parameters are classified into two categories:

- LLT timer tunable parameters
  See "About LLT timer tunable parameters" on page 155.

- LLT flow control tunable parameters
  See "About LLT flow control tunable parameters" on page 159.

You can set the LLT tunable parameters either with the `lltconfig` command or in the `/etc/llttab` file. You can use the `lltconfig` command to change a parameter on the local node at run time. Symantec recommends you run the command on all the nodes in the cluster to change the values of the parameters. To set an LLT parameter across system reboots, you must include the parameter definition in the `/etc/llttab` file. Default values of the parameters are taken if nothing is specified in /etc/llttab. The parameters values specified in the /etc/llttab file come into effect at LLT start-time only. Symantec recommends that you specify the same definition of the tunable parameters in the `/etc/llttab` file of each node.

To set a timer tunable:

- To set a timer tunable parameter using the `lltconfig` command:

```
# lltconfig -T timer tunable:value
```

- To set a timer tunable parameter in the `/etc/llttab` file:

```
set-timer  timer tunable:value
```

To set a flow control tunable

- To set a flow control tunable parameter using the `lltconfig` command:

```
# lltconfig -F flowcontrol tunable:value
```

- To set a flow control tunable parameter in the `/etc/llttab` file:

```
set-flow  flowcontrol tunable:value
```

See the `lltconfig`(1M) and `llttab`(1M) manual pages.

About LLT timer tunable parameters

Table 5-3 lists the LLT timer tunable parameters. The timer values are set in .01 sec units. The command `lltconfig -T query` can be used to display current timer values.

Table 5-3 LLT timer tunable parameters

LLT parameter	Description	Default	When to change	Dependency with other LLT tunable parameters
peerinact	LLT marks a link of a peer node as “inactive," if it does not receive any packet on that link for this timer interval. Once a link is marked as "inactive," LLT will not send any data on that link.	1600	■ Change this value for delaying or speeding up node/link inactive notification mechanism as per client’s notification processing logic. ■ Increase the value for planned replacement of faulty network cable /switch. ■ In some circumstances, when the private networks links are very slow or the network traffic becomes very bursty, increase this value so as to avoid false notifications of peer death. Set the value to 0 for planned replacement of faulty network cable or faulty switch.	The timer value should always be higher than the peertrouble timer value.
peertrouble	LLT marks a high-pri link of a peer node as "troubled", if it does not receive any packet on that link for this timer interval. Once a link is marked as "troubled", LLT will not send any data on that link till the link is up.	200	■ In some circumstances, when the private networks links are very slow or nodes in the cluster are very busy, increase the value. ■ Increase the value for planned replacement of faulty network cable /faulty switch.	This timer value should always be lower than peerinact timer value. Also, It should be close to its default value.

Table 5-3 LLT timer tunable parameters *(continued)*

LLT parameter	Description	Default	When to change	Dependency with other LLT tunable parameters
peertroublelo	LLT marks a low-pri link of a peer node as "troubled", if it does not receive any packet on that link for this timer interval. Once a link is marked as "troubled", LLT will not send any data on that link till the link is available.	400	■ In some circumstances, when the private networks links are very slow or nodes in the cluster are very busy, increase the value. ■ Increase the value for planned replacement of faulty network cable /faulty switch.	This timer value should always be lower than peerinact timer value. Also, It should be close to its default value.
heartbeat	LLT sends heartbeat packets repeatedly to peer nodes after every heartbeat timer interval on each high link.	50	In some circumstances, when the private networks links are very slow (or congested) or nodes in the cluster are very busy, increase the value.	This timer value should be lower than peertrouble timer value. Also, it should not be close to peertrouble timer value.
heartbeatlo	LLT sends heartbeat packets repeatedly to peer nodes after every heartbeatlo timer interval on each low pri link.	100	In some circumstances, when the networks links are very slow or nodes in the cluster are very busy, increase the value.	This timer value should be lower than peertroublelo timer value. Also, it should not be close to peertroublelo timer value.
timetoreqhb	LLT attempts to request heartbeat mechanism (explicitly sends 5 hbreqs, special kind of heartbeat requests to the peer node on this link) if it did not receive any packets on this link from that peer node for this timer interval. If the peer node does not respond to the special heartbeat requests, LLT marks this link as “expired” for that peer node. Value can be set within range starting from 0 to (peerinact – 200). If this timer is set to 0, the request heartbeat mechanism is disabled.	1400	Decrease the value of this tunable for speeding up node/link inactive notification mechanism as per client’s notification processing logic. Disable the request heartbeat mechanism by setting the value of this timer to 0 for planned replacement of faulty network cable /switch. In some circumstances, when the private networks links are very slow or the network traffic becomes very bursty, don’t change the value of this timer tunable.	This timer is set to ‘peerinact - 200’ automatically every time when the peerinact timer is changed.

