

Veritas™ Volume Manager Troubleshooting Guide

Solaris

5.1 Service Pack 1

Veritas™ Volume Manager Troubleshooting Guide

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

Product version: 5.1 SP1

Document version: 5.1SP1.1

Legal Notice

Copyright © 2010 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

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:

docs@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>

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

Contents

Technical Support	4
Chapter 1 Recovering from hardware failure	11
About recovery from hardware failure	11
Listing unstartable volumes	12
Displaying volume and plex states	13
The plex state cycle	13
Recovering an unstartable mirrored volume	16
Recovering an unstartable volume with a disabled plex in the RECOVER state	17
Forcibly restarting a disabled volume	17
Clearing the failing flag on a disk	18
Reattaching failed disks	19
Failures on RAID-5 volumes	20
System failures	20
Disk failures	21
Default startup recovery process for RAID-5	23
Recovery of RAID-5 volumes	23
Recovery after moving RAID-5 subdisks	26
Unstartable RAID-5 volumes	27
Recovering from an incomplete disk group move	29
Recovery from failure of a DCO volume	30
Recovering a version 0 DCO volume	32
Recovering a version 20 DCO volume	35
Chapter 2 Recovering from instant snapshot failure	37
Recovering from the failure of vxsnap prepare	37
Recovering from the failure of vxsnap make for full-sized instant snapshots	38
Recovering from the failure of vxsnap make for break-off instant snapshots	39
Recovering from the failure of vxsnap make for space-optimized instant snapshots	39
Recovering from the failure of vxsnap restore	40
Recovering from the failure of vxsnap reattach or refresh	40

	Recovering from copy-on-write failure	41
	Recovering from I/O errors during resynchronization	42
	Recovering from I/O failure on a DCO volume	42
Chapter 3	Recovering from boot disk failure	43
	VxVM and boot disk failure	43
	Possible root, swap, and usr configurations	44
	Booting from alternate boot disks	45
	Booting from an alternate boot disk on Solaris SPARC systems	45
	Booting from an alternate (mirror) boot disk on Solaris x64 systems	47
	Hot-relocation and boot disk failure	53
	Unrelocation of subdisks to a replacement boot disk	54
	Recovery from boot failure	54
	Boot device cannot be opened	55
	Cannot boot from unusable or stale plexes	55
	Invalid UNIX partition	57
	Incorrect entries in /etc/vfstab	57
	Missing or damaged configuration files	59
	Repair of root or /usr file systems on mirrored volumes	61
	Recovering a root disk and root mirror from a backup	62
	Replacement of boot disks	65
	Re-adding a failed boot disk	66
	Replacing a failed boot disk	67
	Recovery by reinstallation	68
	General reinstallation information	69
	Reinstalling the system and recovering VxVM	69
Chapter 4	Logging commands and transactions	81
	Command logs	81
	Transaction logs	83
	Association of command and transaction logs	85
	Associating CVM commands issued from slave to master node	85
Chapter 5	Backing up and restoring disk group configurations	89
	About disk group configuration backup	89
	Backing up a disk group configuration	90
	Restoring a disk group configuration	91
	Resolving conflicting backups for a disk group	93

Chapter 6	Restoring a previous array support library	95
	Downgrading the array support	95
Chapter 7	DMP troubleshooting	97
	Displaying extended attributes after upgrading to DMP 5.1SP1	97
Chapter 8	Error messages	99
	About error messages	99
	How error messages are logged	99
	Configuring logging in the startup script	100
	Types of messages	102
	Messages	103
Index		161

Recovering from hardware failure

This chapter includes the following topics:

- [About recovery from hardware failure](#)
- [Listing unstartable volumes](#)
- [Displaying volume and plex states](#)
- [The plex state cycle](#)
- [Recovering an unstartable mirrored volume](#)
- [Recovering an unstartable volume with a disabled plex in the RECOVER state](#)
- [Forcibly restarting a disabled volume](#)
- [Clearing the failing flag on a disk](#)
- [Reattaching failed disks](#)
- [Failures on RAID-5 volumes](#)
- [Recovering from an incomplete disk group move](#)
- [Recovery from failure of a DCO volume](#)

About recovery from hardware failure

Symantec's 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.

If a volume has a disk I/O failure (for example, because the disk has an uncorrectable error), VxVM can detach the plex involved in the failure. I/O stops on that plex but continues on the remaining plexes of the volume.

If a disk fails completely, VxVM can detach the disk from its disk group. All plexes on the disk are disabled. If there are any unmirrored volumes on a disk when it is detached, those volumes are also disabled.

Note: Apparent disk failure may not be due to a fault in the physical disk media or the disk controller, but may instead be caused by a fault in an intermediate or ancillary component such as a cable, host bus adapter, or power supply.

The hot-relocation feature in VxVM automatically detects disk failures, and notifies the system administrator and other nominated users of the failures by electronic mail. Hot-relocation also attempts to use spare disks and free disk space to restore redundancy and to preserve access to mirrored and RAID-5 volumes.

For more information about administering hot-relocation, see the *Veritas Volume Manager Administrator's Guide*.

Recovery from failures of the boot (`root`) disk and repairing the `root (/)` and `usr` file systems require the use of the special procedures.

See “[VxVM and boot disk failure](#)” on page 43.

Listing unstartable volumes

An unstartable volume can be incorrectly configured or have other errors or conditions that prevent it from being started. To display unstartable volumes, use the `vxinfo` command. This displays information about the accessibility and usability of volumes.

To list unstartable volumes

- ◆ Type the following command:

```
# vxinfo [-g diskgroup] [volume ...]
```

The following example output shows one volume, `mkting`, as being unstartable:

home	fsgen	Started
mkting	fsgen	Unstartable
src	fsgen	Started
rootvol	root	Started
swapvol	swap	Started

Displaying volume and plex states

To display detailed information about the configuration of a volume including its state and the states of its plexes, use the `vxprint` command.

To display volume and plex states

- ◆ Type the following command:

```
# vxprint [-g diskgroup] -hvt [volume ...]
```

The following example shows a disabled volume, `vol`, which has two clean plexes, `vol-01` and `vol-02`, each with a single subdisk:

```
# vxprint -g mydg -hvt vol
```

```
Disk group: mydg
```

V	NAME	RVG/VSET/CO	KSTATE	STATE	LENGTH	READPOL	PREFPLEX	UTYPE
PL	NAME	VOLUME	KSTATE	STATE	LENGTH	LAYOUT	NCOL/WID	MODE
SD	NAME	PLEX	DISK	DISKOFFSLENGTH		[COL/] OFF	DEVICE	MODE
SV	NAME	PLEX	VOLNAME	NVOLLAYRLLENGTH		[COL/] OFF	AM/NM	MODE
SC	NAME	PLEX	CACHE	DISKOFFSLENGTH		[COL/] OFF	DEVICE	MODE
DC	NAME	PARENTVOL	LOGVOL					
SP	NAME	SNAPVOL	DCO					
v	vol	-	DISABLED	ACTIVE	212880	SELECT	-	fsgen
pl	vol-01	vol	DISABLED	CLEAN	212880	CONCAT	-	RW
sd	mydg11-01	vol-01	mydg11	0	212880	0	c1t0d0	ENA
pl	vol-02	vol	DISABLED	CLEAN	212880	CONCAT	-	RW
sd	mydg12-01	vol-02	mydg12	0	212880	0	c1t1d0	ENA

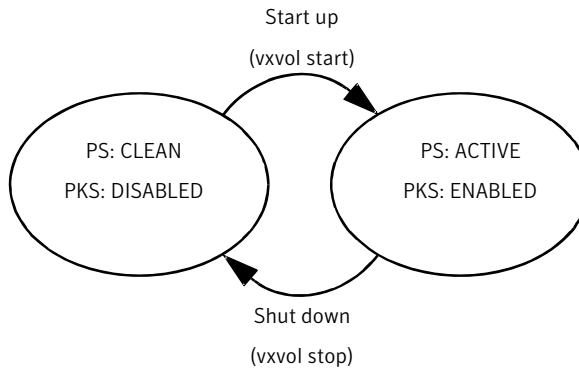
See the *Veritas Volume Manager Administrator's Guide* for a description of the possible plex and volume states.

The plex state cycle

Changing plex states are part of normal operations, and do not necessarily indicate abnormalities that must be corrected. A clear understanding of the various plex states and their interrelationship is necessary if you want to be able to perform any recovery procedures.

[Figure 1-1](#) shows the main transitions that take place between plex states in VxVM.

Figure 1-1 Main plex state cycle



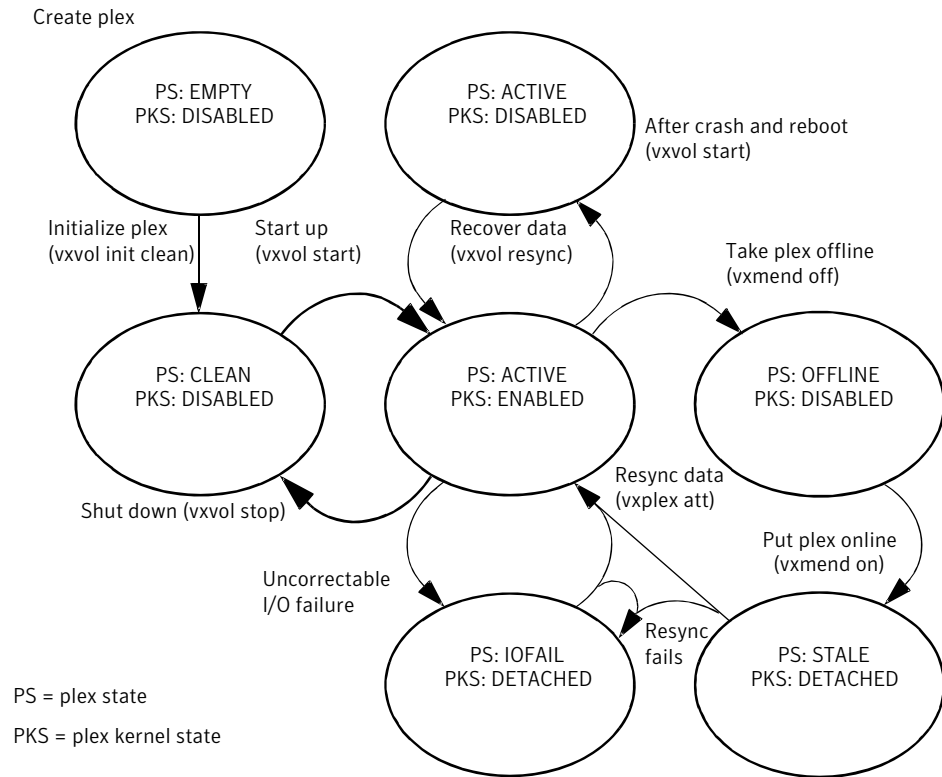
PS = plex state

PKS = plex kernel state

For more information about plex states, see the *Veritas Volume Manager Administrator's Guide*.

At system startup, volumes are started automatically and the `vxvol start` task makes all CLEAN plexes ACTIVE. At shutdown, the `vxvol stop` task marks all ACTIVE plexes CLEAN. If all plexes are initially CLEAN at startup, this indicates that a controlled shutdown occurred and optimizes the time taken to start up the volumes.

[Figure 1-2](#) shows additional transitions that are possible between plex states as a result of hardware problems, abnormal system shutdown, and intervention by the system administrator.

Figure 1-2 Additional plex state transitions

When first created, a plex has state **EMPTY** until the volume to which it is attached is initialized. Its state is then set to **CLEAN**. Its plex kernel state remains set to **DISABLED** and is not set to **ENABLED** until the volume is started.

After a system crash and reboot, all plexes of a volume are **ACTIVE** but marked with plex kernel state **DISABLED** until their data is recovered by the `vxvol resync` task.

A plex may be taken offline with the `vxmend off` command, made available again using `vxmend on`, and its data resynchronized with the other plexes when it is reattached using `vxplex att`. A failed resynchronization or uncorrectable I/O failure places the plex in the **IOFAIL** state.

There are various actions that you can take if a system crash or I/O error leaves no plexes of a mirrored volume in a **CLEAN** or **ACTIVE** state.

See [“Recovering an unstartable mirrored volume”](#) on page 16.

See [“Failures on RAID-5 volumes”](#) on page 20.

Recovering an unstartable mirrored volume

A system crash or an I/O error can corrupt one or more plexes of a mirrored volume and leave no plex `CLEAN` or `ACTIVE`. You can mark one of the plexes `CLEAN` and instruct the system to use that plex as the source for reviving the others.

To recover an unstartable mirrored volume

- 1 Place the desired plex in the `CLEAN` state using the following command:

```
# vxmend [-g diskgroup] fix clean plex
```

For example, to place the plex `vol101-02` in the `CLEAN` state:

```
# vxmend -g mydg fix clean vol101-02
```

- 2 To recover the other plexes in a volume from the `CLEAN` plex, the volume must be disabled, and the other plexes must be `STALE`. If necessary, make any other `CLEAN` or `ACTIVE` plexes `STALE` by running the following command on each of these plexes in turn:

```
# vxmend [-g diskgroup] fix stale plex
```

Following severe hardware failure of several disks or other related subsystems underlying all the mirrored plexes of a volume, it may be impossible to recover the volume using `vxmend`. In this case, remove the volume, recreate it on hardware that is functioning correctly, and restore the contents of the volume from a backup or from a snapshot image.

See the `vxmend(1M)` manual page.

- 3 To enable the `CLEAN` plex and to recover the `STALE` plexes from it, use the following command:

```
# vxvol [-g diskgroup] start volume
```

For example, to recover volume `vol101`:

```
# vxvol -g mydg start vol101
```

See the `vxvol(1M)` manual page.

Recovering an unstartable volume with a disabled plex in the RECOVER state

A plex is shown in the RECOVER state if its contents are out-of-date with respect to the volume. This can happen if a disk containing one or more of the plex's subdisks has been replaced or reattached. If a plex is shown as being in this state, it can be recovered by using the `vxmend` and `vxvol` commands.

To recover an unstartable volume with a disabled plex in the RECOVER state

- 1 Use the following command to force the plex into the OFFLINE state:

```
# vxmend [-g diskgroup] -o force off plex
```

- 2 Place the plex into the STALE state using this command:

```
# vxmend [-g diskgroup] on plex
```

- 3 If there are other ACTIVE or CLEAN plexes in the volume, use the following command to reattach the plex to the volume:

```
# vxplex [-g diskgroup] att volume plex
```

If the volume is already enabled, resynchronization of the plex is started immediately.

If there are no other clean plexes in the volume, use this command to make the plex DISABLED and CLEAN:

```
# vxmend [-g diskgroup] fix clean plex
```

- 4 If the volume is not already enabled, use the following command to start it, and preform any resynchronization of the plexes in the background:

```
# vxvol [-g diskgroup] -o bg start volume
```

If the data in the plex was corrupted, and the volume has no ACTIVE or CLEAN redundant plexes from which its contents can be resynchronized, it must be restored from a backup or from a snapshot image.

Forcibly restarting a disabled volume

If a disk failure caused a volume to be disabled, and the volume does not contain any valid redundant plexes, you must restore the volume from a backup after

replacing the failed disk. Any volumes that are listed as `Unstartable` must be restarted using the `vxvol` command before restoring their contents from a backup.

To forcibly restart a disabled volume

- ◆ Type the following command:

```
# vxvol [-g diskgroup] -o bg -f start volume
```

The `-f` option forcibly restarts the volume, and the `-o bg` option resynchronizes its plexes as a background task. For example, to restart the volume `myvol` so that it can be restored from backup, use the following command:

```
# vxvol -g mydg -o bg -f start myvol
```

Clearing the failing flag on a disk

If I/O errors are intermittent rather than persistent, Veritas Volume Manager sets the `failing` flag on a disk, rather than detaching the disk. Such errors can occur due to the temporary removal of a cable, controller faults, a partially faulty LUN in a disk array, or a disk with a few bad sectors or tracks.

If the hardware fault is not with the disk itself (for example, it is caused by problems with the controller or the cable path to the disk), you can use the `vxedit` command to unset the `failing` flag after correcting the source of the I/O error.

Warning: Do not unset the `failing` flag if the reason for the I/O errors is unknown. If the disk hardware truly is failing, and the flag is cleared, there is a risk of data loss.

To clear the failing flag on a disk

- 1 Use the `vxdisk list` command to find out which disks are failing:

```
# vxdisk list
```

DEVICE	TYPE	DISK	GROUP	STATUS
c1t1d0s2	auto:sliced	mydg01	mydg	online
c1t2d0s2	auto:sliced	mydg02	mydg	online failing
c1t3d0s2	auto:sliced	mydg03	mydg	online
. . .				

- 2 Use the `vxedit set` command to clear the flag for each disk that is marked as failing (in this example, `mydg02`):

```
# vxedit set failing=off mydg02
```

- 3 Use the `vxdisk list` command to verify that the `failing` flag has been cleared:

```
# vxdisk list
```

DEVICE	TYPE	DISK	GROUP	STATUS
c1t1d0s2	auto:sliced	mydg01	mydg	online
c1t2d0s2	auto:sliced	mydg02	mydg	online
c1t3d0s2	auto:sliced	mydg03	mydg	online
. . .				

Reattaching failed disks

You can perform a reattach operation if a disk could not be found at system startup, or if VxVM is started with some disk drivers unloaded and unloadable (causing disks to enter the `failed` state). If the underlying problem has been fixed (such as a cable or controller fault), use the `vxreattach` command to reattach the disks without plexes being flagged as STALE. However, the reattach must occur before any volumes on the disk are started.

The `vxreattach` command is called as part of disk recovery from the `vxdiskadm` menus and during the boot process. If possible, `vxreattach` reattaches the failed disk media record to the disk with the same device name. Reattachment places a disk in the same disk group that it was located in before and retains its original disk media name.

To reattach a failed disk

- 1 Use the `vxdisk list` command to see which disks have failed, as shown in the following example:

```
# vxdisk list
```

DEVICE	TYPE	DISK	GROUP	STATUS
c1t1d0s2	auto:sliced	mydg01	mydg	online
c1t2d0s2	auto:sliced	mydg02	mydg	online
-	-	mydg03	mydg	failed was: c1t3d0s2
-	-	mydg04	mydg	failed was: c1t4d0s2

- 2 Once the fault has been corrected, the disks can be reattached by using the following command to rescan the device list:

```
# /usr/sbin/vxdctl enable
```

- 3 Use the `vxreattach` command with no options to reattach the disks:

```
# /etc/vx/bin/vxreattach
```

After reattachment takes place, recovery may not be necessary unless a disk was faulty and had to be replaced. Reattachment can fail if the original (or another) cause for the disk failure still exists.

You can use the command `vxreattach -c` to check whether reattachment is possible, without performing the operation. Instead, it displays the disk group and disk media name where the disk can be reattached.

See the `vxreattach(1M)` manual page.

Failures on RAID-5 volumes

Failures are seen in two varieties: system failures and disk failures. A system failure means that the system has abruptly ceased to operate due to an operating system panic or power failure. Disk failures imply that the data on some number of disks has become unavailable due to a system failure (such as a head crash, electronics failure on disk, or disk controller failure).

System failures

RAID-5 volumes are designed to remain available with a minimum of disk space overhead, if there are disk failures. However, many forms of RAID-5 can have data loss after a system failure. Data loss occurs because a system failure causes the data and parity in the RAID-5 volume to become unsynchronized. Loss of

synchronization occurs because the status of writes that were outstanding at the time of the failure cannot be determined.

If a loss of sync occurs while a RAID-5 volume is being accessed, the volume is described as having stale parity. The parity must then be reconstructed by reading all the non-parity columns within each stripe, recalculating the parity, and writing out the parity stripe unit in the stripe. This must be done for every stripe in the volume, so it can take a long time to complete.

Warning: While the resynchronization of a RAID-5 volume without log plexes is being performed, any failure of a disk within the volume causes its data to be lost.

Besides the vulnerability to failure, the resynchronization process can tax the system resources and slow down system operation.

RAID-5 logs reduce the damage that can be caused by system failures, because they maintain a copy of the data being written at the time of the failure. The process of resynchronization consists of reading that data and parity from the logs and writing it to the appropriate areas of the RAID-5 volume. This greatly reduces the amount of time needed for a resynchronization of data and parity. It also means that the volume never becomes truly stale. The data and parity for all stripes in the volume are known at all times, so the failure of a single disk cannot result in the loss of the data within the volume.

Disk failures

An uncorrectable I/O error occurs when disk failure, cabling or other problems cause the data on a disk to become unavailable. For a RAID-5 volume, this means that a subdisk becomes unavailable. The subdisk cannot be used to hold data and is considered stale and detached. If the underlying disk becomes available or is replaced, the subdisk is still considered stale and is not used.

If an attempt is made to read data contained on a stale subdisk, the data is reconstructed from data on all other stripe units in the stripe. This operation is called a reconstructing-read. This is a more expensive operation than simply reading the data and can result in degraded read performance. When a RAID-5 volume has stale subdisks, it is considered to be in degraded mode.

A RAID-5 volume in degraded mode can be recognized from the output of the `vxprint -ht` command as shown in the following display:

V	NAME	RVG/VSET/COKSTATE	STATE	LENGTH	READPOL	PREFPLEX	UTYPE
PL	NAME	VOLUME	KSTATE	STATE	LENGTH	LAYOUT	NCOL/WID
SD	NAME	PLEX	DISK	DISKOFFS	LENGTH	[COL/]OFF	DEVICE
SV	NAME	PLEX	VOLNAME	NVOLLAYR	LENGTH	[COL/]OFF	AM/NM

```

...
v   r5vol      -          ENABLED  DEGRADED  204800   RAID      -          raid5
pl  r5vol-01   r5vol      ENABLED  ACTIVE    204800   RAID      3/16       RW
sd  disk01-01  r5vol-01disk01    0         102400   0/0       c2t9d0     ENA
sd  disk02-01  r5vol-01disk02    0         102400   1/0       c2t10d0    dS
sd  disk03-01  r5vol-01disk03    0         102400   2/0       c2t11d0    ENA
pl  r5vol-02   r5vol      ENABLED  LOG       1440     CONCAT    -          RW
sd  disk04-01  r5vol-02disk04    0         1440     0         c2t12d0    ENA
pl  r5vol-03   r5vol      ENABLED  LOG       1440     CONCAT    -          RW
sd  disk05-01  r5vol-03disk05    0         1440     0         c2t14d0    ENA

```

The volume `r5vol` is in degraded mode, as shown by the volume state, which is listed as `DEGRADED`. The failed subdisk is `disk02-01`, as shown by the `MODE` flags; `d` indicates that the subdisk is detached, and `s` indicates that the subdisk's contents are stale.

Warning: Do not run the `vxr5check` command on a RAID-5 volume that is in degraded mode.

A disk containing a RAID-5 log plex can also fail. The failure of a single RAID-5 log plex has no direct effect on the operation of a volume provided that the RAID-5 log is mirrored. However, loss of all RAID-5 log plexes in a volume makes it vulnerable to a complete failure. In the output of the `vxprint -ht` command, failure within a RAID-5 log plex is indicated by the plex state being shown as `BADLOG` rather than `LOG`.

In the following example, the RAID-5 log plex `r5vol-02` has failed:

```

V   NAME      RVG/VSET/COKSTATE  STATE  LENGTH  READPOL  PREFPLEX  UTYPE
PL  NAME      VOLUME   KSTATE  STATE  LENGTH  LAYOUT   NCOL/WID  MODE
SD  NAME      PLEX     DISK    DISKOFFS LENGTH  [COL/]OFF DEVICE  MODE
SV  NAME      PLEX     VOLNAME NVOLLAYR LENGTH  [COL/]OFF AM/NM    MODE
...
v   r5vol      -          ENABLED  ACTIVE  204800   RAID      -          raid5
pl  r5vol-01   r5vol      ENABLED  ACTIVE  204800   RAID      3/16       RW
sd  disk01-01  r5vol-01disk01    0       102400   0/0       c2t9d0     ENA
sd  disk02-01  r5vol-01disk02    0       102400   1/0       c2t10d0    ENA
sd  disk03-01  r5vol-01disk03    0       102400   2/0       c2t11d0    ENA
pl  r5vol-02   r5vol      DISABLED  BADLOG  1440     CONCAT    -          RW
sd  disk04-01  r5vol-02disk04    0       1440     0         c2t12d0    ENA
pl  r5vol-03   r5vol      ENABLED  LOG     1440     CONCAT    -          RW
sd  disk05-01  r5vol-12disk05    0       1440     0         c2t14d0    ENA

```

Default startup recovery process for RAID-5

VxVM may need to perform several operations to restore fully the contents of a RAID-5 volume and make it usable. Whenever a volume is started, any RAID-5 log plexes are zeroed before the volume is started. This prevents random data from being interpreted as a log entry and corrupting the volume contents. Also, some subdisks may need to be recovered, or the parity may need to be resynchronized (if RAID-5 logs have failed).

VxVM takes the following steps when a RAID-5 volume is started:

- If the RAID-5 volume was not cleanly shut down, it is checked for valid RAID-5 log plexes.
- If valid log plexes exist, they are replayed. This is done by placing the volume in the `DETACHED` volume kernel state and setting the volume state to `REPLAY`, and enabling the RAID-5 log plexes.
- If no valid logs exist, the parity must be resynchronized. Resynchronization is done by placing the volume in the `DETACHED` volume kernel state and setting the volume state to `SYNC`. Any log plexes are left in the `DISABLED` plex kernel state.

The volume is not made available while the parity is resynchronized because any subdisk failures during this period makes the volume unusable. This can be overridden by using the `-o unsafe` start option with the `vxvol` command. If any stale subdisks exist, the RAID-5 volume is unusable.

Warning: The `-o unsafe` start option is considered dangerous, as it can make the contents of the volume unusable. Using it is not recommended.

- Any existing log plexes are zeroed and enabled. If all logs fail during this process, the start process is aborted.
- If no stale subdisks exist or those that exist are recoverable, the volume is put in the `ENABLED` volume kernel state and the volume state is set to `ACTIVE`. The volume is now started.

Recovery of RAID-5 volumes

The following types of recovery may be required for RAID-5 volumes:

- Resynchronization of parity
- Reattachment of a failed RAID-5 log plex
- Recovery of a stale subdisk

Parity resynchronization and stale subdisk recovery are typically performed when the RAID-5 volume is started, or shortly after the system boots. They can also be performed by running the `vxrecover` command.

See “Unstartable RAID-5 volumes” on page 27.

If hot-relocation is enabled at the time of a disk failure, system administrator intervention is not required unless no suitable disk space is available for relocation. Hot-relocation is triggered by the failure and the system administrator is notified of the failure by electronic mail.

Hot relocation automatically attempts to relocate the subdisks of a failing RAID-5 plex. After any relocation takes place, the hot-relocation daemon (`vxrelocd`) also initiates a parity resynchronization.

In the case of a failing RAID-5 log plex, relocation occurs only if the log plex is mirrored; the `vxrelocd` daemon then initiates a mirror resynchronization to recreate the RAID-5 log plex. If hot-relocation is disabled at the time of a failure, the system administrator may need to initiate a resynchronization or recovery.

Note: Following severe hardware failure of several disks or other related subsystems underlying a RAID-5 plex, it may be only be possible to recover the volume by removing the volume, recreating it on hardware that is functioning correctly, and restoring the contents of the volume from a backup.

Resynchronizing parity on a RAID-5 volume

In most cases, a RAID-5 volume does not have stale parity. Stale parity only occurs after all RAID-5 log plexes for the RAID-5 volume have failed, and then only if there is a system failure. Even if a RAID-5 volume has stale parity, it is usually repaired as part of the volume start process.

If a volume without valid RAID-5 logs is started and the process is killed before the volume is resynchronized, the result is an active volume with stale parity.

The following example is output from the `vxprint -ht` command for a stale RAID-5 volume:

V	NAME	RVG/VSET/COKSTATE	STATE	LENGTH	READPOL	PREFPLEX	UTYPE
PL	NAME	VOLUME	KSTATE	STATE	LENGTH	LAYOUT	NCOL/WID
SD	NAME	PLEX	DISK	DISKOFFS	LENGTH	[COL/]OFF	DEVICE
SV	NAME	PLEX	VOLNAME	NVOLLAYR	LENGTH	[COL/]OFF	AM/NM
...							
v	r5vol	-	ENABLED	NEEDSYNC	204800	RAID	-
pl	r5vol-01	r5vol	ENABLED	ACTIVE	204800	RAID	3/16
sd	disk01-01	r5vol-01disk01	0	102400	0/0	c2t9d0	ENA

```
sd  disk02-01  r5vol-01disk02    0          102400    1/0          c2t10d0    dS
sd  disk03-01  r5vol-01disk03    0          102400    2/0          c2t11d0    ENA
...
```

This output lists the volume state as `NEEDSYNC`, indicating that the parity needs to be resynchronized. The state could also have been `SYNC`, indicating that a synchronization was attempted at start time and that a synchronization process should be doing the synchronization. If no such process exists or if the volume is in the `NEEDSYNC` state, a synchronization can be manually started by using the `resync` keyword for the `vxvol` command.

Parity is regenerated by issuing `VOL_R5_RESYNC ioctl`s to the RAID-5 volume. The resynchronization process starts at the beginning of the RAID-5 volume and resynchronizes a region equal to the number of sectors specified by the `-o iosize` option. If the `-o iosize` option is not specified, the default maximum I/O size is used. The `resync` operation then moves onto the next region until the entire length of the RAID-5 volume has been resynchronized.

For larger volumes, parity regeneration can take a long time. It is possible that the system could be shut down or crash before the operation is completed. In case of a system shutdown, the progress of parity regeneration must be kept across reboots. Otherwise, the process has to start all over again.

To avoid the restart process, parity regeneration is checkpointed. This means that the offset up to which the parity has been regenerated is saved in the configuration database. The `-o checkpoint=size` option controls how often the checkpoint is saved. If the option is not specified, the default checkpoint size is used.

Because saving the checkpoint offset requires a transaction, making the checkpoint size too small can extend the time required to regenerate parity. After a system reboot, a RAID-5 volume that has a checkpoint offset smaller than the volume length starts a parity resynchronization at the checkpoint offset.

To resynchronize parity on a RAID-5 volume

- ◆ Type the following command:

```
# vxvol -g diskgroup resync r5vol
```

Reattaching a failed RAID-5 log plex

RAID-5 log plexes can become detached due to disk failures. These RAID-5 logs can be reattached by using the `att` keyword for the `vxplex` command.

To reattach a failed RAID-5 log plex

- ◆ Type the following command:

```
# vxplex -g diskgroup att r5vol r5vol-plex
```

Recovering a stale subdisk in a RAID-5 volume

Stale subdisk recovery is usually done at volume start time. However, the process doing the recovery can crash, or the volume may be started with an option such as `-o delayrecover` that prevents subdisk recovery. In addition, the disk on which the subdisk resides can be replaced without recovery operations being performed. In such cases, you can perform subdisk recovery by using the `vxvol recover` command.

To recover a stale subdisk in the RAID-5 volume

- ◆ Type the following command:

```
# vxvol -g diskgroup recover r5vol subdisk
```

A RAID-5 volume that has multiple stale subdisks can be recovered in one operation. To recover multiple stale subdisks, use the `vxvol recover` command on the volume:

```
# vxvol -g diskgroup recover r5vol
```

Recovery after moving RAID-5 subdisks

When RAID-5 subdisks are moved and replaced, the new subdisks are marked as `STALE` in anticipation of recovery. If the volume is active, the `vxsd` command may be used to recover the volume. If the volume is not active, it is recovered when it is next started. The RAID-5 volume is degraded for the duration of the recovery operation.

Any failure in the stripes involved in the move makes the volume unusable. The RAID-5 volume can also become invalid if its parity becomes stale.

To avoid a volume becoming unusable, the `vxsd` command does not allow a subdisk move in the following situations:

- A stale subdisk occupies any of the same stripes as the subdisk being moved.
- The RAID-5 volume is stopped but was not shut down cleanly; that is, the parity is considered stale.
- The RAID-5 volume is active and has no valid log areas.

Only the third case can be overridden by using the `-o force` option.

Subdisks of RAID-5 volumes can also be split and joined by using the `vxsd split` command and the `vxsd join` command. These operations work the same way as those for mirrored volumes.

RAID-5 subdisk moves are performed in the same way as subdisk moves for other volume types, but without the penalty of degraded redundancy.

Unstartable RAID-5 volumes

When a RAID-5 volume is started, it can be in one of many states. After a normal system shutdown, the volume should be clean and require no recovery. However, if the volume was not closed, or was not unmounted before a crash, it can require recovery when it is started, before it can be made available.

Under normal conditions, volumes are started automatically after a reboot and any recovery takes place automatically or is done through the `vxrecover` command.

A RAID-5 volume is unusable if some part of the RAID-5 plex does not map the volume length in the following circumstances:

- The RAID-5 plex is sparse in relation to the RAID-5 volume length.
- The RAID-5 plex does not map a region where two subdisks have failed within a stripe, either because they are stale or because they are built on a failed disk.

When this occurs, the `vxvol start` command returns the following error message:

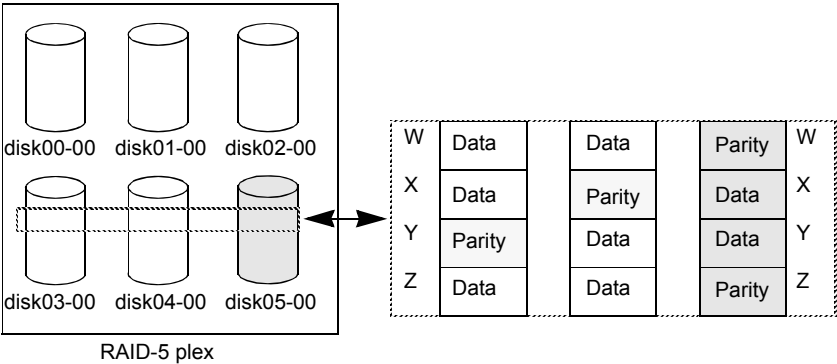
```
VxVM vxvol ERROR V-5-1-1236 Volume r5vol is not startable; RAID-5 plex
does not map entire volume length.
```

At this point, the contents of the RAID-5 volume are unusable.

Another possible way that a RAID-5 volume can become unstartable is if the parity is stale and a subdisk becomes detached or stale. This occurs because within the stripes that contain the failed subdisk, the parity stripe unit is invalid (because the parity is stale) and the stripe unit on the bad subdisk is also invalid.

[Figure 1-3](#) illustrates a RAID-5 volume that has become invalid due to stale parity and a failed subdisk.

Figure 1-3 Invalid RAID-5 volume



There are four stripes in the RAID-5 array. All parity is stale and subdisk `disk05-00` has failed. This makes stripes X and Y unusable because two failures have occurred within those stripes.

This qualifies as two failures within a stripe and prevents the use of the volume. In this case, the output display from the `vxvol start` command is as follows:

```
VxVM vxvol ERROR V-5-1-1237 Volume r5vol is not startable;  
some subdisks are unusable and the parity is stale.
```

This situation can be avoided by always using two or more RAID-5 log plexes in RAID-5 volumes. RAID-5 log plexes prevent the parity within the volume from becoming stale which prevents this situation.

See [“System failures”](#) on page 20.

Forcibly starting a RAID-5 volume with stale subdisks

You can start a volume even if subdisks are marked as stale: for example, if a stopped volume has stale parity and no RAID-5 logs, and a disk becomes detached and then reattached.

The subdisk is considered stale even though the data is not out of date (because the volume was in use when the subdisk was unavailable) and the RAID-5 volume is considered invalid. To prevent this case, always have multiple valid RAID-5 logs associated with the volume whenever possible.

To forcibly start a RAID-5 volume with stale subdisks

- ◆ Specify the `-f` option to the `vxvol start` command.

```
# vxvol [-g diskgroup] -f start r5vol
```

This causes all stale subdisks to be marked as non-stale. Marking takes place before the `start` operation evaluates the validity of the RAID-5 volume and what is needed to start it. You can mark individual subdisks as non-stale by using the following command:

```
# vxmend [-g diskgroup] fix unstale subdisk
```

If some subdisks are stale and need recovery, and if valid logs exist, the volume is enabled by placing it in the `ENABLED` kernel state and the volume is available for use during the subdisk recovery. Otherwise, the volume kernel state is set to `DETACHED` and it is not available during subdisk recovery. This is done because if the system were to crash or if the volume were ungracefully stopped while it was active, the parity becomes stale, making the volume unusable. If this is undesirable, the volume can be started with the `-o unsafe start` option.

Warning: The `-o unsafe start` option is considered dangerous, as it can make the contents of the volume unusable. It is therefore not recommended.

The volume state is set to `RECOVER`, and stale subdisks are restored. As the data on each subdisk becomes valid, the subdisk is marked as no longer stale. If the recovery of any subdisk fails, and if there are no valid logs, the volume start is aborted because the subdisk remains stale and a system crash makes the RAID-5 volume unusable. This can also be overridden by using the `-o unsafe start` option.

If the volume has valid logs, subdisk recovery failures are noted but they do not stop the start procedure.

When all subdisks have been recovered, the volume is placed in the `ENABLED` kernel state and marked as `ACTIVE`.

Recovering from an incomplete disk group move

If the system crashes or a subsystem fails while a disk group move, split or join operation is being performed, VxVM attempts either to reverse or to complete the operation when the system is restarted or the subsystem is repaired. Whether the operation is reversed or completed depends on how far it had progressed.

Automatic recovery depends on being able to import both the source and target disk groups. However, automatic recovery may not be possible if, for example, one of the disk groups has been imported on another host.

To recover from an incomplete disk group move

- 1 Use the `vxprint` command to examine the configuration of both disk groups. Objects in disk groups whose move is incomplete have their TUTILO fields set to MOVE.
- 2 Enter the following command to attempt completion of the move:

```
# vxdg recover sourcedg
```

This operation fails if one of the disk groups cannot be imported because it has been imported on another host or because it does not exist:

```
VxVM vxdg ERROR V-5-1-2907 diskgroup: Disk group does not exist
```

If the recovery fails, perform one of the following steps as appropriate.

- 3 If the disk group has been imported on another host, export it from that host, and import it on the current host. If all the required objects already exist in either the source or target disk group, use the following command to reset the MOVE flags in that disk group:

```
# vxdg -o clean recover diskgroup1
```

Use the following command on the other disk group to remove the objects that have TUTILO fields marked as MOVE:

```
# vxdg -o remove recover diskgroup2
```

- 4 If only one disk group is available to be imported, use the following command to reset the MOVE flags on this disk group:

```
# vxdg -o clean recover diskgroup
```

Recovery from failure of a DCO volume

The procedure to recover from the failure of a data change object (DCO) volume depends on the DCO version number.

For information about DCO versioning, see the *Veritas Volume Manager Administrator's Guide*.

Persistent FastResync uses a DCO volume to perform tracking of changed regions in a volume. If an error occurs while reading or writing a DCO volume, it is detached

and the `badlog` flag is set on the DCO. All further writes to the volume are not tracked by the DCO.