Table 5-3 LLT timer tunable parameters *(continued)*

LLT parameter	Description	Default	When to change	Dependency with other LLT tunable parameters
reqhbtime	This value specifies the time interval between two successive special heartbeat requests. See the timetoreqhb parameter for more information on special heartbeat requests.	40	Do not change this value	Not applicable
timetosendhb	LLT sends out of timer context heartbeats to keep the node alive when LLT timer does not run at regular interval. This option specifies the amount of time to wait before sending a heartbeat in case of timer not running. If this timer tunable is set to 0, the out of timer context heartbeating mechanism is disabled.	200	Disable the out of timer context heart-beating mechanism by setting the value of this timer to 0 for planned replacement of faulty network cable /switch. In some circumstances, when the private networks links are very slow or nodes in the cluster are very busy, increase the value	This timer value should not be more than peerinact timer value. Also, it should not be close to the peerinact timer value.
sendhbcap	This value specifies the maximum time for which LLT will send contiguous out of timer context heartbeats.	18000		
oos	If the out-of-sequence timer has expired for a node, LLT sends an appropriate NAK to that node. LLT does not send a NAK as soon as it receives an oos packet. It waits for the oos timer value before sending the NAK.	10	Do not change this value for performance reasons. Lowering the value can result in unnecessary retransmissions/negative acknowledgement traffic. You can increase the value of oos if the round trip time is large in the cluster (for example, campus cluster).	Not applicable

Table 5-3 LLT timer tunable parameters *(continued)*

LLT parameter	Description	Default	When to change	Dependency with other LLT tunable parameters
retrans	LLT retransmits a packet if it does not receive its acknowledgement for this timer interval value.	10	Do not change this value. Lowering the value can result in unnecessary retransmissions. You can increase the value of retrans if the round trip time is large in the cluster (for example, campus cluster).	Not applicable
service	LLT calls its service routine (which delivers messages to LLT clients) after every service timer interval.	100	Do not change this value for performance reasons.	Not applicable
arp	LLT flushes stored address of peer nodes when this timer expires and relearns the addresses.	0	Do not change this value for performance reasons.	Not applicable
arpreq	LLT sends an arp request when this timer expires to detect other peer nodes in the cluster.	3000	Do not change this value for performance reasons.	Not applicable

About LLT flow control tunable parameters

Table 5-4 lists the LLT flow control tunable parameters. The flow control values are set in number of packets. The command `lltconfig -F query` can be used to display current flow control settings.

Table 5-4 LLT flow control tunable parameters

LLT parameter	Description	Default	When to change	Dependency with other LLT tunable parameters
highwater	When the number of packets in transmit queue for a node reaches highwater, LLT is flow controlled.	200	If a client generates data in bursty manner, increase this value to match the incoming data rate. Note that increasing the value means more memory consumption so set an appropriate value to avoid wasting memory unnecessarily. Lowering the value can result in unnecessary flow controlling the client.	This flow control value should always be higher than the lowwater flow control value.
lowwater	When LLT has flow controlled the client, it will not start accepting packets again till the number of packets in the port transmit queue for a node drops to lowwater.	100	Not applicable	This flow control value should be lower than the highwater flow control value. The value should not be close the highwater flow control value.
rporthighwater	When the number of packets in the receive queue for a port reaches highwater, LLT is flow controlled.	200	If a client generates data in bursty manner, increase this value to match the incoming data rate. Note that increasing the value means more memory consumption so set an appropriate value to avoid wasting memory unnecessarily. Lowering the value can result in unnecessary flow controlling the client.	This flow control value should always be higher than the rportlowwater flow control value.
rportlowwater	When LLT has flow controlled the client, it will not start accepting packets for that client again till the number of packets in the port receive queue for the port drops to rportlowwater.	100	Not applicable	This flow control value should be lower than the rpothighwater flow control value. The value should not be close the rporthighwater flow control value.