The following sample output from the `vxprint` command shows a complete volume with a detached DCO volume (the `TUTIL0` and `PUTIL0` fields are omitted for clarity):

TY	NAME	ASSOC	KSTATE	LENGTH	PLOFFS	STATE ...
dg	mydg	mydg	-	-	-	-
dm	mydg01	c4t50d0s2	-	35521408	-	-
dm	mydg02	c4t51d0s2	-	35521408	-	-
dm	mydg03	c4t52d0s2	-	35521408	-	FAILING
dm	mydg04	c4t53d0s2	-	35521408	-	FAILING
dm	mydg05	c4t54d0s2	-	35521408	-	-
v	SNAP-vol1	fsgen	ENABLED	204800	-	ACTIVE
pl	vol1-03	SNAP-vol1	ENABLED	204800	-	ACTIVE
sd	mydg05-01	vol1-03	ENABLED	204800	0	-
dc	SNAP-vol1_dco	SNAP-vol1	-	-	-	-
v	SNAP-vol1_dcl	gen	ENABLED	144	-	ACTIVE
pl	vol1_dcl-03	SNAP-vol1_dcl	ENABLED	144	-	ACTIVE
sd	mydg05-02	vol1_dcl-03	ENABLED	144	0	-
sp	vol1_snp	SNAP-vol1	-	-	-	-
v	vol1	fsgen	ENABLED	204800	-	ACTIVE
pl	vol1-01	vol1	ENABLED	204800	-	ACTIVE
sd	mydg01-01	vol1-01	ENABLED	204800	0	-
pl	vol1-02	vol1	ENABLED	204800	-	ACTIVE
sd	mydg02-01	vol1-01	ENABLED	204800	0	-
dc	vol1_dco	vol1	-	-	-	BADLOG
v	vol1_dcl	gen	DETACHED	144	-	DETACH
pl	vol1_dcl-01	vol1_dcl	ENABLED	144	-	ACTIVE
sd	mydg03-01	vol1_dcl-01	ENABLED	144	0	-
pl	vol1_dcl-02	vol1_dcl	DETACHED	144	-	IOFAIL
sd	mydg04-01	vol1_dcl-02	ENABLED	144	0	RELOCATE
sp	SNAP-vol1_snp	vol1	-	-	-	-

This output shows the mirrored volume, `vol1`, its snapshot volume, `SNAP-vol1`, and their respective DCOs, `vol1_dco` and `SNAP-vol1_dco`. The two disks, `mydg03` and `mydg04`, that hold the DCO plexes for the DCO volume, `vol1_dcl`, of `vol1` have failed. As a result, the DCO volume, `vol1_dcl`, of the volume, `vol1`, has been detached and the state of `vol1_dco` has been set to `BADLOG`. For future reference, note the entries for the snap objects, `vol1_snp` and `SNAP-vol1_snp`, that point to `vol1` and `SNAP-vol1` respectively.

You can use such output to deduce the name of a volume's DCO (in this example, `voll_dco`), or you can use the following `vxprint` command to display the name of a volume's DCO:

```
# vxprint [-g diskgroup] -F%dco_name volume
```

You can use the `vxprint` command to check if the `badlog` flag is set for the DCO of a volume as shown here:

```
# vxprint [-g diskgroup] -F%badlog dco_name
```

This command returns the value `on` if the `badlog` flag is set. For the example output, the command would take this form:

```
# vxprint -g mydg -F%badlog voll_dco
on
```

Use the following command to verify the version number of the DCO:

```
# vxprint [-g diskgroup] -F%version dco_name
```

This returns a value of 0 or 20. For the example output, the command would take this form:

```
# vxprint -g mydg -F%version voll_dco
```

The DCO version number determines the recovery procedure that you should use.

See [“Recovering a version 0 DCO volume”](#) on page 32.

See [“Recovering a version 20 DCO volume”](#) on page 35.

Recovering a version 0 DCO volume

To recover a version 0 DCO volume

- 1 Correct the problem that caused the I/O failure.
- 2 Use the following command to remove the `badlog` flag from the DCO:

```
# vxdco [-g diskgroup] -o force enable dco_name
```

For the example output, the command would take this form:

```
# vxdco -g mydg -o force enable voll_dco
```

The entry for `voll_dco` in the output from `vxprint` now looks like this:

```
dc voll_dco      voll      -      -      -      -
```

- 3 Restart the DCO volume using the following command:

```
# vxvol [-g diskgroup] start dco_log_vol
```

For the example output, the command would take this form:

```
# vxvol -g mydg start vol1_dcl
```

- 4 Use the `vxassist snapclear` command to clear the FastResync maps for the original volume and for all its snapshots. This ensures that potentially stale FastResync maps are not used when the snapshots are snapped back (a full resynchronization is performed). FastResync tracking is re-enabled for any subsequent snapshots of the volume.

Warning: You must use the `vxassist snapclear` command on all the snapshots of the volume after removing the `badlog` flag from the DCO. Otherwise, data may be lost or corrupted when the snapshots are snapped back.

If a volume and its snapshot volume are in the same disk group, the following command clears the FastResync maps for both volumes:

```
# vxassist [-g diskgroup] snapclear volume \
    snap_obj_to_snapshot
```

Here *snap_obj_to_snapshot* is the name of the snap object associated with *volume* that points to the snapshot volume.

For the example output, the command would take this form:

```
# vxassist -g mydg snapclear vol1 SNAP-vol1_snp
```

If a snapshot volume and the original volume are in different disk groups, you must perform a separate `snapclear` operation on each volume:

```
# vxassist -g diskgroup1 snapclear volume snap_obj_to_snapshot
# vxassist -g diskgroup2 snapclear snapvol snap_obj_to_volume
```

Here *snap_obj_to_volume* is the name of the snap object associated with the snapshot volume, *snapvol*, that points to the original volume.

For the example output, the commands would take this form if `SNAP-vol1` had been moved to the disk group, `snapdg`:

```
# vxassist -g mydg snapclear vol1 SNAP-vol1_snp
# vxassist -g snapdg snapclear SNAP-vol1 vol1_snp
```

- 5 To snap back the snapshot volume on which you performed a `snapclear`, use the following command (after using the `vxchg move` command to move the snapshot plex back to the original disk group, if necessary):

```
# vxplex -f [-g diskgroup] snapback volume snapvol_plex
```

For the example output, the command would take this form:

```
# vxplex -f -g mydg snapback vol1 vol1-03
```

You cannot use the `vxassist snapback` command because the `snapclear` operation removes the snapshot association information.

Recovering a version 20 DCO volume

To recover a version 20 DCO volume

- 1 Correct the problem that caused the I/O failure.
- 2 Use the `vxsnap` command to dissociate each full-sized instant snapshot volume that is associated with the volume:

```
# vxsnap [-g diskgroup] dis snapvol
```

For the example output, the command would take this form:

```
# vxsnap -g mydg dis SNAP-vol1
```

- 3 Unprepare the volume using the following command:

```
# vxsnap [-g diskgroup] unprepare volume
```

For the example output, the command would take this form:

```
# vxsnap -g mydg unprepare vol1
```

- 4 Start the volume using the `vxvol` command:

```
# vxvol [-g diskgroup] start volume
```

For the example output, the command would take this form:

```
# vxvol -g mydg start vol1
```

- 5 Prepare the volume again using the following command:

```
# vxsnap [-g diskgroup] prepare volume [ndcomirs=number] \  
[regionsize=size] [drl=yes|no|sequential] \  
[storage_attribute ...]
```

For the example output, the command might take this form:

```
# vxsnap -g mydg prepare vol1 ndcomirs=2 drl=yes
```

This adds a DCO volume with 2 plexes, and also enables DRL and FastResync (if licensed).

For full details of how to use the `vxsnap prepare` command, see the *Veritas Volume Manager Administrator's Guide* and the `vxsnap(1M)` manual page.

Recovering from instant snapshot failure

This chapter includes the following topics:

- [Recovering from the failure of vxsnap prepare](#)
- [Recovering from the failure of vxsnap make for full-sized instant snapshots](#)
- [Recovering from the failure of vxsnap make for break-off instant snapshots](#)
- [Recovering from the failure of vxsnap make for space-optimized instant snapshots](#)
- [Recovering from the failure of vxsnap restore](#)
- [Recovering from the failure of vxsnap reattach or refresh](#)
- [Recovering from copy-on-write failure](#)
- [Recovering from I/O errors during resynchronization](#)
- [Recovering from I/O failure on a DCO volume](#)

Recovering from the failure of vxsnap prepare

If a `vxsnap prepare` operation fails prematurely, the `vxprint` command may show the new DCO volume in the `INSTSNAPTMP` state. VxVM can usually recover the DCO volume without intervention. However, in certain situations, this recovery may not succeed. If this happens, the DCO volume must be deleted.

To recover from the failure of the vxsnap prepare command

- ◆ Type the following command:

```
# vxedit [-g diskgroup] rm DCO_volume
```

Alternatively, the DCO volume is removed automatically when the system is next restarted. When the DCO volume has been removed, run the vxsnap prepare command again.

Recovering from the failure of vxsnap make for full-sized instant snapshots

If a vxsnap make operation fails during the creation of a full-sized instant snapshot, the snapshot volume may go into the DISABLED state, be marked invalid, and be rendered unstartable. You can use the following command to verify that the inst_invalid flag is set to on:

```
# vxprint [-g diskgroup] -F%inst_invalid snapshot_volume
```

VxVM usually recovers the snapshot volume without intervention. However, in certain situations, this recovery may not succeed. If this happens, the DCO volume must be deleted.

To recover from the failure of the vxsnap make command for full-sized instant snapshots

- 1 Clear the snapshot volume's tuti10 field. Enter the following command:

```
# vxmend [-g diskgroup] clear tuti10 snapshot_volume
```

- 2 Unprepare the snapshot volume. Enter the following command:

```
# vxsnap [-g diskgroup] unprepare snapshot_volume
```

- 3 If the snapshot volume is in DISABLED state, start the snapshot volume. Enter the following command:

```
# vxvol [-g diskgroup] start snapshot_volume
```

- 4 Prepare the snapshot volume again for snapshot operations. Enter the following command:

```
# vxsnap [-g diskgroup] prepare snapshot_volume
```

- 5 Clear the volume's `tutil0` field (if it is set). Enter the following command.

```
# vxmend [-g diskgroup] clear tutil0 original_volume
```

Recovering from the failure of vxsnap make for break-off instant snapshots

If a `vxsnap make` operation fails during the creation of a third-mirror break-off instant snapshot, the snapshot volume may go into the `INSTSNAPTMP` state. VxVM can usually recover the snapshot volume without intervention. However, in certain situations, this recovery may not succeed. If this happens, the snapshot volume must be deleted.

To recover from the failure of the `vxsnap make` command for break-off instant snapshots

- ◆ Type the following command:

```
# vxedit [-g diskgroup] rm snapshot_volume
```

Alternatively, the snapshot volume is removed automatically when the system is next restarted.

Recovering from the failure of vxsnap make for space-optimized instant snapshots

If a `vxsnap make` operation fails during the creation of a space-optimized instant snapshot, the snapshot volume may go into the `INSTSNAPTMP` state. VxVM can usually recover the snapshot volume without intervention. However, in certain situations, this recovery may not succeed. If this happens, the snapshot volume must be deleted.

To recover from the failure of the vxsnap make command for space-optimized instant snapshots

- ◆ Type the following command:

```
# vxedit [-g diskgroup] rm snapshot_volume
```

Alternatively, the snapshot volume is removed automatically when the system is next restarted.

If the vxsnap make operation was being performed on a prepared cache object by specifying the cache attribute, the cache object remains intact after deleting the snapshot. If the cachesize attribute was used to specify a new cache object, the cache object does not exist after deleting the snapshot.

Recovering from the failure of vxsnap restore

If a vxsnap restore operation fails, the volume being restored may go into the DISABLED state.

To recover from the failure of the vxsnap restore command

- ◆ Type the following command:

```
# vxvol [-g diskgroup] start volume
```

Recovering from the failure of vxsnap reattach or refresh

If a vxsnap reattach or refresh operation fails, the volume being refreshed may go into the DISABLED state, be marked invalid and be rendered unstartable.

To recover from the failure of the vxsnap reattach or refresh commands

- 1 Use the following command to check that the inst_invalid flag is set to on:

```
# vxprint [-g diskgroup] -F%inst_invalid volume
```

- 2 Use the vxmend command to clear the volume's tuti10 field:

```
# vxmend [-g diskgroup] clear tuti10 volume
```

- 3 Use the `vxsnap` command to dissociate the volume from the snapshot hierarchy:

```
# vxsnap [-g diskgroup] dis volume
```

- 4 Use the following command to start the volume:

```
# vxvol [-g diskgroup] start volume
```

- 5 Re-run the failed `reattach` or `refresh` command.

This results in a full resynchronization of the volume. Alternatively, remove the snapshot volume and recreate it if required.

Recovering from copy-on-write failure

If an error is encountered while performing an internal resynchronization of a volume's snapshot, the snapshot volume goes into the `INVALID` state, and is made inaccessible for I/O and instant snapshot operations.

To recover from copy-on-write failure

- 1 Use the `vxsnap` command to dissociate the volume from the snapshot hierarchy:

```
# vxsnap [-g diskgroup] dis snapshot_volume
```

- 2 Unprepare the volume using the following command:

```
# vxsnap [-g diskgroup] unprepare snapshot_volume
```

- 3 Prepare the volume using the following command:

```
# vxsnap [-g diskgroup] prepare volume [ndcomirs=number] \
[regionsize=size] [drl=yes|no|sequential] \
[storage_attribute ...]
```

The volume can now be used again for snapshot operations.

Alternatively, you can remove the snapshot volume and recreate it if required.

Recovering from I/O errors during resynchronization

Snapshot resynchronization (started by `vxsnap syncstart`, or by specifying `sync=on` to `vxsnap`) stops if an I/O error occurs, and displays the following message on the system console:

```
VxVM vxsnap ERROR V-5-1-6840 Synchronization of the volume  
volume stopped due to I/O error
```

After correcting the source of the error, restart the resynchronization operation.

To recover from I/O errors during resynchronization

- ◆ Type the following command:

```
# vxsnap [-b] [-g diskgroup] syncstart volume
```

Recovering from I/O failure on a DCO volume

If an I/O failure occurs on a DCO volume, its FastResync maps and DRL log cannot be accessed, and the DCO volume is marked with the BADLOG flag. DRL logging and recovery, and instant snapshot operations are not possible with the volume until you recover its DCO volume.

If the I/O failure also affects the data volume, it must be recovered before its DCO volume can be recovered.

See [“Recovering a version 20 DCO volume”](#) on page 35.

Recovering from boot disk failure

This chapter includes the following topics:

- [VxVM and boot disk failure](#)
- [Possible root, swap, and usr configurations](#)
- [Booting from alternate boot disks](#)
- [Hot-relocation and boot disk failure](#)
- [Recovery from boot failure](#)
- [Repair of root or /usr file systems on mirrored volumes](#)
- [Replacement of boot disks](#)
- [Recovery by reinstallation](#)

VxVM and boot disk 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 the failure of the boot (root) disk. It also includes procedures for repairing the `root (/)` and `usr` file systems.

The procedures for recovering volumes and their data on boot disks differ from the procedures that are used for non-boot disks.

See [“About recovery from hardware failure”](#) on page 11.

Note: The examples assume that the boot (root) disk is configured on the device `c0t0d0s2`. Your system may be configured to use a different device.

Possible root, swap, and usr configurations

During installation, it is possible to set up a variety of configurations for the `root` (/) and `usr` file systems, and for `swap`.

The following are some of the possible configurations for the `/usr` file system:

- `usr` is a directory under `/` and no separate partition is allocated for it. In this case, `usr` becomes part of the `rootvol` volume when the root disk is encapsulated and put under Veritas Volume Manager control.
- `usr` is on a separate partition from the root partition on the root disk. In this case, a separate volume is created for the `usr` partition. `vxmirror` mirrors the `usr` volume on the destination disk.
- `usr` is on a disk other than the root disk. In this case, a volume is created for the `usr` partition only if you use VxVM to encapsulate the disk. Note that encapsulating the root disk and having mirrors of the root volume is ineffective in maintaining the availability of your system if the separate `usr` partition becomes inaccessible for any reason. For maximum availability of the system, it is recommended that you encapsulate both the root disk and the disk containing the `usr` partition, and have mirrors for the `usr`, `rootvol`, and `swapvol` volumes.

The `rootvol` volume must exist in the boot disk group.

There are other restrictions on the configuration of `rootvol` and `usr` volumes.

See the *Veritas Volume Manager Administrator's Guide*.

VxVM allows you to put `swap` partitions on any disk; it does not need an initial `swap` area during early phases of the boot process. By default, the Veritas Volume Manager installation chooses partition 0 on the selected root disk as the `root` partition, and partition 1 as the `swap` partition. However, it is possible to have the `swap` partition on a partition not located on the root disk. In such cases, you are advised to encapsulate that disk and create mirrors for the `swap` volume. If you do not do this, damage to the `swap` partition eventually causes the system to crash. It may be possible to boot the system, but having mirrors for the `swapvol` volume prevents system failures.

Booting from alternate boot disks

If the root disk is encapsulated and mirrored, you can use one of its mirrors to boot the system if the primary boot disk fails. This procedure differs between Solaris SPARC systems and Solaris x64 systems.

Booting from an alternate boot disk on Solaris SPARC systems

If the root disk is encapsulated and mirrored, you can use one of its mirrors to boot the system if the primary boot disk fails.

On a Solaris SPARC system, booting from an alternate boot disk requires that some EEPROM settings are changed.

See [“The boot process on Solaris SPARC systems”](#) on page 45.

The boot process on Solaris SPARC systems

A Solaris SPARC® system prompts for a boot command unless the `autoboot` flag has been set in the nonvolatile storage area used by the firmware. Machines with older PROMs have different prompts than that for the newer V2 and V3 versions. These newer versions of PROM are also known as OpenBoot PROMs (OBP). The `boot` command syntax for the newer types of PROMs is:

```
ok boot [OBP names] [filename] [boot-flags]
```

The OBP names specify the OpenBoot PROM designations. For example, on Desktop SPARC systems, the designation `sbus/esp@0,800000/sd@3,0:a` indicates a SCSI disk (`sd`) at target 3, lun 0 on the SCSI bus, with the `esp` host bus adapter plugged into slot 0.

You can use Veritas Volume Manager boot disk alias names instead of OBP names. Example aliases are `vx-rootdisk` or `vx-disk01`. To list the available boot devices, use the `devalias` command at the OpenBoot prompt.

The filename argument is the name of a file that contains the kernel. The default is `/kernel/unix` in the `root` partition. If necessary, you can specify another program (such as `/stand/diag`) by specifying the `-a` flag. (Some versions of the firmware allow the default filename to be saved in the nonvolatile storage area of the system.)

Warning: Do not boot a system running VxVM with rootability enabled using all the defaults presented by the `-a` flag.

See [“Restoring a copy of the system configuration file”](#) on page 59.

Boot flags are not interpreted by the `boot` program. The `boot` program passes all boot-flags to the file identified by filename.

See the `kadb` (1M) manual page.

See the `kernel` (1) manual page.

Defining a root disk mirror as bootable

After creating a `true` root disk mirror, you can make it available for booting.

To define a root disk mirror as bootable on a SPARC system

- 1 Check that the EEPROM variable `use-nvramrc?` is set to `true`. Enter the following command at the boot prompt:

```
ok printenv use-nvramrc?
```

If the system is up and running, enter the following command:

```
# eeprom use-nvramrc?
```

If `use-nvramrc?` is set to `true`, this allows the use of alternate boot disks.

If `use-nvramrc?` is set to `false`, the system fails to boot from the `devalias` and displays an error message such as the following:

```
Rebooting with command: boot vx-mirdisk
Boot device: /pci@1f,4000/scsi@3/disk@0,0 File and args:vx-
mirdisk
boot: cannot open vx-mirdisk
Enter filename [vx-mirdisk]:
```

- 2 Set the value of `use-nvramrc?` to `true`.

At the `ok` boot prompt, enter:

```
ok setenv use-nvramrc? true
```

If the system is up and running, use the following command:

```
# eeprom use-nvramrc?=true
```

- 3 Define an alternate boot disk by entering the following command at the `ok` boot prompt:

```
ok nvramrc=devalias vx-altboot_disk
```

where *altboot_disk* is the device name of an alternate disk from which the system can be booted.

If the system is already up and running, enter the following command to define an alternate boot disk:

```
# eeprom nvramrc="devalias vx-altboot_disk"
```

- 4 Use the `devalias` command at the boot prompt to discover the alternate disks from which the system may be booted:

```
ok devalias
```

Suitable mirrors of the root disk are listed with names of the form *vx-diskname*.

- 5 You should now be able to boot the system from an alternate boot disk, *vx-altboot_disk*, by entering the following command at the `ok` boot prompt:

```
ok boot vx-altboot_disk
```

If a selected disk contains a root mirror that is stale, `vxconfigd` displays an error stating that the mirror is unusable and lists any non-stale alternate bootable disks.

Booting from an alternate (mirror) boot disk on Solaris x64 systems

On a Solaris x64 system, the alternate boot disk is added to the GRUB boot menu when a boot disk is mirrored. If one root disk fails, the system stays up and lets you replace the disk. No reboot is required to perform this maintenance with internal SAS controllers and other CRU-type drives that are hot swappable. Replace the disk, then rescan with the `vxdtl enable` command to discover the replacement.

Alternatively, the `bootpath` can be redefined in the EEPROM without changing the GRUB configuration.

See [“The boot process on x64 systems”](#) on page 48.

Console access and the ability to select from the GRUB menu is required for the following procedure. The system should not have rebooted because of plex failure, but may have rebooted for other reasons.

To boot from an alternate boot disk on a Solaris x64 system

1 Select the "Alternate" GRUB menu entry:

```
title Solaris 10 11/06 s10x_u3wos_10 x64 <VxVM: Alternate Boot Disk>
    root (hd0,0,a)
    kernel /platform/i64pc/multiboot
    module /platform/i64pc/boot_archive.alt
```

2 After the system has booted, replace the failed drive.

See [“Replacement of boot disks”](#) on page 65.

The boot process on x64 systems

From Update 1 of the Solaris 10 OS, x64 systems are configured to use the GRUB boot loader. The devices from which a system may be booted are defined in the GRUB configuration file, `/boot/grub/menu.lst`. From the GRUB menu, you can select from the available bootable partitions that are known to the system. By default, the system will boot from the device that is defined by the `bootpath` variable in the EEPROM.

Defining root disk mirrors as bootable

After creating a root disk mirror, you can make it available for booting.

On Sun x64 systems, VxVM automatically creates a GRUB menu entry for the alternate boot disk when the boot disk is mirrored. During the booting process, select the alternate GRUB menu entry from the system console.

An alternate method is to change the 'default' GRUB menu setting in the `/boot/grub/menu.lst` file to select this entry automatically during the booting process.

See [“Manual GRUB procedure to set up an alternate boot disk”](#) on page 48.

Manual GRUB procedure to set up an alternate boot disk

You can set up the root disk mirror to boot from the primary and alternate boot disk. If one root disk fails, the system stays up and allows you to replace the disk. No reboot is required to perform this maintenance with internal SAS controllers and other CRU-type drives that are hot swappable. Replace the disk, then rescan with the `vxdctl enable` command to discover the replacement.

To set up an alternate boot disk using the manual GRUB procedure

- 1 List the device that corresponds to the root file system on the root disk mirror:

```
# ls -la /dev/dsk/altboot_disk_root_slice
```

Make a note of the PROM path for the root disk mirror that is displayed. For example:

```
# ls -al /dev/dsk/c0t3d0s0
lrwxrwxrwx 1 root root 60 Dec 13 13:17 /dev/dsk/c0t3d0s0 ->
../devices/pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@3,0:a
```

In this example, the PROM path is:

```
/pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@3,0:a
```

- 2 Set the alternate boot path in the EEPROM storage to the path of the boot disk mirror.

On Solaris x64, the EEPROM storage is simulated by the `/boot/solaris/bootenv.rc` file. You can update the EEPROM storage by either manually editing the `bootenv.rc` file or by issuing the **eeprom** command. For example:

```
# eeprom altbootpath="/pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@3,0:a
```

The EEPROM storage is now configured to boot from the primary boot disk:

```
bootpath=primary_boot_disk_path
altbootpath=alternate_boot_disk_path
```

- 3 Save the primary EEPROM file settings.

```
# cp /boot/solaris/bootenv.rc /boot/solaris/bootenv.pri
```

- 4 Edit the `/boot/solaris/bootenv.rc` to create an alternate EEPROM storage file.

Swap the device entries for `bootpath` and `altbootpath`:

```
bootpath=alternate_boot_disk_path
altbootpath=primary_boot_disk_path
```

The EEPROM storage is now configured to boot from the mirror boot disk.

- 5 Save the alternate EEPROM file settings:

```
# cp /boot/solaris/bootenv.rc /boot/solaris/bootenv.alt
```

6 Create the alternate boot archive:

```
# bootadm update-archive
```

Any changes to the `/boot` directory must be updated with this command, otherwise booting may fail.

7 Save the alternate boot archive:

```
# cp /platform/i64pc/boot_archive /platform/i64pc/boot_archive.alt
```

8 Restore the primary boot archive and EEPROM file:

```
# cp /boot/solaris/bootenv.pri /boot/solaris/bootenv.rc
```

9 Create the primary boot archive:

```
# bootadm update-archive
```

Any changes to the `/boot` directory must be updated with this command, otherwise booting may fail.

10 Locate the active GRUB menu in the `/boot/grub/menu.lst` file:

```
# bootadm list-menu
```

Typically the default GRUB menu entry is the entry that is automatically selected when the system boots up. **Default 0** refers to the first menu entry.

- 11** Edit the `/boot/grub/menu.lst` and duplicate the active menu entry. For example:

```
title Solaris 10 11/06 s10x_u3wos_10 X64 <Primary>
    root (hd0,0,a)
    kernel /platform/i64pc/multiboot
    module /platform/i64pc/boot_archive
```

Or:

```
title Solaris 10 11/06 s10x_u3wos_10 X64 <Primary>
    kernel /platform/i64pc/multiboot
    module /platform/i64pc/boot_archive
```

If the GRUB root statement is missing from the menu entry, root defaults to `root (hd0,0,a)`.

- 12** Edit the duplicated entry and change `boot_archive` to `boot_archive.alt`. Add Alternate to the title. For example:

```
title Solaris 10 11/06 s10x_u3wos_10 X64 <Alternate>
    root (hd0,0,a)
    kernel /platform/i64pc/multiboot
    module /platform/i64pc/boot_archive.alt
```

The GRUB menu title displays when the system is booting. You can select the alternate menu entry if the primary boot drive fails but was not replaced.

Booting from an alternate boot disk when the GRUB menu entry is missing

If the system has to be rebooted before the failed original primary drive has been replaced, you might be required to change the BIOS boot order to allow the system to get to the GRUB menu on the alternate boot disk. The drive to be booted needs to be "hd0" or the first BIOS-detectable disk. If the failed disk is still present and detected by the BIOS as "hd0," change the BIOS **boot disk priority** to select the surviving disk.

To boot from an alternate boot disk when the GRUB menu entry is missing

- 1 Boot into failsafe mode from the GRUB menu. This should be the only case where failsafe mode is used, as mounting and writing on one plex outside of VxVM when both plexes are attached corrupts the file system and makes both disks unusable.

Failsafe mode scans the disks to find operating system installations. The scan should only find one.

- 2 If the scan finds two operating system installations, you must pull the failed drive to proceed.

If the scan finds only one installation or after you pulled the failed drive, the failsafe mode discovers the good mirror and prompts you to mount that root file system on /a.

- 3 Enter **y** at the prompt.

- 4 List the device that corresponds to the root file system on the root disk mirror:

```
# ls -la /a/dev/dsk/altboot_disk_root_slice
```

Make a note of the PROM path for the root disk mirror that displays. For example:

```
# ls -al /a/dev/dsk/c0t3d0s0
lrwxrwxrwx 1 root root 60 Dec 13 13:17 /dev/dsk/c0t3d0s0 ->
../../../../devices/pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@3,0:a
/pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@3,0:a
```

- 5 Save the original `bootenv.rc` file:

```
# cp /a/boot/solaris/bootenv.rc /a/boot/solaris/bootenv.orig
```

- 6 Edit the `/a/boot/solaris/bootenv.rc` file.

On Solaris x64, the EEPROM storage is simulated by the `/boot/solaris/bootenv.rc` file. Set the `bootpath` in the EEPROM storage file to the path of the boot disk mirror. For example:

```
bootpath=/pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@3,0:a
```

- 7 Update boot archive

```
# bootadm update-archive -R /a
```

Any changes to the `/a/boot` directory must be updated with this command, otherwise booting may fail.

8 Unmount /a:

```
# unmount /a
```

9 Reboot the system with `init 6`:

```
# init 6
```

The system finds GRUB and boot the remaining mirror plex.

10 After the system has booted, replace the failed drive and sync the mirror plex.

See the *Veritas Volume Manager Troubleshooting Guide*.

11 Restore the original `bootenv.rc` file:

```
# cp /boot/solaris/bootenv.orig /boot/solaris/bootenv.rc
```

12 Configure the system for recovery from a Primary boot drive failure.

See “[Manual GRUB procedure to set up an alternate boot disk](#)” on page 48.

Hot-relocation and boot disk failure

If the boot (`root`) disk fails and it is mirrored, hot-relocation automatically attempts to replace the failed root disk mirror with a new mirror. To achieve this, hot-relocation uses a surviving mirror of the root disk to create a new mirror, either on a spare disk, or on a disk with sufficient free space. This ensures that there are always at least two mirrors of the root disk that can be used for booting. The hot-relocation daemon also calls the `vxbootsetup` utility to configure the disk with the new mirror as a bootable disk.

Hot-relocation can fail for a root disk if the boot disk group does not contain sufficient spare or free space to fit the volumes from the failed root disk. The `rootvol` and `swapvol` volumes require contiguous disk space. If the `root` volume and other volumes on the failed root disk cannot be relocated to the same new disk, each of these volumes may be relocated to different disks.

Mirrors of `rootvol` and `swapvol` volumes must be cylinder-aligned. This means that they can only be created on disks that have enough space to allow their subdisks to begin and end on cylinder boundaries. Hot-relocation fails to create the mirrors if these disks are not available.

Unrelocation of subdisks to a replacement boot disk

When a boot disk is encapsulated, the `root` file system and other system areas, such as the swap partition, are made into volumes. VxVM creates a private region using part of the existing swap area, which is usually located in the middle of the disk. However, when a disk is initialized as a VM disk, VxVM creates the private region at the beginning of the disk.

If a mirrored encapsulated boot disk fails, hot-relocation creates new copies of its subdisks on a spare disk. The name of the disk that failed and the offsets of its component subdisks are stored in the subdisk records as part of this process. After the failed boot disk is replaced with one that has the same storage capacity, it is “initialized” and added back to the disk group. `vxunreloc` can be run to move all the subdisks back to the disk. However, the difference of the disk layout between an initialized disk and an encapsulated disk affects the way the offset into a disk is calculated for each unrelocated subdisk. Use the `-f` option to `vxunreloc` to move the subdisks to the disk, but not to the same offsets. For this to be successful, the replacement disk should be at least 2 megabytes larger than the original boot disk.

`vxunreloc` makes the new disk bootable after it moves all the subdisks to the disk.

The system dump device is usually configured to be the swap partition of the root disk. Whenever a swap subdisk is moved (by hot-relocation, or using `vxunreloc`) from one disk to another, the dump device must be re-configured on the new disk.

You can use the `dumpadm` command to view and set the dump device.

See the `dumpadm(1M)` manual page.

Recovery from boot failure

While there are many types of failures that can prevent a system from booting, the same basic procedure can be taken to bring the system up. When a system fails to boot, you should first try to identify the failure by the evidence left behind on the screen and then attempt to repair the problem (for example, by turning on a drive that was accidentally powered off). If the problem is one that cannot be repaired (such as data errors on the boot disk), boot the system from an alternate boot disk that contains a mirror of the `root` volume, so that the damage can be repaired or the failing disk can be replaced.

The following are some possible failure modes:

- The boot device cannot be opened.
- The system cannot be booted from unusable or stale plexes.
- A UNIX partition is invalid.

- There are incorrect entries in `/etc/vfstab`.
- Configuration files are missing or damaged.

Boot device cannot be opened

Early in the boot process, immediately following system initialization, there may be messages similar to the following:

```
SCSI device 0,0 is not responding
Can't open boot device
```

The following are common causes for the system PROM being unable to read the boot program from the boot drive:

- The boot disk is not powered on.
- The SCSI bus is not terminated.
- There is a controller failure of some sort.
- A disk is failing and locking the bus, preventing any disks from identifying themselves to the controller, and making the controller assume that there are no disks attached.

The first step in diagnosing this problem is to check carefully that everything on the SCSI bus is in order. If disks are powered off or the bus is unterminated, correct the problem and reboot the system. If one of the disks has failed, remove the disk from the bus and replace it.

If no hardware problems are found, the error is probably due to data errors on the boot disk. In order to repair this problem, attempt to boot the system from an alternate boot disk (containing a mirror of the root volume). If you are unable to boot from an alternate boot disk, there is still some type of hardware problem. Similarly, if switching the failed boot disk with an alternate boot disk fails to allow the system to boot, this also indicates hardware problems.

Cannot boot from unusable or stale plexes

If a disk is unavailable when the system is running, any mirrors of volumes that reside on that disk become stale. This means that the data on that disk is inconsistent relative to the other mirrors of that volume. During the boot process, the system accesses only one copy of the `root` volume (the copy on the boot disk) until a complete configuration for this volume can be obtained.

If it turns out that the plex of this volume that was used for booting is stale, the system must be rebooted from an alternate boot disk that contains non-stale plexes. This problem can occur, for example, if the system was booted from one

of the disks made bootable by VxVM with the original boot disk turned off. The system boots normally, but the plexes that reside on the unpowered disk are stale. If the system reboots from the original boot disk with the disk turned back on, the system boots using that stale plex.

Another possible problem can occur if errors in the Veritas Volume Manager headers on the boot disk prevent VxVM from properly identifying the disk. In this case, VxVM does not know the name of that disk. This is a problem because plexes are associated with disk names, so any plexes on the unidentified disk are unusable.

A problem can also occur if the root disk has a failure that affects the root volume plex. At the next boot attempt, the system still expects to use the failed root plex for booting. If the root disk was mirrored at the time of the failure, an alternate root disk (with a valid root plex) can be specified for booting.

If any of these situations occur, the configuration daemon, `vxconfigd`, notes it when it is configuring the system as part of the `init` processing of the boot sequence. `vxconfigd` displays a message describing the error and what can be done about it, and then halts the system. For example, if the plex `rootvol-01` of the root volume `rootvol` on disk `rootdisk` is stale, `vxconfigd` may display this message:

```
VxVM vxconfigd ERROR V-5-1-1049: System boot disk does not have
a valid root plex
Please boot from one of the following disks:
Disk: disk01                Device: c0t1d0s2
vxvm:vxconfigd:      Error: System startup failed
The system is down.
```

This informs the administrator that the alternate boot disk named `disk01` contains a usable copy of the root plex and should be used for booting. When this message is displayed, reboot the system from the alternate boot disk.

See [“Booting from alternate boot disks”](#) on page 45.

Once the system has booted, the exact problem needs to be determined. If the plexes on the boot disk were simply stale, they are caught up automatically as the system comes up. If, on the other hand, there was a problem with the private area on the disk or the disk failed, you need to re-add or replace the disk.

If the plexes on the boot disk are unavailable, you should receive mail from Veritas Volume Manager utilities describing the problem. Another way to determine the problem is by listing the disks with the `vxdisk` utility. If the problem is a failure in the private area of root disk (such as due to media failures or accidentally overwriting the Veritas Volume Manager private region on the disk), `vxdisk list` shows a display such as this:

DEVICE	TYPE	DISK	GROUP	STATUS
-	-	rootdisk	bootdg	failed was: c0t3d0s2
c0t1d0s2	sliced	disk01	bootdg	ONLINE

Invalid UNIX partition

Once the boot program has loaded, it attempts to access the boot disk through the normal UNIX partition information. If this information is damaged, the boot program fails with an error such as:

```
File just loaded does not appear to be executable
```

If this message appears during the boot attempt, the system should be booted from an alternate boot disk. While booting, most disk drivers display errors on the console about the invalid UNIX partition information on the failing disk. The messages are similar to this:

```
WARNING: unable to read label  
WARNING: corrupt label_sdo
```

This indicates that the failure was due to an invalid disk partition. You can attempt to re-add the disk.

See [“Re-adding a failed boot disk”](#) on page 66.

However, if the reattach fails, then the disk needs to be replaced.

See [“Replacing a failed boot disk”](#) on page 67.

Incorrect entries in /etc/vfstab

When the root disk is encapsulated and put under Veritas Volume Manager control, as part of the normal encapsulation process, volumes are created for all of the partitions on the disk. VxVM modifies the `/etc/vfstab` to use the corresponding volumes instead of the disk partitions. Care should be taken while editing the `/etc/vfstab` file manually, and you should always make a backup copy before committing any changes to it. The most important entries are those corresponding to `/` and `/usr`. The `vfstab` that existed prior to Veritas Volume Manager installation is saved in `/etc/vfstab.prevm`.

Damaged root (/) entry in /etc/vfstab

If the entry in `/etc/vfstab` for the `root` file system (`/`) is lost or is incorrect, the system boots in single-user mode. Messages similar to the following are displayed on booting the system:

```
INIT: Cannot create /var/adm/utmp or /var/adm/utmpx
INIT: failed write of utmpx entry: " "
```

It is recommended that you first run `fsck` on the root partition as shown in this example:

```
# fsck -F ufs /dev/rdisk/c0t0d0s0
```

At this point in the boot process, `/` is mounted read-only, not read/write. Since the entry in `/etc/vfstab` was either incorrect or deleted, mount `/` as read/write manually, using this command:

```
# mount -o remount /dev/vx/dsk/rootvol /
```

After mounting `/` as read/write, exit the shell. The system prompts for a new run level. For multi-user mode, enter run level 3:

```
ENTER RUN LEVEL (0-6,s or S): 3
```

Restore the entry in `/etc/vfstab` for `/` after the system boots.

Damaged `/usr` entry in `/etc/vfstab`

The `/etc/vfstab` file has an entry for `/usr` only if `/usr` is located on a separate disk partition. After encapsulation of the disk containing the `/usr` partition, VxVM changes the entry in `/etc/vfstab` to use the corresponding volume.

In the event of loss of the entry for `/usr` from `/etc/vfstab`, the system cannot be booted (even if you have mirrors of the `/usr` volume). In this case, boot the system from the CD-ROM and restore `/etc/vfstab`.

To repair a damaged `/usr` entry in `/etc/vfstab`

- 1 Boot the operating system into single-user mode from its installation CD-ROM using the following command at the boot prompt:

```
ok boot cdrom -s
```

- 2 Mount `/dev/dsk/c0t0d0s0` on a suitable mount point such as `/a` or `/mnt`:

```
# mount /dev/dsk/c0t0d0s0 /a
```

- 3 Edit `/a/etc/vfstab`, and ensure that there is an entry for the `/usr` file system, such as the following:

```
/dev/vx/dsk/usr /dev/vx/rdsk/usr /usr ufs 1 yes -
```

- 4 Shut down and reboot the system from the same `root` partition on which the `vfstab` file was restored.