Table 5-4 LLT flow control tunable parameters *(continued)*

LLT parameter	Description	Default	When to change	Dependency with other LLT tunable parameters
window	This is the maximum number of un-ACKed packets LLT will put in flight.	50	Change the value as per the private networks speed. Lowering the value irrespective of network speed may result in unnecessary retransmission of out of window sequence packets.	This flow control value should not be higher than the difference between the highwater flow control value and the lowwater flow control value. The value of this parameter (window) should be aligned with the value of the bandwidth delay product.
linkburst	This flow control value should not be higher than the difference between the highwater flow control value and the lowwater flow control value.	32	For performance reasons, its value should be either 0 or at least 32.	Not applicable
ackval	LLT sends acknowledgement of a packet by piggybacking an ACK packet on the next outbound data packet to the sender node. If there are no data packets on which to piggyback the ACK packet, LLT waits for ackval number of packets before sending an explicit ACK to the sender.	10	Do not change this value for performance reasons. Increasing the value can result in unnecessary retransmissions.	Not applicable
sws	To avoid Silly Window Syndrome, LLT transmits more packets only when the count of un-acked packet goes to below of this tunable value.	40	For performance reason, its value should be changed whenever the value of the window tunable is changed as per the formula given below: sws = window *4/5.	Its value should be lower than that of window. Its value should be close to the value of window tunable.
largepktlen	When LLT has packets to transmit to multiple ports, LLT transmits one large packet or up to five small packets to a port at a time. This parameter specifies the size of the large packet.	1024	Do not change this value for performance reasons.	Not applicable

About VXFEN tunable parameters

The section describes the VXFEN tunable parameters and how to reconfigure the VXFEN module.

Table 5-5 describes the tunable parameters for the VXFEN driver.

Table 5-5 VXFEN tunable parameters

vxfen Parameter	Description and Values: Default, Minimum, and Maximum
dbg_log_size	Size of debug log in bytes ■ Values Default: 131072 (128 KB) Minimum: 65536 (64 KB) Maximum: 524288 (512 KB)
vxfen_max_delay	Specifies the maximum number of seconds that the smaller sub-cluster waits before racing with larger sub-clusters for control of the coordinator disks when a network partition occurs. This value must be greater than the vxfen_min_delay value. ■ Values Default: 60 Minimum: 1 Maximum: 60
vxfen_min_delay	Specifies the minimum number of seconds that the smaller sub-cluster waits before racing with larger sub-clusters for control of the coordinator disks when a network partition occurs. This value must be smaller than or equal to the vxfen_max_delay value. ■ Values Default: 1 Minimum: 1 Maximum: 60
vxfen_vxfnd_tmt	Specifies the time in seconds that the I/O fencing driver VxFEN waits for the I/O fencing daemon VXFEND to return after completing a given task. ■ Values Default: 60 Minimum: 10 Maximum: 600

Table 5-5 VXFEN tunable parameters *(continued)*

vxfen Parameter	Description and Values: Default, Minimum, and Maximum
panic_timeout_offst	Specifies the time in seconds based on which the I/O fencing driver VxFEN computes the delay to pass to the GAB module to wait until fencing completes its arbitration before GAB implements its decision in the event of a split-brain. You can set this parameter in the `vxfenmode` file and use the `vxfenadm` command to check the value. Depending on the vxfen_mode, the GAB delay is calculated as follows: ■ For sybase mode: 1000 * (panic_timeout_offst + vxfen_max_delay) ■ For customized mode: 1000 * (panic_timeout_offst + max (vxfen_vxfnd_tmt, vxfen_loser_exit_delay)) ■ Default: 10

In the event of a network partition, the smaller sub-cluster delays before racing for the coordinator disks. The time delay 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.