Missing or damaged configuration files

VxVM no longer maintains entries for tunables in `/etc/system` as was the case for VxVM 3.2 and earlier releases. All entries for Veritas Volume Manager device driver tunables are now contained in files named `/kernel/drv/vx*.conf`, such as `/kernel/drv/vxio.conf`.

See the *Veritas Volume Manager Administrator's Guide*.

Warning: If you need to modify configuration files such as `/etc/system`, make a copy of the file in the `root` file system before editing it.

Restoring a copy of the system configuration file

If the system configuration file is damaged, the system can be booted if the file is restored from a saved copy.

To restore a copy of the system configuration file

- 1 Boot the system with the following command:

```
ok boot -a
```

- 2 Press Return to accept the default for all prompts except the following:

- The default pathname for the kernel program, `/kernel/unix`, may not be appropriate for your system's architecture. If this is so, enter the correct pathname, such as `/platform/sun4u/kernel/unix`, at the following prompt:

```
Enter filename [/kernel/unix]: /platform/sun4u/kernel/unix
```

- Enter the name of the saved `system` file, such as `/etc/system.save` at the following prompt:

```
Name of system file [/etc/system]: /etc/system.save
```

- Enter `/pseudo/vxio@0:0` as the physical name of the root device at the following prompt:

```
Enter physical name of root device  
[...]:/pseudo/vxio@0:0
```

Restoring `/etc/system` if a copy is not available on the root disk

If `/etc/system` is damaged or missing, and a saved copy of this file is not available on the root disk, the system cannot be booted with the Veritas Volume Manager rootability feature turned on.

The following procedure assumes the device name of the root disk to be `c0t0d0s2`, and that the `root (/)` file system is on partition `s0`.

To boot the system without Veritas Volume Manager rootability and restore the configuration files

- 1 Boot the operating system into single-user mode from its installation CD-ROM using the following command at the boot prompt:

```
ok boot cdrom -s
```

- 2 Mount `/dev/dsk/c0t0d0s0` on a suitable mount point such as `/a` or `/mnt`:

```
# mount /dev/dsk/c0t0d0s0 /a
```

- 3 If a backup copy of `/etc/system` is available, restore this as the file `/a/etc/system`. If a backup copy is not available, create a new `/a/etc/system` file. Ensure that `/a/etc/system` contains the following entries that are required by VxVM:

```
set vxio:vol_rootdev_is_volume=1
forceload: drv/driver
...
forceload: drv/vxio
forceload: drv/vxspec
forceload: drv/vxdmp
rootdev: /pseudo/vxio@0:0
```

Lines of the form `forceload: drv/driver` are used to forcibly load the drivers that are required for the `root` mirror disks. Example driver names are `pci`, `sd`, `ssd`, `dad` and `ide`. To find out the names of the drivers, use the `ls` command to obtain a long listing of the special files that correspond to the devices used for the root disk, for example:

```
# ls -al /dev/dsk/c0t0d0s2
```

This produces output similar to the following (with irrelevant detail removed):

```
lrwxrwxrwx ... /dev/dsk/c0t0d0s2 ->
.../..../devices/pci@1f,0/pci@1/pci@1/SUNW,isptwo@4/sd@0,0:c
```

This example would require lines to force load both the `pci` and the `sd` drivers:

```
forceload: drv/pci
forceload: drv/sd
```

- 4 Shut down and reboot the system from the same `root` partition on which the configuration files were restored.

Repair of root or /usr file systems on mirrored volumes

If the `root` or `/usr` file system is defined on a mirrored volume, errors in the partition that underlies one of the mirrors can result in data corruption or system errors at boot time (when VxVM is started and assumes that the mirrors are synchronized).

The following workarounds exist for this situation:

- Mount one plex of the `root` or `/usr` file system, repair it, unmount it, and use `dd` to copy the fixed plex to all other plexes. This procedure is not recommended as it can be error prone.
- Restore the system from a valid backup. This procedure does not require the operating system to be re-installed from the base CD-ROM. It provides a simple, efficient, and reliable means of recovery when both the root disk and its mirror are damaged.

Recovering a root disk and root mirror from a backup

This procedure assumes that you have the following resources available:

- A listing of the partition table for the original root disk before you encapsulated it.
- A current full backup of all the file systems on the original root disk that was under Veritas Volume Manager control. If the `root` file system is of type `ufs`, you can back it up using the `ufsdump` command.
See the `ufsdump(1M)` manual page.
- A new boot disk installed to replace the original failed boot disk if the original boot disk was physically damaged.

This procedure requires the reinstallation of the root disk. To prevent the loss of data on disks not involved in the reinstallation, only involve the root disk in the reinstallation procedure.

Several of the automatic options for installation access disks other than the root disk without requiring confirmation from the administrator. Therefore, disconnect all other disks containing volumes from the system prior to starting this procedure. This will ensure that these disks are unaffected by the reinstallation. Reconnect the disks after completing the procedure.

The procedure assumes the device name of the new root disk to be `c0t0d0s2`, and that you need to recover both the `root (/)` file system on partition `s0`, and the `/usr` file system on partition `s6`.

To recover a root disk and root mirror from a backup

- 1 Boot the operating system into single-user mode from its installation CD-ROM using the following command at the boot prompt:

```
ok boot cdrom -s
```

- 2 Use the `format` command to create partitions on the new root disk (`c0t0d0s2`). These should be identical in size to those on the original root disk before encapsulation unless you are using this procedure to change their sizes. If you change the size of the partitions, ensure that they are large enough to store the data that is restored to them.

See the `format(1M)` manual page.

A maximum of five partitions may be created for file systems or swap areas as encapsulation reserves two partitions for Veritas Volume Manager `private` and `public` regions.

- 3 Use the `mkfs` command to make new file systems on the `root` and `usr` partitions. For example, to make a `ufs` file system on the `root` partition, enter:

```
# mkfs -F ufs /dev/rdisk/c0t0d0s0
```

See the `mkfs(1M)` manual page.

See the `mkfs_ufs(1M)` manual page.

- 4 Mount `/dev/dsk/c0t0d0s0` on a suitable mount point such as `/a` or `/mnt`:

```
# mount /dev/dsk/c0t0d0s0 /a
```

- 5 Restore the `root` file system from tape into the `/a` directory hierarchy. For example, if you used `ufsdump` to back up the file system, use the `ufsrestore` command to restore it.

See the `ufsrestore(1M)` manual page.

- 6 Use the `installboot` command to install a bootblock device on `/a`.

- 7 If the `/usr` file system is separate from the root file system, use the `mkdir` command to create a suitable mount point, such as `/a/usr/`, and mount `/dev/dsk/c0t0d0s6` on it:

```
# mkdir -p /a/usr
# mount /dev/dsk/c0t0d0s6 /a/usr
```

- 8 If the /usr file system is separate from the root file system, restore the /usr file system from tape into the /a/usr directory hierarchy.
- 9 Disable startup of VxVM by modifying files in the restored root file system.
- 10 Create the file /a/etc/vx/reconfig.d/state.d/install-db to prevent the configuration daemon, vxconfigd, from starting:

```
# touch /a/etc/vx/reconfig.d/state.d/install-db
```

- 11 Copy /a/etc/system to a backup file such as /a/etc/system.old.
- 12 Comment out the following lines from /a/etc/system by putting a * character in front of them:

```
set vxio:vol_rootdev_is_volume=1
rootdev:/pseudo/vxio@0:0
```

These lines should then read:

```
* set vxio:vol_rootdev_is_volume=1
* rootdev:/pseudo/vxio@0:0
```

- 13 Copy /a/etc/vfstab to a backup file such as /a/etc/vfstab.old.
- 14 Edit /a/etc/vfstab, and replace the volume device names (beginning with /dev/vx/dsk) for the / and /usr file system entries with their standard disk devices, /dev/dsk/c0t0d0s0 and /dev/dsk/c0t0d0s6. For example, replace the following lines:

```
/dev/vx/dsk/rootvol /dev/vx/rdisk/rootvol / ufs 1 no -
/dev/vx/dsk/usrvol /dev/vx/rdisk/usrvol /usr ufs 1 yes -
```

with this line:

```
/dev/dsk/c0t0d0s0 /dev/rdisk/c0t0d0s0 / ufs 1 no -
/dev/dsk/c0t0d0s6 /dev/rdisk/c0t0d0s6 /usr ufs 1 yes -
```

- 15 Remove /a/dev/vx/dsk/bootdg and /a/dev/vx/rdisk/bootdg:

```
# rm /a/dev/vx/dsk/bootdg
# rm /a/dev/vx/rdisk/bootdg
```

- 16 Shut down the system cleanly using the init 0 command, and reboot from the new root disk. The system comes up thinking that VxVM is not installed.

- 17 If there are only root disk mirrors in the old boot disk group, remove any volumes that were associated with the encapsulated root disk (for example, `rootvol`, `swapvol` and `usrvol`) from the `/dev/vx/dsk/bootdg` and `/dev/vx/rdsk/bootdg` directories.

- 18 If there are other disks in the old boot disk group that are not used as root disk mirrors, remove files involved with the installation that are no longer needed:

```
# rm -r /etc/vx/reconfig.d/state.d/install-db
```

Start the Veritas Volume Manager I/O daemons:

```
# vxiod set 10
```

Start the Veritas Volume Manager configuration daemon in disabled mode:

```
# vxconfigd -m disable
```

Initialize the `volboot` file:

```
# vxdctl init
```

Enable the old boot disk group excluding the root disk that VxVM interprets as failed::

```
# vxdctl enable
```

Use the `vxedit` command (or the Veritas Enterprise Administrator (VEA)) to remove the old root disk volumes and the root disk itself from Veritas Volume Manager control.

- 19 Use the `vxdiskadm` command to encapsulate the new root disk and initialize any disks that are to serve as root disk mirrors. After the required reboot, mirror the root disk onto the root disk mirrors.

Replacement of boot disks

Data that is not critical for booting the system is only accessed by VxVM after the system is fully operational, so it does not have to be located in specific areas. VxVM can find it. However, boot-critical data must be placed in specific areas on the bootable disks for the boot process to find it.

On some systems, the controller-specific actions performed by the disk controller in the process and the system BIOS constrain the location of this critical data.

If a boot disk fails, one of the following procedures can be used to correct the problem:

- If the errors are transient or correctable, re-use the same disk. This is known as re-adding a disk. In some cases, reformatting a failed disk or performing a surface analysis to rebuild the alternate-sector mappings are sufficient to make a disk usable for re-addition.
See [“Re-adding a failed boot disk ”](#) on page 66.
- If the disk has failed completely, replace it.
See [“Replacing a failed boot disk”](#) on page 67.

Re-adding a failed boot disk

Re-adding a disk is the same procedure as replacing the disk, except that the same physical disk is used. Normally, a disk that needs to be re-added has been detached. This means that VxVM has detected the disk failure and has ceased to access the disk.

For example, consider a system that has two disks, `disk01` and `disk02`, which are normally mapped into the system configuration during boot as disks `c0t0d0s2` and `c0t1d0s2`, respectively. A failure has caused `disk01` to become detached. This can be confirmed by listing the disks with the `vxdisk` utility with this command:

```
# vxdisk list
```

`vxdisk` displays this (example) list:

DEVICE	TYPE	DISK	GROUP	STATUS
c0t0d0s2	sliced	-	-	error
c0t1d0s2	sliced	disk02	bootdg	online
-	-	disk01	bootdg	failed was:c0t0d0s2

Note that the disk `disk01` has no device associated with it, and has a status of `failed` with an indication of the device that it was detached from. It is also possible for the device (such as `c0t0d0s2` in the example) not to be listed at all should the disk fail completely.

In some cases, the `vxdisk list` output can differ. For example, if the boot disk has uncorrectable failures associated with the UNIX partition table, a missing root partition cannot be corrected but there are no errors in the Veritas Volume Manager private area. The `vxdisk list` command displays a listing such as this:

DEVICE	TYPE	DISK	GROUP	STATUS
c0t0d0s2	sliced	disk01	bootdg	online
c0t1d0s2	sliced	disk02	bootdg	online

However, because the error was not correctable, the disk is viewed as failed. In such a case, remove the association between the failing device and its disk name using the `vxdiskadm` “Remove a disk for replacement” menu item.

See the `vxdiskadm` (1M) manual page.

You can then perform any special procedures to correct the problem, such as reformatting the device.

To re-add a failed boot disk

- 1 Select the `vxdiskadm` “Replace a failed or removed disk” menu item to replace the disk, and specify the same device as the replacement. For example, you might replace `disk01` with the device `c0t0d0s2`.
- 2 If hot-relocation is enabled when a mirrored boot disk fails, an attempt is made to create a new mirror and remove the failed subdisks from the failing boot disk. If a re-add succeeds after a successful hot-relocation, the `root` and other volumes affected by the disk failure no longer exist on the re-added disk. Run `vxunreloc` to move the hot-relocated subdisks back to the newly replaced disk.

Replacing a failed boot disk

The replacement disk for a failed boot disk must have at least as much storage capacity as was in use on the disk being replaced. It must be large enough to accommodate all subdisks of the original disk at their current disk offsets.

To estimate the size of the replacement disk, use this command:

```
# vxprint [-g diskgroup] -st -e 'sd_disk='
diskname
''
```

where `diskname` is the name of the disk that failed or of one of its mirrors.

The following is sample output from running this command:

```
# vxprint -g rtdg -st -e 'sd_disk="rtdg01"'
Disk group: rtdg
```

SD NAME	PLEX	DISK	DISKOFFS	LENGTH	[COL/]OFF	DEVICE	MODE
...							
sd rtdg01-01	swapvol-01	rtdg01	0	1045296	0	c0t0d0	ENA
sd rtdg01-02	rootvol-01	rtdg01	1045296	16751952	0	c0t0d0	ENA

From the resulting output, add the `DISKOFFS` and `LENGTH` values for the last subdisk listed for the disk. This size is in 512-byte sectors. Divide this number by 2 for the

size in kilobytes. In this example, the `DISKOFFS` and `LENGTH` values for the subdisk `rt dg01-02` are 1,045,296 and 16,751,952, so the disk size is $(1,045,296 + 16,751,952)/2$, which equals 8,898,624 kilobytes or approximately 8.5 gigabytes.

Note: Disk sizes reported by manufacturers usually represent the unformatted capacity of disks. Also, most manufacturers use the terms megabyte and gigabyte to mean a million (1,000,000) and a billion (1,000,000,000) bytes respectively, rather than the usual meaning of 1,048,576 and 1,073,741,824 bytes.

To replace a failed boot disk:

- 1 Boot the system from an alternate boot disk.
See [“Booting from alternate boot disks”](#) on page 45.
- 2 Remove the association between the failing device and its disk name using the “Remove a disk for replacement” function of `vxdiskadm`.
See the `vxdiskadm (1M)` manual page.
- 3 Shut down the system and replace the failed hardware.
- 4 After rebooting from the alternate boot disk, use the `vxdiskadm` “Replace a failed or removed disk” menu item to notify VxVM that you have replaced the failed disk.

Warning: If the replacement disk was previously an encapsulated root disk under VxVM control, select to reorganize the disk when prompted. Otherwise, the disk is left with an invalid VTOC that renders the system unbootable. Ensure that you have made at least one valid copy of any existing data on the disk.

- 5 Use `vxdiskadm` to mirror the alternate boot disk to the replacement boot disk.
- 6 When the volumes on the boot disk have been restored, shut down the system, and test that the system can be booted from the replacement boot disk.

Recovery by reinstallation

Reinstallation is necessary if all copies of your boot (`root`) disk are damaged, or if certain critical files are lost due to file system damage.

If these types of failures occur, attempt to preserve as much of the original VxVM configuration as possible. Any volumes that are not directly involved in the failure

do not need to be reconfigured. You do not have to reconfigure any volumes that are preserved.

General reinstallation information

This section describes procedures used to reinstall VxVM and preserve as much of the original configuration as possible after a failure.

Warning: You should assume that reinstallation can potentially destroy the contents of any disks that are touched by the reinstallation process

All VxVM-related information is removed during reinstallation. Data removed includes data in private areas on removed disks that contain the disk identifier and copies of the VxVM configuration. The removal of this information makes the disk unusable as a VM disk.

The system root disk is always involved in reinstallation. Other disks can also be involved. If the root disk was placed under VxVM control, that disk and any volumes or mirrors on it are lost during reinstallation. Any other disks that are involved in the reinstallation, or that are removed and replaced, can lose VxVM configuration data (including volumes and mirrors).

If a disk, including the root disk, is not under VxVM control prior to the failure, no VxVM configuration data is lost at reinstallation.

Although it simplifies the recovery process after reinstallation, not having the root disk under Veritas Volume Manager control increases the possibility of a reinstallation being necessary. By having the root disk under VxVM control and creating mirrors of the root disk contents, you can eliminate many of the problems that require system reinstallation.

When reinstallation is necessary, the only volumes saved are those that reside on, or have copies on, disks that are not directly involved with the failure and reinstallation. Any volumes on the root disk and other disks involved with the failure or reinstallation are lost during reinstallation. If backup copies of these volumes are available, the volumes can be restored after reinstallation.

Reinstalling the system and recovering VxVM

To reinstall the system and recover the Veritas Volume Manager configuration, the following steps are required:

- Replace any failed disks or other hardware, and detach any disks not involved in the reinstallation.

See [“Prepare the system for reinstallation”](#) on page 70.

- Reinstall the base system and any other unrelated Volume Manager packages. See [“Reinstall the operating system”](#) on page 70.
- Add the Volume Manager package, but do not execute the `vxinstall` command. See [“Reinstalling Veritas Volume Manager”](#) on page 71.
- Recover the Veritas Volume Manager configuration. See [“Recovering the Veritas Volume Manager configuration”](#) on page 71.
- Restore any information in volumes affected by the failure or reinstallation, and recreate system volumes (`rootvol`, `swapvol`, `usr`, and other system volumes). See [“Cleaning up the system configuration”](#) on page 72.

Prepare the system for reinstallation

To prevent the loss of data on disks not involved in the reinstallation, involve only the root disk and any other disks that contain portions of the operating system in the reinstallation procedure. For example, if the `/usr` file system is configured on a separate disk, leave that disk connected. Several of the automatic options for installation access disks other than the root disk without requiring confirmation from the administrator.

Disconnect all other disks containing volumes (or other data that should be preserved) prior to reinstalling the operating system. For example, if you originally installed the operating system with the `home` file system on a separate disk, disconnect that disk to ensure that the `home` file system remains intact.

Reinstall the operating system

Once any failed or failing disks have been replaced and disks not involved with the reinstallation have been detached, reinstall the operating system as described in your operating system documentation. Install the operating system prior to installing VxVM.

Ensure that no disks other than the root disk are accessed in any way while the operating system installation is in progress. If anything is written on a disk other than the root disk, the Veritas Volume Manager configuration on that disk may be destroyed.

Note: During reinstallation, you can change the system’s host name (or host ID). It is recommended that you keep the existing host name, as this is assumed by the procedures in the following sections.

Reinstalling Veritas Volume Manager

To reinstall Veritas Volume Manager

- 1 Reinstall the Veritas software from the installation disks.

See the *Installation Guide*.

Warning: Do not use `vxinstall` to initialize VxVM.

- 2 If required, use the `vxlicinst` command to install the Veritas Volume Manager license key.

See the `vxlicinst(1)` manual page.

Recovering the Veritas Volume Manager configuration

Once the Veritas Volume Manager packages have been loaded, and you have installed the license for VxVM, recover the Veritas Volume Manager configuration.

To recover the Veritas Volume Manager configuration

- 1 Touch `/etc/vx/reconfig.d/state.d/install-db`.
- 2 Shut down the system.
- 3 Reattach the disks that were removed from the system.
- 4 Reboot the system.
- 5 When the system comes up, bring the system to single-user mode using the following command:

```
# exec init S
```

- 6 When prompted, enter the password and press Return to continue.
- 7 Remove files involved with installation that were created when you loaded VxVM, but which are no longer needed, using the following command:

```
# rm -rf /etc/vx/reconfig.d/state.d/install-db
```

- 8 Start some Veritas Volume Manager I/O daemons using the following command:

```
# vxiod set 10
```

- 9 Start the Veritas Volume Manager configuration daemon, `vxconfigd`, in disabled mode using the following command:

```
# vxconfigd -m disable
```

- 10 Initialize the `vxconfigd` daemon using the following command:

```
# vxdctl init
```

- 11 Initialize the DMP subsystem using the following command:

```
# vxdctl initdmp
```

- 12 Enable `vxconfigd` using the following command:

```
# vxdctl enable
```

The configuration preserved on the disks not involved with the reinstallation has now been recovered. However, because the root disk has been reinstalled, it does not appear to VxVM as a VM disk. The configuration of the preserved disks does not include the root disk as part of the VxVM configuration.

If the root disk of your system and any other disks involved in the reinstallation were not under VxVM control at the time of failure and reinstallation, then the reconfiguration is complete at this point.

If the root disk (or another disk) was involved with the reinstallation, any volumes or mirrors on that disk (or other disks no longer attached to the system) are now inaccessible. If a volume had only one plex contained on a disk that was reinstalled, removed, or replaced, then the data in that volume is lost and must be restored from backup.

Cleaning up the system configuration

After reinstalling VxVM, you must clean up the system configuration.

To clean up the system configuration

- 1 Remove any volumes associated with rootability. This must be done if the root disk (and any other disk involved in the system boot process) was under Veritas Volume Manager control.

The following volumes must be removed:

<code>rootvol</code>	Contains the <code>root</code> file system.
<code>swapvol</code>	Contains the swap area.
<code>standvol</code> (if present)	Contains the <code>stand</code> file system.
<code>usr</code> (if present)	contains the <code>usr</code> file system.

To remove the root volume, use the `vxedit` command:

```
# vxedit -fr rm rootvol
```

Repeat this command, using `swapvol`, `standvol` and `usr` in place of `rootvol`, to remove the `swap`, `stand`, and `usr` volumes.

- 2 After completing the rootability cleanup, you must determine which volumes need to be restored from backup. The volumes to be restored include those with all mirrors (all copies of the volume) residing on disks that have been reinstalled or removed. These volumes are invalid and must be removed, recreated, and restored from backup. If only some mirrors of a volume exist on reinstalled or removed disks, these mirrors must be removed. The mirrors can be re-added later.

Establish which VM disks have been removed or reinstalled using the following command:

```
# vxdisk list
```

This displays a list of system disk devices and the status of these devices. For example, for a reinstalled system with three disks and a reinstalled root disk, the output of the `vxdisk list` command is similar to this:

DEVICE	TYPE	DISK	GROUP	STATUS
c0t0d0s2	sliced	-	-	error
c0t1d0s2	sliced	disk02	mydg	online
c0t2d0s2	sliced	disk03	mydg	online
-	-	disk01	mydg	failed was:c0t0d0s2

The display shows that the reinstalled root device, `c0t0d0s2`, is not associated with a VM disk and is marked with a status of `error`. The disks `disk02` and `disk03` were not involved in the reinstallation and are recognized by VxVM and associated with their devices (`c0t1d0s2` and `c0t2d0s2`). The former `disk01`, which was the VM disk associated with the replaced disk device, is no longer associated with the device (`c0t0d0s2`).

If other disks (with volumes or mirrors on them) had been removed or replaced during reinstallation, those disks would also have a disk device listed in `error` state and a VM disk listed as not associated with a device.

- 3 Once you know which disks have been removed or replaced, locate all the mirrors on failed disks using the following command:

```
# vxprint [-g diskgroup] -sF "%vname" -e'sd_disk = "
disk
"'
```

where `disk` is the access name of a disk with a `failed` status. Be sure to enclose the disk name in quotes in the command. Otherwise, the command returns an error message. The `vxprint` command returns a list of volumes that have mirrors on the failed disk. Repeat this command for every disk with a `failed` status.

The following is sample output from running this command:

```
# vxprint -g mydg -sF "%vname" -e'sd_disk = "disk01"'
v01
```

- 4 Check the status of each volume and print volume information using the following command:

```
# vxprint -th volume
```

where `volume` is the name of the volume to be examined. The `vxprint` command displays the status of the volume, its plexes, and the portions of disks that make up those plexes. For example, a volume named `v01` with only one plex resides on the reinstalled disk named `disk01`. The `vxprint -th v01` command produces the following output:

V	NAME	RVG/VSET/COKSTATE	STATE	LENGTH	READPOL	PREFPLEXUTYPE		
PL	NAME	VOLUME	KSTATE	STATE	LENGTH	LAYOUT	NCOL/WID	MODE
SD	NAME	PLEX	DISK	DISKOFFS	LENGTH	[COL/]OFF	DEVICE	MODE
v	v01	-	DISABLED	ACTIVE	24000	SELECT	-	fsgen
pl	v01-01	v01	DISABLED	NODEVICE	24000	CONCAT	-	RW
sd	disk01-06	v01-01	disk01	245759	24000	0	c1t5d1	ENA

The only plex of the volume is shown in the line beginning with `pl`. The `STATE` field for the plex named `v01-01` is `NODEVICE`. The plex has space on a disk that has been replaced, removed, or reinstalled. The plex is no longer valid and must be removed.

- 5 Because `v01-01` was the only plex of the volume, the volume contents are irrecoverable except by restoring the volume from a backup. The volume must also be removed. If a backup copy of the volume exists, you can restore the volume later. Keep a record of the volume name and its length, as you will need it for the backup procedure.

Remove irrecoverable volumes (such as `v01`) using the following command:

```
# vxedit -r rm v01
```

- 6 It is possible that only part of a plex is located on the failed disk. If the volume has a striped plex associated with it, the volume is divided between several disks. For example, the volume named `v02` has one striped plex striped across three disks, one of which is the reinstalled disk `disk01`. The `vxprint -th v02` command produces the following output:

V	NAME	RVG/VSET/COKSTATE		STATE	LENGTH	READPOL	PREFPLEX	UTYPE
PL	NAME	VOLUME	KSTATE	STATE	LENGTH	LAYOUT	NCOL/WID	MODE
SD	NAME	PLEX	DISK	DISKOFFS	LENGTH	[COL/]OFF	DEVICE	MODE
v	v02	-	DISABLED	ACTIVE	30720	SELECT	v02-01	fsgen
pl	v02-01	v02	DISABLED	NODEVICE	30720	STRIPE	3/128	RW
sd	disk02-02v02-01		disk01	424144	10240	0/0	c1t5d2	ENA
sd	disk01-05v02-01		disk01	620544	10240	1/0	c1t5d3	DIS
sd	disk03-01v02-01		disk03	620544	10240	2/0	c1t5d4	ENA

The display shows three disks, across which the plex `v02-01` is striped (the lines starting with `sd` represent the stripes). One of the stripe areas is located on a failed disk. This disk is no longer valid, so the plex named `v02-01` has a state of `NODEVICE`. Since this is the only plex of the volume, the volume is invalid and must be removed. If a copy of `v02` exists on the backup media, it can be restored later. Keep a record of the volume name and length of any volume you intend to restore from backup.

Remove invalid volumes (such as `v02`) using the following command:

```
# vxedit -r rm v02
```

- 7** A volume that has one mirror on a failed disk can also have other mirrors on disks that are still valid. In this case, the volume does not need to be restored from backup, since the data is still valid on the valid disks.

The output of the `vxprint -th` command for a volume with one plex on a failed disk (`disk01`) and another plex on a valid disk (`disk02`) is similar to the following:

V	NAME	RVG/VSET/COKSTATE	STATE	LENGTH	READPOL	PREFPLEX	UTYPE
PL	NAME	VOLUME	KSTATE	STATE	LENGTH	LAYOUT	NCOL/WID MODE
SD	NAME	PLEX	DISK	DISKOFFS	LENGTH	[COL/]OFF	DEVICE MODE
v	v03	-	DISABLED	ACTIVE	0720	SELECT	- fsgen
pl	v03-01	v03	DISABLED	ACTIVE	30720	CONCAT	- RW
sd	disk02-01	v03-01	disk01	620544	30720	0	clt5d5 ENA
pl	v03-02	v03	DISABLED	NODEVICE	30720	CONCAT	- RW
sd	disk01-04	v03-02	disk03	262144	30720	0	clt5d6 DIS

This volume has two plexes, `v03-01` and `v03-02`. The first plex (`v03-01`) does not use any space on the invalid disk, so it can still be used. The second plex (`v03-02`) uses space on invalid disk `disk01` and has a state of `NODEVICE`. Plex `v03-02` must be removed. However, the volume still has one valid plex containing valid data. If the volume needs to be mirrored, another plex can be added later. Note the name of the volume to create another plex later.

To remove an invalid plex, use the `vxplex` command to dissociate and then remove the plex from the volume. For example, to dissociate and remove the plex `v03-02`, use the following command:

```
# vxplex -o rm dis v03-02
```

- 8** Once all invalid volumes and plexes have been removed, the disk configuration can be cleaned up. Each disk that was removed, reinstalled, or replaced (as determined from the output of the `vxdisk list` command) must be removed from the configuration.

To remove a disk, use the `vxdg` command. For example, to remove the failed disk `disk01`, use the following command:

```
# vxdg rmdisk disk01
```

If the `vxdg` command returns an error message, invalid mirrors exist.

Repeat step 2 through step 7 until all invalid volumes and mirrors are removed.

- 9 Once all the invalid disks have been removed, the replacement or reinstalled disks can be added to Veritas Volume Manager control. If the root disk was originally under Veritas Volume Manager control or you now wish to put the root disk under Veritas Volume Manager control, add this disk first.

To add the root disk to Veritas Volume Manager control, use the `vxdiskadm` command:

```
# vxdiskadm
```

From the `vxdiskadm` main menu, select menu item 2 (Encapsulate a disk). Follow the instructions and encapsulate the root disk for the system.

- 10 When the encapsulation is complete, reboot the system to multi-user mode.
- 11 Once the root disk is encapsulated, any other disks that were replaced should be added using the `vxdiskadm` command. If the disks were reinstalled during the operating system reinstallation, they should be encapsulated; otherwise, they can be added.
- 12 Once all the disks have been added to the system, any volumes that were completely removed as part of the configuration cleanup can be recreated and their contents restored from backup. The volume recreation can be done by using the `vxassist` command or the graphical user interface.

For example, to recreate the volumes `v01` and `v02`, use the following command:

```
# vxassist make v01 24000
# vxassist make v02 30720 layout=stripe nstripe=3
```

Once the volumes are created, they can be restored from backup using normal backup/restore procedures.

- 13 Recreate any plexes for volumes that had plexes removed as part of the volume cleanup. To replace the plex removed from volume `v03`, use the following command:

```
# vxassist mirror v03
```

Once you have restored the volumes and plexes lost during reinstallation, recovery is complete and your system is configured as it was prior to the failure.

- 14 Start up hot-relocation, if required, by either rebooting the system or manually start the relocation watch daemon, `vxrelocd` (this also starts the `vxnotify` process).

Warning: Hot-relocation should only be started when you are sure that it will not interfere with other reconfiguration procedures.

To determine if hot-relocation has been started, use the following command to search for its entry in the process table:

```
# ps -ef | grep vxrelocd
```

See the *Veritas Volume Manager Administrator's Guide*.

See the `vxrelocd(1M)` manual page.

Logging commands and transactions

This chapter includes the following topics:

- [Command logs](#)
- [Transaction logs](#)
- [Association of command and transaction logs](#)
- [Associating CVM commands issued from slave to master node](#)

Command logs

The `vxcmdlog` command allows you to log the invocation of other Veritas Volume Manager (VxVM) commands to a file.

The following examples demonstrate the usage of `vxcmdlog`:

<code>vxcmdlog -l</code>	List current settings for command logging.
<code>vxcmdlog -m on</code>	Turn on command logging.
<code>vxcmdlog -s 512k</code>	Set the maximum command log file size to 512K.
<code>vxcmdlog -n 10</code>	Set the maximum number of historic command log files to 10.
<code>vxcmdlog -n no_limit</code>	Remove any limit on the number of historic command log files.
<code>vxcmdlog -m off</code>	Turn off command logging.

Command lines are logged to the file, `cmdlog`, in the directory `/etc/vx/log`. This path name is a symbolic link to a directory whose location depends on the operating system. If required, you can redefine the directory which is linked.

If you want to preserve the settings of the `vxcmdlog` utility, you must also copy the settings file, `.cmdlog`, to the new directory.

Warning: The `.cmdlog` file is a binary and should not be edited.

The size of the command log is checked after an entry has been written so the actual size may be slightly larger than that specified. When the log reaches a maximum size, the current command log file, `cmdlog`, is renamed as the next available historic log file, `cmdlog.number`, where *number* is an integer from 1 up to the maximum number of historic log files that is currently defined, and a new current log file is created.

A limited number of historic log files is preserved to avoid filling up the file system. If the maximum number of historic log files has been reached, the oldest historic log file is removed, and the current log file is renamed as that file.

Each log file contains a header that records the host name, host ID, and the date and time that the log was created.

The following are sample entries from a command log file:

```
# 0, 2329, Wed Feb 12 21:19:31 2003
    /usr/sbin/vxdctl mode
# 17051, 2635, Wed Feb 12 21:19:33 2003
    /usr/sbin/vxdisk -q -o alldgs list
# 0, 2722, Wed Feb 12 21:19:34 2003
    /etc/vx/diag.d/vxprivutil dumpconfig /dev/vx/rdmp/Disk_4s2
# 26924, 3001, Thu Feb 13 19:30:57 2003
    /usr/sbin/vxdisk list Disk_1
```

Each entry usually contains a client ID that identifies the command connection to the `vxconfigd` daemon, the process ID of the command that is running, a time stamp, and the command line including any arguments.

If the client ID is 0, as in the third entry shown here, this means that the command did not open a connection to `vxconfigd`.

The client ID is the same as that recorded for the corresponding transactions in the transactions log.

See [“Transaction logs”](#) on page 83.

See [“Association of command and transaction logs”](#) on page 85.

Most command scripts are not logged, but the command binaries that they call are logged. Exceptions are the `vxdisksetup`, `vxinstall`, and `vxdiskunsetup` scripts, which are logged.

If there is an error reading from the settings file, command logging switches to its built-in default settings. This may mean, for example, that logging remains enabled after being disabled using `vxcmdlog -m off` command. If this happens, use the `vxcmdlog` utility to recreate the settings file, or restore the file from a backup.

See the `vxcmdlog(1M)` manual page.

Transaction logs

The `vxtranslog` command allows you to log VxVM transactions to a file.

The following examples demonstrate the usage of `vxtranslog`:

<code>vxtranslog -l</code>	List current settings for transaction logging.
<code>vxtranslog -m on</code>	Turn on transaction logging.
<code>vxtranslog -s 512k</code>	Set the maximum transaction log file size to 512K.
<code>vxtranslog -n 10</code>	Set the maximum number of historic transaction log files to 10.
<code>vxtranslog -n no_limit</code>	Remove any limit on the number of historic transaction log files.
<code>vxtranslog -q on</code>	Turn on query logging.
<code>vxtranslog -q off</code>	Turn off query logging.
<code>vxtranslog -m off</code>	Turn off transaction logging.

Transactions are logged to the file, `translog`, in the directory `/etc/vx/log`. This path name is a symbolic link to a directory whose location depends on the operating system. If required, you can redefine the directory which is linked. If you want to preserve the settings of the `vxtranslog` utility, you must also copy the settings file, `.translog`, to the new directory.

Warning: The `.translog` file is a binary and should not be edited.

The size of the transaction log is checked after an entry has been written so the actual size may be slightly larger than that specified. When the log reaches a

maximum size, the current transaction log file, *translog*, is renamed as the next available historic log file, *translog.number*, where *number* is an integer from 1 up to the maximum number of historic log files that is currently defined, and a new current log file is created.

A limited number of historic log files is preserved to avoid filling up the file system. If the maximum number of historic log files has been reached, the oldest historic log file is removed, and the current log file is renamed as that file.

Each log file contains a header that records the host name, host ID, and the date and time that the log was created.

The following are sample entries from a transaction log file:

```
Thu Feb 13 19:30:57 2003
Clid = 26924, PID = 3001, Part = 0, Status = 0, Abort Reason = 0
  DA_GET SENA0_1
  DISK_GET_ATTRS SENA0_1
  DISK_DISK_OP SENA0_1 8
  DEVNO_GET SENA0_1
  DANAME_GET 0x1d801d8 0x1d801a0
  GET_ARRAYNAME SENA 50800200000e78b8
  CTLR_PTOLNAME /pci@1f,4000/pci@5/SUNW,qlc@4/fp@0,0
  GET_ARRAYNAME SENA 50800200000e78b8
  CTLR_PTOLNAME /pci@1f,4000/pci@5/SUNW,qlc@5/fp@0,0
  DISCONNECT <no request data>
```

The first line of each log entry is the time stamp of the transaction. The *Clid* field corresponds to the client ID for the connection that the command opened to *vxconfigd*. The *PID* field shows the process ID of the utility that is requesting the operation. The *Status* and *Abort Reason* fields contain error codes if the transaction does not complete normally. The remainder of the record shows the data that was used in processing the transaction.

The client ID is the same as that recorded for the corresponding command line in the command log.

See [“Command logs”](#) on page 81.

See [“Association of command and transaction logs”](#) on page 85.

If there is an error reading from the settings file, transaction logging switches to its built-in default settings. This may mean, for example, that logging remains enabled after being disabled using *vxtranslog -m off* command. If this happens, use the *vxtranslog* utility to recreate the settings file, or restore the file from a backup.

Association of command and transaction logs

The Client and process IDs that are recorded for every request and command assist you in correlating entries in the command and transaction logs. To find out which command issued a particular request in transaction log, use a command such as the following to search for the process ID and the client ID in the command log:

```
# egrep -n PID cmdlog | egrep Clid
```

In this example, the following request was recorded in the transaction log:

```
Wed Feb 12 21:19:36 2003
Clid = 8309, PID = 2778, Part = 0, Status = 0, Abort Reason = 0
  DG_IMPORT foodg
  DG_IMPORT foodg
  DISCONNECT <no request data>
```

To locate the utility that issued this request, the command would be:

```
# egrep -n 2778 cmdlog | egrep 8309
7310:# 8309, 2778, Wed Feb 12 21:19:36 2003
```

The output from the example shows a match at line 7310 in the command log. Examining lines 7310 and 7311 in the command log indicates that the `vx dg import` command was run on the `foodg` disk group:

```
# sed -e '7310,7311!d' cmdlog
# 8309, 2778, Wed Feb 12 21:19:36 2003 7311
/usr/sbin/vxdg -m import foodg
```

If there are multiple matches for the combination of the client and process ID, you can determine the correct match by examining the time stamp.

If a utility opens a conditional connection to `vxconfigd`, its client ID is shown as zero in the command log, and as a non-zero value in the transaction log. You can use the process ID and time stamp to relate the log entries in such cases.

Associating CVM commands issued from slave to master node

When you run commands on the CVM slave node that change the shared disk group configuration, CVM ships the commands to the CVM master node for execution.