Configuring the VXFEN module parameters

After adjusting the tunable kernel driver parameters, you must reconfigure the VXFEN module for the parameter changes to take effect.

The following example procedure changes the value of the vxfen_min_delay parameter.

On each Linux node, edit the file /etc/sysconfig/vxfen to change the value of the vxfen driver tunable global parameters, vxfen_max_delay and vxfen_min_delay.

Note: You must restart the VXFEN module to put any parameter change into effect.

To configure the VxFEN parameters and reconfigure the VxFEN module

1 Shut down all sybasece service groups on the node.

```
# hagrp -offline sybasece -sys system1

# hagrp -offline binmnt -sys system1
```

2 Stop all the applications that are not configured under VCS. Use native application commands to stop the application.

3 Stop VCS on all the nodes. Run the following command on each node:

```
# hastop -local
```

4 Stop the VxFEN driver.

```
# /etc/init.d/vxfen stop
```

5 Edit the /etc/sysconfig/vxfen file.

For example, change the entry from:

```
vxfen_min_delay=1
```

to:

```
vxfen_min_delay=30
```

6 Start the VXFEN module.

```
# /etc/init.d/vxfen start
```

7 Bring the service groups online.

```
# hagrp -online binmnt -sys system1
# hagrp -online sybasece -sys system1
```

8 Start all the applications that are not configured under VCS. Use native application commands to start the applications.

Appendix A

Error messages

This appendix includes the following topics:

- About error messages
- VxVM error messages
- VXFEN driver error messages

About error messages

Error messages can be generated by the following software modules:

- Veritas Volume Manager (VxVM)
- Veritas Fencing (VXFEN) driver

VxVM error messages

Table A-1 contains VxVM error messages that are related to I/O fencing.

Table A-1 VxVM error messages for I/O fencing

Message	Explanation
`vold_pgr_register(disk_path): failed to open the vxfen device. Please make sure that the vxfen driver is installed and configured.`	The vxfen driver is not configured. Follow the instructions to set up these disks and start I/O fencing. You can then clear the faulted resources and bring the service groups online.
`vold_pgr_register(disk_path): Probably incompatible vxfen driver.`	Incompatible versions of VxVM and the vxfen driver are installed on the system. Install the proper version of SF Sybase CE.

VXFEN driver error messages

Table A-2 contains VXFEN driver error messages. In addition to VXFEN driver error messages, informational messages can also be displayed.

See "VXFEN driver informational message" on page 166.

See "Node ejection informational messages" on page 167.

Table A-2 VXFEN driver error messages

Message	Explanation
Unable to register with coordinator disk with serial number: xxxx	This message appears when the vxfen driver is unable to register with one of the coordinator disks. The serial number of the coordinator disk that failed is displayed.
Unable to register with a majority of the coordinator disks. Dropping out of cluster.	This message appears when the vxfen driver is unable to register with a majority of the coordinator disks. The problems with the coordinator disks must be cleared before fencing can be enabled. This message is preceded with the message "VXFEN: Unable to register with coordinator disk with serial number xxxx."
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!	This message appears when there is a preexisting split-brain in the cluster. In this case, the configuration of vxfen driver fails. Clear the split-brain using the instructions given in the chapter on Troubleshooting SF Oracle RAC before configuring vxfen driver.
Unable to join running cluster since cluster is currently fencing a node out of the cluster	This message appears while configuring the vxfen driver, if there is a fencing race going on in the cluster. The vxfen driver can be configured by retrying after some time (after the cluster completes the fencing).