For example, on the slave node, you run the following command, to create a volume in a shared disk group. CVM ships the command to the master node, and CVM executes the command on the master node.

```
# vxassist -g shareddg make shared-voll 200M
```

On the CVM slave node, enter the following command to identify the shipped command from the transaction log (translog):

```
# egrep CMDSHIP_REQUEST translog
```

In this example, the following entry was recorded in the transaction log on slave node:

```
Thu Jul 15 06:30:16 2010
Clid = 5302, PID = 589906, Part = 0, Status = 0, Abort Reason = 0
    DG_SET_CURRENT_ID shareddg
    DG_SET_CURRENT shareddg
    DG_GETCFG_ID 0xdde49f shareddg
    DG_GETCFG_NAME 0xdde49f shareddg
    DG_SET_CURRENT_ID shareddg
    DG_SET_CURRENT shareddg
    DG_SET_CURRENT_ID shareddg
    DG_SET_CURRENT shareddg
    DG_GETCFG_ALL 0x420
    DG_GETCFG_ALL 0x420
    VOL_TRANS ds4700-0_7 ds4700-0_3 ds4700-0_
    DG_GET_DEFAULT <no request data>
    CMDSHIP_REQUEST Command Shipped = /usr/sbin/vxassist -g

shareddg make shared-voll 200M
    Default dg = nodg
    DROPPED <no request data>
```

To locate the utility that issued this request on slave node, enter the following command:

```
# egrep -n PID cmdlog | egrep Clid
```

In this example, enter the following command:

```
# egrep -n 589906 cmdlog | egrep 5302
```

```
7310#: 5302, 589906, Thu Jul 15 06:30:14 2010 /usr/sbin/vxassist -g
```

The output from the example shows a match at line 7310 in the command log. Examining lines 7310 and 7311 in the command log indicates that the `vxassist make` command was run on the `sharedddg` disk group:

```
# sed -e '7310,7311!d' cmdlog

# 5302, 589906, Thu Jul 15 06:30:14 2010
/usr/sbin/vxassist -g sharedddg make shared-voll 200M
```

To determine the cluster monitor nodeid (CM nid) of the slave node, enter the following command:

```
# /etc/vx/bin/vxclustadm nidmap
```

If the command uses disk access (DA) names, the shipped command converts the DA names to unique disk IDs (UDID) or Disk media (DM) names. On the CVM master node, the `vxconfigd` log shows the entry for the received command. To determine the commands received from slave nodes on the master, enter the command:

```
# egrep CMDSHIP_REQUEST /var/adm/messages
```

Note: The file to which the `vxconfigd` messages are logged may differ, depending on where the messages are redirected.

In this example, the following received command would be recorded in the `vxconfigd` log on master node:

```
07/15 06:29:02: V-5-1-0 receive_cmdship_message:
CMDSHIP_REQUEST: Received command:
Text - /usr/sbin/vxassist -g sharedddg make shared-voll 200M len = 53
CLID = 5302 SlaveID = 0 Defaultdg = nodg
```

The CVM master node executes the received command and sends the response to the slave node.

To find the response that the master node sent to the slave node, enter a command such as the following on the master node:

```
# egrep CMDSHIP_RESPONSE translog | egrep Slave-Clid
```

In this example, enter the following command to find the response that the master node sent:

```
# egrep CMDSHIP_RESPONSE translog | egrep 5302
Thu Jul 15 06:29:03 2010
```

```
Clid = 27741, PID = 475212, Part = 0, Status = 0, Abort Reason = 0  
CMDSHIP_RESPONSE SlaveCLID = 5302 SlaveCMID = 0  
ExitCode = 12 Flags = 1 stdoutlen = 0 stderrlen = 98 Response =
```

```
VxVM vxassist ERROR V-5-1-10127 creating volume shared-vol1:
```

```
Record already exists in disk group  
DROPPED <no request data>
```

Backing up and restoring disk group configurations

This chapter includes the following topics:

- [About disk group configuration backup](#)
- [Backing up a disk group configuration](#)
- [Restoring a disk group configuration](#)

About disk group configuration backup

Disk group configuration backup and restoration allows you to backup and restore all configuration data for Veritas Volume Manager (VxVM) disk groups, and for VxVM objects such as volumes that are configured within the disk groups. Using this feature, you can recover from corruption of a disk group's configuration that is stored as metadata in the private region of a VM disk. After the disk group configuration has been restored, and the volume enabled, the user data in the public region is available again without the need to restore this from backup media.

Warning: The backup and restore utilities act only on VxVM configuration data. They do not back up or restore any user or application data that is contained within volumes or other VxVM objects. If you use `vxdiskunsetup` and `vxdisksetup` on a disk, and specify attributes that differ from those in the configuration backup, this may corrupt the public region and any data that it contains.

The `vxconfigbackupd` daemon monitors changes to the VxVM configuration and automatically records any configuration changes that occur. Two utilities,

`vxconfigbackup` and `vxconfigrestore`, are provided for backing up and restoring a VxVM configuration for a disk group.

When importing a disk group, any of the following errors indicate that the disk group configuration and/or disk private region headers have become corrupted:

```
VxVM vxconfigd ERROR V-5-1-569 Disk group group,Disk disk:Cannot
auto-import group: reason
```

The reason for the error is usually one of the following:

```
Configuration records are inconsistent
Disk group has no valid configuration copies
Duplicate record in configuration
Errors in some configuration copies
Format error in configuration copy
Invalid block number
Invalid magic number
```

If VxVM cannot update a disk group's configuration because of disk errors, it disables the disk group and displays the following error:

```
VxVM vxconfigd ERROR V-5-1-123 Disk group group: Disabled by errors
```

If such errors occur, you can restore the disk group configuration from a backup after you have corrected any underlying problem such as failed or disconnected hardware.

Configuration data from a backup allows you to reinstall the private region headers of VxVM disks in a disk group whose headers have become damaged, to recreate a corrupted disk group configuration, or to recreate a disk group and the VxVM objects within it. You can also use the configuration data to recreate a disk group on another system if the original system is not available.

Note: Restoration of a disk group configuration requires that the same physical disks are used as were configured in the disk group when the backup was taken.

See [“Backing up a disk group configuration”](#) on page 90.

See [“Restoring a disk group configuration”](#) on page 91.

Backing up a disk group configuration

VxVM uses the disk group configuration daemon to monitor the configuration of disk groups, and to back up the configuration whenever it is changed. By default,

the five most recent backups are preserved. If required, you can also back up a disk group configuration by running the `vxconfigbackup` command.

The following files record disk group configuration information:

<code>/etc/vx/cbr/bk/diskgroup.dgid/dgid.dginfo</code>	Disk group information.
<code>/etc/vx/cbr/bk/diskgroup.dgid/dgid.diskinfo</code>	Disk attributes.
<code>/etc/vx/cbr/bk/diskgroup.dgid/dgid.binconfig</code>	Binary configuration copy.
<code>/etc/vx/cbr/bk/diskgroup.dgid/dgid.cfgrec</code>	Configuration records in <code>vxprint -m</code> format.

Here *diskgroup* is the name of the disk group, and *dgid* is the disk group ID. If a disk group is to be recreated on another system, copy these files to that system.

Warning: Take care that you do not overwrite any files on the target system that are used by a disk group on that system.

To back up a disk group configuration

- ◆ Type the following command:

```
# /etc/vx/bin/vxconfigbackup diskgroup
```

To back up all disk groups, use this version of the command:

```
# /etc/vx/bin/vxconfigbackup
```

See the `vxconfigbackup(1M)` manual page.

Restoring a disk group configuration

You can use the `vxconfigrestore` utility to restore or recreate a disk group from its configuration backup. The restoration process consists of a precommit operation followed by a commit operation. At the precommit stage, you can examine the configuration of the disk group that would be restored from the backup. The actual disk group configuration is not permanently restored until you choose to commit the changes.

Warning: None of the disks or VxVM objects in the disk group may be open or in use by any application while the restoration is being performed.

You can choose whether or not any corrupted disk headers are to be reinstalled at the precommit stage. If any of the disks' private region headers are invalid, restoration may not be possible without reinstalling the headers for the affected disks.

See the `vxconfigrestore(1M)` manual page.

To perform the precommit operation

- ◆ Use the following command to perform a precommit analysis of the state of the disk group configuration, and to reinstall the disk headers where these have become corrupted:

```
# /etc/vx/bin/vxconfigrestore -p [-l directory] \  
  {diskgroup | dgid}
```

The disk group can be specified either by name or by ID.

The `-l` option allows you to specify a directory for the location of the backup configuration files other than the default location, `/etc/vx/cbr/bk`.

See [“Backing up a disk group configuration”](#) on page 90.

To specify that the disk headers are not to be reinstalled

- ◆ Type the following command:

```
# /etc/vx/bin/vxconfigrestore -n [-l directory] \  
  {diskgroup | dgid}
```

At the precommit stage, you can use the `vxprint` command to examine the configuration that the restored disk group will have. You can choose to proceed to commit the changes and restore the disk group configuration. Alternatively, you can cancel the restoration before any permanent changes have been made.

To abandon restoration at the precommit stage

- ◆ Type the following command:

```
# /etc/vx/bin/vxconfigrestore -d [-l directory] \  
  {diskgroup | dgid}
```

To perform the commit operation

- ◆ To commit the changes that are required to restore the disk group configuration, use the following command:

```
# /etc/vx/bin/vxconfigrestore -c [-l directory] \  
  {diskgroup | dgid}
```

If no disk headers are reinstalled, the configuration copies in the disks' private regions are updated from the latest binary copy of the configuration that was saved for the disk group.

If any of the disk headers are reinstalled, a saved copy of the disks' attributes is used to recreate their private and public regions. These disks are also assigned new disk IDs. The VxVM objects within the disk group are then recreated using the backup configuration records for the disk group. This process also has the effect of creating new configuration copies in the disk group.

Volumes are synchronized in the background. For large volume configurations, it may take some time to perform the synchronization. You can use the `vxtask -l list` command to monitor the progress of this operation.

Disks that are in use or whose layout has been changed are excluded from the restoration process.

If the back-up is taken of a shared disk group, the `vxconfigrestore` command restores it as a private disk group. After the disk group is restored, run the following commands to make the disk group shared.

To make the disk group shared

- 1 Deport the disk group:

```
# vxdg deport dg_name
```

- 2 Import the disk group as shared:

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

Resolving conflicting backups for a disk group

In some circumstances where disks have been replaced on a system, there may exist several conflicting backups for a disk group. In this case, you see a message similar to the following from the `vxconfigrestore` command:

```
VxVM vxconfigrestore ERROR V-5-1-6012 There are two backups that  
have the same diskgroup name with different diskgroup id :
```

```
1047336696.19.xxx.veritas.com
```

```
1049135264.31.xxx.veritas.com
```

The solution is to specify the disk group by its ID rather than by its name to perform the restoration. The backup file, */etc/vx/cbr/bk/diskgroup. dgid/dgid.dginfo*, contains a timestamp that records when the backup was taken.

The following is a sample extract from such a backup file that shows the timestamp and disk group ID information:

```
TIMESTAMP
Tue Apr 15 23:27:01 PDT 2003
.
.
.
DISK_GROUP_CONFIGURATION
Group:      mydg
dgid: 1047336696.19.xxx.veritas.com
.
.
.
```

Use the timestamp information to decide which backup contains the relevant information, and use the `vxconfigrestore` command to restore the configuration by specifying the disk group ID instead of the disk group name.

Restoring a previous array support library

This chapter includes the following topics:

- [Downgrading the array support](#)

Downgrading the array support

The array support is available in a single package, `VRTSaslapm`, that includes Array Support Libraries (ASLs) and Array Policy Modules (APMs). Each major release of Veritas Volume Manager includes the supported `VRTSaslapm` package, which is installed as part of the product installation. Between major releases, Symantec may provide additional array support through updates to the `VRTSaslapm` package.

In some circumstances, you may need to revert to a previous version of the ASL/APM package. You can only revert to a package that is supported for the installed release of Veritas Volume Manager. To perform the downgrade while the system is online, do not remove the installed package. Instead, you can install the previous version of the package over the new package. This method prevents multiple instances of the `VRTSaslapm` package from being installed.

Use the following method to downgrade the `VRTSaslapm` package.

To downgrade the ASL/APM package

- 1 Create a response file to the `pkgadd` command that specifies `instance=overwrite`. The following example shows a response file:

```
#
# Copyright 2004 Sun Microsystems, Inc. All rights reserved.
# Use is subject to license terms.
#
#ident  "@(#)default    1.7      04/12/21 SMI"
#
mail=
instance=overwrite
partial=ask
runlevel=ask
idepend=ask
rdepend=ask
space=ask
setuid=ask
conflict=ask
action=ask
networktimeout=60
networkretries=3
authentication=quit
keystore=/var/sadm/security
proxy=
basedir=default
```

- 2 To downgrade the package, use the following command:

```
pkgadd -a <response_file> -d ./VRTSaslapm.pkg
```

DMP troubleshooting

This chapter includes the following topics:

- [Displaying extended attributes after upgrading to DMP 5.1SP1](#)

Displaying extended attributes after upgrading to DMP 5.1SP1

You may see the following changes in functionality from the Storage Foundation 5.1 release:

- The device names that are listed in the `vxdisk list` output do not display the Array Volume IDs (AVIDs).
- The `vxdisk -e list` output does not display extended attributes.
- An Active/Passive (A/P) or ALUA array is claimed as Active/Active (A/A).

This behavior may be because the LUNs are controlled by the native multi-pathing driver, MPxIO.

To check whether LUNs are controlled by the native multi-pathing driver

- ◆ Check the output of the following command. The LUN is controlled by MPxIO if the controller of the affected LUN has the physical name (PNAME) as `/scsi_vhci`:

```
# vxddm adm getctlr all
```

You can migrate the LUNs from the control of the native multi-pathing driver to DMP control.

- To migrate to DMP with Veritas Volume Manager, refer to the section on disabling MPxIO in the *Veritas Volume Manager Administrator's Guide*.

- To migrate to DMP with OS native volume support, refer to the section on migrating to DMP from MPxIO in the *Veritas Dynamic Multi-Pathing Administrator's Guide*.

Error messages

This chapter includes the following topics:

- [About error messages](#)
- [How error messages are logged](#)
- [Types of messages](#)

About error messages

Informational, failure, and other error messages may be displayed on the console by the Veritas Volume Manager (VxVM) configuration daemon (`vxconfigd`), the VxVM kernel driver, `vxio`, and the various VxVM commands. These messages may indicate errors that are infrequently encountered and difficult to troubleshoot.

Note: Some error messages described here may not apply to your system.

You may find it useful to consult the VxVM command and transaction logs to understand the context in which an error occurred.

See [“Command logs”](#) on page 81.

How error messages are logged

VxVM provides the option of logging debug messages to a file. This logging is useful in that any messages output just before a system crash will be available in the log file (presuming that the crash does not result in file system corruption).

If enabled, the default debug log file is `/var/vxvm/vxconfigd.log`.

To enable logging of debug output to the default debug log file, edit the startup script for `vxconfigd`.

`vxconfigd` also supports the use of `syslog` to log all of its regular console messages. When this is enabled, all console output is directed through the `syslog` interface.

`syslog` and log file logging can be used together to provide reliable logging to a private log file, along with distributed logging through `syslogd`.

Note: `syslog` logging is enabled by default. Debug message logging is disabled by default.

If `syslog` output is enabled, messages with a priority higher than `Debug` are written to `/var/log/syslog`.

See [“Configuring logging in the startup script”](#) on page 100.

Alternatively, you can use the following command to change the debug level:

```
# vxctl debug level [pathname]
```

There are 10 possible levels of debug logging with the values 0 through 9. Level 1 provides the least detail, and 9 the most. Level 0 turns off logging. If a path name is specified, this file is used to record the debug output instead of the default debug log file. If the `vxctl debug` command is used, the new debug logging level and debug log file remain in effect until the VxVM configuration daemon, `vxconfigd`, is next restarted.

See the `vxctl(1M)` manual page.

See the `vxconfigd(1M)` manual page.

Configuring logging in the startup script

To enable log file or `syslog` logging on a permanent basis, you can edit the `/lib/svc/method/vxvm-sysboot` (in Solaris 10) or `/etc/init.d/vxvm-sysboot` (in previous releases of the Solaris OS) script that starts the VxVM configuration daemon, `vxconfigd`.

To configure logging in the startup script

- 1 Comment-out or uncomment any of the following lines to enable or disable the corresponding feature in `vxconfigd`:

```
opts="$opts -x syslog"
# use syslog for console messages
#opts="$opts -x log"
# messages to vxconfigd.log
#opts="$opts -x logfile=/foo/bar" # specify an alternate log file
#opts="$opts -x timestamp"
# timestamp console messages

# To turn on debugging console output, uncomment the following line.
# The debug level can be set higher for more output. The highest
# debug level is 9.

#debug=1
# enable debugging console output
```

The `opts="$opts -x syslog"` string is usually uncommented so that `vxconfigd` uses `syslog` logging by default. Inserting a `#` character at the beginning of the line turns off `syslog` logging for `vxconfigd`.

If you do not specify a debug level, only Error, Fatal Error, Warning, and Notice messages are logged. Debug messages are not logged.

By default, `vxconfigd` is started at boot time with the `-x syslog` option. This redirects `vxconfigd` console messages to `syslog`. If you want to retain this behavior when restarting `vxconfigd` from the command line, include the `-x syslog` argument, as restarting `vxconfigd` does not preserve the option settings with which it was previously running. Similarly, any Veritas Volume Manager operations that require `vxconfigd` to be restarted may not retain the behavior that was previously specified by option settings.

- 2 After making changes to the way `vxconfigd` is invoked in the startup file, run the following command on a Solaris 10 system to notify that the service configuration has been changed:

```
# svcadm refresh vxvm/vxvm-sysboot
```

Types of messages

VxVM is fault-tolerant and resolves most problems without system administrator intervention. If the configuration daemon, `vxconfigd`, recognizes the actions that are necessary, it queues up the transactions that are required. VxVM provides atomic changes of system configurations; either a transaction completes fully, or the system is left in the same state as though the transaction was never attempted. If `vxconfigd` is unable to recognize and fix system problems, the system administrator needs to handle the task of problem solving using the diagnostic messages that are returned from the software. The following sections describe error message numbers and the types of error message that may be seen, and provide a list of the more common errors, a detailed description of the likely cause of the problem together with suggestions for any actions that can be taken.

Messages have the following generic format:

```
product component severity message_number message_text
```

For Veritas Volume Manager, the product is set to `VxVM`. The component can be the name of a kernel module or driver such as `vxldmp`, a configuration daemon such as `vxconfigd`, or a command such as `vxassist`.

Note: For full information about saving system crash information, see the *Solaris System Administration Guide*.

Messages are divided into the following types of severity in decreasing order of impact on the system:

PANIC

A panic is a severe event as it halts a system during its normal operation. A panic message from the kernel module or from a device driver indicates a hardware problem or software inconsistency so severe that the system cannot continue. The operating system may also provide a dump of the CPU register contents and a stack trace to aid in identifying the cause of the panic. The following is an example of such a message:

```
VxVM vxio PANIC V-5-0-239 Object association  
depth overflow
```

FATAL ERROR	A fatal error message from a configuration daemon, such as <code>vxconfigd</code>, indicates a severe problem with the operation of VxVM that prevents it from running. The following is an example of such a message: <pre><i>VxVM vxconfigd FATAL ERROR V-5-0-591 Disk group bootdg: Inconsistency -- Not loaded into kernel</i></pre>
ERROR	An error message from a command indicates that the requested operation cannot be performed correctly. The following is an example of such a message: <pre><i>VxVM vxassist ERROR V-5-1-5150 Insufficient number of active snapshot mirrors in snapshot_volume .</i></pre>
WARNING	A warning message from the kernel indicates that a non-critical operation has failed, possibly because some resource is not available or the operation is not possible. The following is an example of such a message: <pre><i>VxVM vxio WARNING V-5-0-55 Cannot find device number for boot_path</i></pre>
NOTICE	A notice message indicates that an error has occurred that should be monitored. Shutting down the system is unnecessary, although you may need to take action to remedy the fault at a later date. The following is an example of such a message: <pre><i>VxVM vxio NOTICE V-5-0-252 read error on object subdisk of mirror plex in volume volume (start offset, length length) corrected.</i></pre>
INFO	An informational message does not indicate an error, and requires no action.