VXFEN driver informational message

The following informational message appears when a node is ejected from the cluster to prevent data corruption when a split-brain occurs.

```
VXFEN CRITICAL V-11-1-20 Local cluster node ejected from cluster
to prevent potential data corruption
```

Node ejection informational messages

Informational messages may appear on the console of one of the cluster nodes when a node is ejected from a disk or LUN.

These informational messages can be ignored.

Index

A

agent for SQL server
  functions 101
agent log
  format 116
  location 116
agents
  intelligent resource monitoring 26
  poll-based resource monitoring 26
AMF driver 26

B

binary message catalogs
  about 121
  location of 121

C

Changing the CVM master 94
cluster
  Group Membership Services/Atomic Broadcast (GAB) 18
  interconnect communication channel 17
  low latency transport (LLT) 17
Cluster File System (CFS)
  architecture 23
  communication 23
  overview 22
cluster manager 106
Cluster master node
  changing 94
Cluster Volume Manager (CVM)
  architecture 20
  communication 21
  overview 20
commands
  format (verify disks) 134
  vxdctl enable (scan disks) 134
communication
  communication stack 16
  data flow 16
communication *(continued)*
  GAB and processes port relationship 19
  Group Membership Services/Atomic Broadcast GAB 18
  interconnect communication channel 17
  requirements 16
configuration wizard 106
configuring service groups
  cluster manager (Java Console) 106
  command line 107
coordinator disks
  DMP devices 31
  for I/O fencing 31
CVM master
  changing 94

D

data corruption
  preventing 29
data disks
  for I/O fencing 31
drivers
  tunable parameters 147

E

engine log
  format 116
  location 116
environment variables
  MANPATH 44
error messages
  agent log 116
  engine log 116
  message catalogs 121
  node ejection 167
  VxVM errors related to I/O fencing 165

F

format command 134

G

GAB
    tunable parameters 147
GAB tunable parameters
    dynamic 149
        Control port seed 149
        Driver state 149
        Gab queue limit 149
        Halt on process death 149
        Halt on rejoin 149
        IOFENCE timeout 149
        Isolate timeout 149
        Keep on killing 149
        Kill_ntries 149
        Missed heartbeat halt 149
        Partition arbitration 149
        Quorum flag 149
        Stable timeout 149
    static 148
        gab_conn_wait 148
        gab_flowctrl 148
        gab_isolate_time 148
        gab_kill_ntries 148
        gab_kstat_size 148
        gab_logbufsize 148
        gab_msglogsize 148
        gab_numnids 148
        gab_numports 148
Global Cluster Option (GCO)
    overview 39

I

I/O fencing
    operations 30
    preventing data corruption 29
    testing and scenarios 31
intelligent resource monitoring
    disabling manually 61
    enabling manually 61

K

kernel
    tunable driver parameters 147

L

LLT
    tunable parameters 154
logging
    agent log 116
    engine log 116
    message tags 116
low latency transport (LLT)
    overview 17

M

MANPATH environment variable 44
Master node
    changing 94
message tags
    about 116
messages
    node ejected 167
    VXFEN driver error messages 166
monitoring
    basic 104
    detail 104

R

reservations
    description 30

S

SCSI-3 PR 30
service group
    viewing log 110
SF Sybase CE
    architecture 11, 14
    communication infrastructure 15
    error messages 165
    high-level functionality 11
    tunable parameters 147
SF Sybase CE components
    Cluster Volume Manager (CVM) 20
SF Sybase CE installation
    pre-installation tasks
        setting MANPATH 44
Switching the CVM master 94
Sybase agent
    configuring using cluster manager 106
    configuring using command line 107
    monitoring options 104
Sybase high availability 39
Sybase instance
    definition 11

T

troubleshooting
    CVMVolDg 136
    logging 116
    overview of topics 134, 137
    restoring communication after cable disconnection 134
    running scripts for analysis 111
    shared disk group cannot be imported 134

V

VCS
    logging 116
Veritas Operations Manager 40
vxdctl command 134
VXFEN driver error messages 166
VXFEN driver informational message 166
VxVM
    error messages related to I/O fencing 165
VxVM (Volume Manager)
    errors related to I/O fencing 165