The unique message number consists of an alpha-numeric string that begins with the letter “V”. For example, in the message number, `V-5-1-3141`, “V” indicates that this is a Veritas product error message, the first numeric field (5) encodes the product (in this case, VxVM), the second field (1) represents information about the product component, and the third field (3141) is the message index. The text of the error message follows the message number.

Messages

This section contains a list of messages that you may encounter during the operation of Veritas Volume Manager. However, the list is not exhaustive and the

second field may contain the name of different command, driver or module from that shown here.

Descriptions are included to elaborate on the situation or problem that generated a particular message. Wherever possible, a recovery procedure is provided to help you to locate and correct the problem.

If you encounter a product error message, record the unique message number preceding the text of the message. Search on the message number at the following URL to find the information about that message:

<http://sort.symantec.com/>

When contacting Veritas Technical Support, either by telephone or by visiting the Veritas Technical Support website, be sure to provide the relevant message number. Veritas Technical Support will use this message number to quickly determine if there are TechNotes or other information available for you.

V-5-0-2

```
VxVM vxio WARNING V-5-0-2 object_type object_name  
blockoffset:Uncorrectable read error ...
```

```
VxVM vxio WARNING V-5-0-2 object_type object_name  
blockoffset:Uncorrectable write error ...
```

Description:

A read or write operation from or to the specified Veritas Volume Manager object failed. An error is returned to the application.

Recommended action:

These errors may represent lost data. Data may need to be restored and failed media may need to be repaired or replaced. Depending on the type of object failing and on the type of recovery suggested for the object type, an appropriate recovery operation may be necessary.

V-5-0-4

```
VxVM vxio WARNING V-5-0-4 Plex plex detached from volume volume
```

Description:

An uncorrectable error was detected by the mirroring code and a mirror copy was detached.

This message may also appear during a plex detach operation in a cluster.

Recommended action:

To restore redundancy, it may be necessary to add another mirror. The disk on which the failure occurred should be reformatted or replaced.

If this message appears during a plex detach operation in a cluster, no action is required.

V-5-0-34

```
VxVM vxdump NOTICE V-5-0-34 added disk array disk_array_serial_number
```

Description:

A new disk array has been added to the host.

Recommended action:

No recovery procedure is required.

V-5-0-35

```
VxVM vxdump NOTICE V-5-0-35 Attempt to disable controller  
controller_name failed. Rootdisk has just one enabled path.
```

Description:

An attempt is being made to disable the one remaining active path to the root disk controller.

Recommended action:

The path cannot be disabled.

V-5-0-55

```
VxVM vxio WARNING V-5-0-55 Cannot find device number for boot_path  
vxvm vxdump WARNING V-5-0-55 Cannot find device number for boot_path
```

Description:

The boot path retrieved from the system PROMs cannot be converted to a valid device number.

Recommended action:

Check your PROM settings for the correct boot string. If a target driver, such as an ATF, coexists with VxVM, and the target driver claims the boot disk, the message may be ignored if the device path corresponds to the boot disk.

V-5-0-64

VxVM vxio WARNING V-5-0-64 cannot log commit record for Diskgroup
bootdg: error 28

Description:

This message usually means that multi-pathing is misconfigured.

Recommended action:

Possible causes and solutions are similar to the message V-5-1-5929.

See [“V-5-1-5929”](#) on page 155.

V-5-0-106

VxVM vxio WARNING V-5-0-106 detaching RAID-5 volume

Description:

Either a double-failure condition in the RAID-5 volume has been detected in the kernel or some other fatal error is preventing further use of the array.

Recommended action:

If two or more disks have been lost due to a controller or power failure, use the `vxrecover` utility to recover them once they have been re-attached to the system. Check for other console error messages that may provide additional information about the failure.

V-5-0-108

VxVM vxio WARNING V-5-0-108 Device *major, minor*: Received spurious
close

Description:

A close was received for an object that was not open. This can only happen if the operating system is not correctly tracking opens and closes.

Recommended action:

No action is necessary; the system will continue.

V-5-0-110

VxVM vxdmp NOTICE V-5-0-110 disabled controller *controller_name*
connected to disk array *disk_array_serial_number*

Description:

All paths through the controller connected to the disk array are disabled. This usually happens if a controller is disabled for maintenance.

Recommended action:

No recovery procedure is required.

V-5-0-111

```
VxVM vxdmp NOTICE V-5-0-111 disabled dmpnode dmpnode_device_number
```

Description:

A Dynamic Multi-Pathing (DMP) node has been marked disabled in the DMP database. It will no longer be accessible for further IO requests. This occurs when all paths controlled by a DMP node are in the disabled state, and therefore inaccessible.

Recommended action:

Check hardware or enable the appropriate controllers to enable at least one path under this DMP node.

V-5-0-112

```
VxVM vxdmp NOTICE V-5-0-112 disabled path path_device_number belonging  
to dmpnode dmpnode_device_number
```

Description:

A path has been marked disabled in the DMP database. This path is controlled by the DMP node indicated by the specified device number. This may be due to a hardware failure.

Recommended action:

Check the underlying hardware if you want to recover the desired path.

V-5-0-144

```
VxVM vxio WARNING V-5-0-144 Double failure condition detected on  
RAID-5 volume
```

Description:

I/O errors have been received in more than one column of a RAID-5 volume.

The error can be caused by one of the following problems:

- a controller failure making more than a single drive unavailable
- the loss of a second drive while running in degraded mode

- two separate disk drives failing simultaneously (unlikely)

Recommended action:

Correct the hardware failures if possible. Then recover the volume using the `vxrecover` command.

V-5-0-145

VxVM vxio WARNING V-5-0-145 DRL volume *volume* is detached

Description:

A Dirty Region Logging volume became detached because a DRL log entry could not be written. If this is due to a media failure, other errors may have been logged to the console.

Recommended action:

The volume containing the DRL log continues in operation. If the system fails before the DRL has been repaired, a full recovery of the volume's contents may be necessary and will be performed automatically when the system is restarted. To recover from this error, use the `vxassist addlog` command to add a new DRL log to the volume.

V-5-0-146

VxVM vxdmp NOTICE V-5-0-146 enabled controller *controller_name* connected to disk array *disk_array_serial_number*

Description:

All paths through the controller connected to the disk array are enabled. This usually happens if a controller is enabled after maintenance.

Recommended action:

No recovery procedure is required.

V-5-0-147

VxVM vxdmp NOTICE V-5-0-147 enabled dmpnode *dmpnode_device_number*

A DMP node has been marked enabled in the DMP database. This happens when at least one path controlled by the DMP node has been enabled.

Recommended action:

No recovery procedure is required.

V-5-0-148

VxVM vxdmp NOTICE V-5-0-148 enabled path *path_device_number* belonging to dmpnode *dmpnode_device_number*

Description:

A path has been marked enabled in the DMP database. This path is controlled by the DMP node indicated by the specified device number. This happens if a previously disabled path has been repaired, the user has reconfigured the DMP database using the `vxdctl(1M)` command, or the DMP database has been reconfigured automatically.

Recommended action:

No recovery procedure is required.

V-5-0-164

VxVM vxio WARNING V-5-0-164 Failed to join cluster *name*, aborting

Description:

A node failed to join a cluster. This may be caused by the node being unable to see all the shared disks. Other error messages may provide more information about the disks that cannot be found.

Recommended action:

Use the `vxdisk -s list` command on the master node to see what disks should be visible to the slave node. Then check that the operating system and VxVM on the failed node can also see these disks. If the operating system cannot see the disks, check the cabling and hardware configuration of the node. If only VxVM cannot see the disks, use the `vxdctl enable` command to make it scan again for the disks. When the disks are visible to VxVM on the node, retry the join.

V-5-0-166

VxVM vxio WARNING V-5-0-166 Failed to log the detach of the DRL volume *volume*

Description:

An attempt failed to write a kernel log entry indicating the loss of a DRL volume. The attempted write to the log failed either because the kernel log is full, or because of a write error to the drive. The volume becomes detached.

Recommended action:

Messages about log failures are usually fatal, unless the problem is transient. However, the kernel log is sufficiently redundant that such errors are unlikely to occur.

If the problem is not transient (that is, the drive cannot be fixed and brought back online without data loss), recreate the disk group from scratch and restore all of its volumes from backups. Even if the problem is transient, reboot the system after correcting the problem.

If error messages are seen from the disk driver, it is likely that the last copy of the log failed due to a disk error. Replace the failed drive in the disk group. The log re-initializes on the new drive. Finally force the failed volume into an active state and recover the data.

V-5-0-168

```
VxVM vxio WARNING V-5-0-168 Failure in RAID-5 logging operation
```

Description:

Indicates that a RAID-5 log has failed.

Recommended action:

To restore RAID-5 logging to a RAID-5 volume, create a new log plex and attach it to the volume.

V-5-0-181

```
VxVM vxio WARNING V-5-0-181 Illegal vminor encountered
```

Description:

An attempt was made to open a volume device (other than the root volume device) before `vxconfigd` loaded the volume configuration.

Recommended action:

No recovery procedure is required. Under normal startup conditions, this message should not occur. If necessary, start VxVM and re-attempt the operation.

V-5-0-194

```
VxVM vxio WARNING V-5-0-194 Kernel log full: volume detached
```

Description:

A plex detach failed because the kernel log was full. As a result, the mirrored volume will become detached.

Recommended action:

This condition is unlikely to occur. The only corrective action is to reboot the system.

V-5-0-196

VxVM vxio WARNING V-5-0-196 Kernel log update failed: *volume* detached

Description:

Detaching a plex failed because the kernel log could not be flushed to disk. As a result, the mirrored volume became detached. This may be caused by all the disks containing a kernel log going bad.

Recommended action:

Repair or replace the failed disks so that kernel logging can once again function.

V-5-0-207

VxVM vxio WARNING V-5-0-207 log object *object_name* detached from RAID-5 volume

Description:

This message indicates that a RAID-5 log has failed.

Recommended action:

To restore RAID-5 logging to a RAID-5 volume, create a new log plex and attach it to the volume.

V-5-0-216

VxVM vxio WARNING V-5-0-216 *mod_install* returned *errno*

Description:

A call made to the operating system *mod_install* function to load the *vxio* driver failed.

Recommended action:

Check for additional console messages that may explain why the load failed. Also check the console messages log file for any additional messages that were logged but not displayed on the console.

V-5-0-237

VxVM vxio WARNING V-5-0-237 object *subdisk* detached from RAID-5 volume at column *column* offset *offset*

Description:

A subdisk was detached from a RAID-5 volume because of the failure of a disk or an uncorrectable error occurring on that disk.

Recommended action:

Check for other console error messages indicating the cause of the failure. Replace a failed disk as soon as possible.

V-5-0-243

```
VxVM vxio WARNING V-5-0-243 Overlapping mirror plex detached from  
volume volume
```

Description:

An error has occurred on the last complete plex in a mirrored volume. Any sparse mirrors that map the failing region are detached so that they cannot be accessed to satisfy that failed region inconsistently.

Recommended action:

The message indicates that some data in the failing region may no longer be stored redundantly.

V-5-0-244

```
VxVM vxdmp NOTICE V-5-0-244 Path failure on major/minor
```

Description:

A path under the control of the DMP driver failed. The device major and minor numbers of the failed device is supplied in the message.

Recommended action:

No recovery procedure is required.

V-5-0-249

```
VxVM vxio WARNING V-5-0-249 RAID-5 volume entering degraded mode  
operation
```

Description:

An uncorrectable error has forced a subdisk to detach. At this point, not all data disks exist to provide the data upon request. Instead, parity regions are used to regenerate the data for each stripe in the array. Consequently, access takes longer and involves reading from all drives in the stripe.

Recommended action:

Check for other console error messages that indicate the cause of the failure.
Replace any failed disks as soon as possible.

V-5-0-251

```
VxVM vxio WARNING V-5-0-251 read error on object object of mirror  
plexin volume volume (start offset length length)
```

Description:

An error was detected while reading from a mirror. This error may lead to further action shown by later error messages.

Recommended action:

If the volume is mirrored, no further action is necessary since the alternate mirror's contents will be written to the failing mirror; this is often sufficient to correct media failures. If this error occurs often, but never leads to a plex detach, there may be a marginally defective region on the disk at the position indicated. It may eventually be necessary to remove data from this disk and then to reformat the drive.

See the `vxevac(1M)` manual page.

If the volume is not mirrored, this message indicates that some data could not be read. The file system or other application reading the data may report an additional error, but in either event, data has been lost. The volume can be partially salvaged and moved to another location if desired.

This message may also appear during a plex detach operation in a cluster. In this case, no action is required.

V-5-0-252

```
VxVM vxio NOTICE V-5-0-252 read error on object subdisk of mirror  
plex in volume volume (start offset length length) corrected
```

Description:

A read error occurred, which caused a read of an alternate mirror and a writeback to the failing region. This writeback was successful and the data was corrected on disk.

Recommended action:

No recovery procedure is required. The problem was corrected automatically. Note the location of the failure for future reference. If the same region of the subdisk fails again, this may indicate a more insidious failure and the disk should be reformatted at the next reasonable opportunity.

V-5-0-258

```
VxVM vxdmp NOTICE V-5-0-258 removed disk array  
disk_array_serial_number
```

Description:

A disk array has been disconnected from the host, or some hardware failure has resulted in the disk array becoming inaccessible to the host.

Recommended action:

Replace disk array hardware if this has failed.

V-5-0-281

```
VxVM vxio WARNING V-5-0-281 Root volumes are not supported on your  
PROM version.
```

Description:

If your system's PROMs are not a recent OpenBoot PROM type, `root` volumes are unusable.

Recommended action:

If you have set up a `root` volume, undo the configuration by running `vxunroot` or removing the `rootdev` line from `/etc/system` as soon as possible. Contact your hardware vendor for an upgrade to your PROM level.

V-5-0-386

```
VxVM vxio WARNING V-5-0-386 subdisk subdisk failed in plex plex in  
volume volume
```

Description:

The kernel has detected a subdisk failure, which may mean that the underlying disk is failing.

Recommended action:

Check for obvious problems with the disk (such as a disconnected cable). If hot-relocation is enabled and the disk is failing, recovery from subdisk failure is handled automatically.

V-5-1-90

```
VxVM vxconfigd ERROR V-5-1-90 mode: Unrecognized operating mode
```

Description:

An invalid string was specified as an argument to the `-m` option. Valid strings are: `enable`, `disable`, and `boot`.

Recommended action:

Supply a correct option argument.

V-5-1-91

```
VxVM vxconfigd WARNING V-5-1-91 Cannot create device device_path:  
reason
```

Description:

`vxconfigd` cannot create a device node either under `/dev/vx/dsk` or under `/dev/vx/rdsk`. This should happen only if the `root` file system has run out of inodes.

Recommended action:

Remove some unwanted files from the `root` file system. Then, regenerate the device node using this command:

```
# vxctl enable
```

V-5-1-92

```
VxVM vxconfigd WARNING V-5-1-92 Cannot exec /usr/bin/rm to remove  
directory: reason
```

Description:

The given directory could not be removed because the `/usr/bin/rm` utility could not be executed by `vxconfigd`. This is not a serious error. The only side effect of a directory not being removed is that the directory and its contents continue to use space in the `root` file system. However, this does imply that the `/usr` file system is not mounted, or on some systems, that the `rm` utility is missing or is not in its usual location. This may be a serious problem for the general running of your system.

Recommended action:

If the `/usr` file system is not mounted, you need to determine how to get it mounted. If the `rm` utility is missing, or is not in the `/usr/bin` directory, restore it.

V-5-1-111

VxVM vxconfigd WARNING V-5-1-111 Cannot fork to remove directory
directory: reason

Description:

The given directory could not be removed because `vxconfigd` could not fork in order to run the `rm` utility. This is not a serious error. The only side effect of a directory not being removed is that the directory and its contents will continue to use space in the `root` file system. The most likely cause for this error is that your system does not have enough memory or paging space to allow `vxconfigd` to fork.

Recommended action:

If your system is this low on memory or paging space, your overall system performance is probably substantially degraded. Consider adding more memory or paging space.

V-5-1-116

VxVM vxconfigd WARNING V-5-1-116 Cannot open log file *log_filename:*
reason

Description:

The `vxconfigd` console output log file could not be opened for the given reason.

Recommended action:

Create any needed directories, or use a different log file path name.

See [“How error messages are logged”](#) on page 99.

V-5-1-117

VxVM vxconfigd ERROR V-5-1-117 Cannot start volume *volume*, no valid
plexes

Description:

This error indicates that the volume cannot be started because it does not contain any valid plexes. This can happen, for example, if disk failures have caused all plexes to be unusable. It can also happen as a result of actions that caused all plexes to become unusable (for example, forcing the dissociation of subdisks or detaching, dissociation, or offlining of plexes).

Recommended action:

It is possible that this error results from a drive that failed to spin up. If so, rebooting may fix the problem. If that does not fix the problem, then the only recourse is to repair the disks involved with the plexes and restore the file system from a backup.

Restoring the `root` or `/usr` file system requires that you have a valid backup.

See [“Repair of root or /usr file systems on mirrored volumes”](#) on page 61.

V-5-1-121

VxVM vxconfigd NOTICE V-5-1-121 Detached disk *disk*

Description:

The named disk appears to have become unusable and was detached from its disk group. Additional messages may appear to indicate other records detached as a result of the disk detach.

Recommended action:

If hot-relocation is enabled, Veritas Volume Manager objects affected by the disk failure are taken care of automatically. Mail is sent to `root` indicating what actions were taken by VxVM and what further actions the administrator should take.

V-5-1-122

VxVM vxconfigd WARNING V-5-1-122 Detaching plex *plex* from volume *volume*

Description:

This error only happens for volumes that are started automatically by `vxconfigd` at system startup (that is, the `root` and `/usr` file system volumes). The plex is being detached as a result of I/O failure, disk failure during startup or prior to the last system shutdown or crash, or disk removal prior to the last system shutdown or crash.

Recommended action:

To ensure that the `root` or `/usr` file system retains the same number of active mirrors, remove the given plex and add a new mirror using the `vxassist mirror` operation. Also consider replacing any bad disks before running this command.

V-5-1-123

VxVM vxconfigd ERROR V-5-1-123 Disk group *group*: Disabled by errors

Description:

This message indicates that some error condition has made it impossible for VxVM to continue to manage changes to a disk group. The major reason for this is that too many disks have failed, making it impossible for `vxconfigd` to continue to update configuration copies. There should be a preceding error message that indicates the specific error that was encountered.

If the disk group that was disabled is the boot disk group, the following additional error is displayed:

```
VxVM vxconfigd ERROR V-5-1-104 All transactions are disabled
```

This additional message indicates that `vxconfigd` has entered the disabled state, which makes it impossible to change the configuration of any disk group, not just the boot disk group.

Recommended action:

If the underlying error resulted from a transient failure, such as a disk cabling error, then you may be able to repair the situation by rebooting. Otherwise, the disk group configuration may have to be recreated, and the contents of any volumes restored from a backup.

See [“Restoring a disk group configuration”](#) on page 91.

Failure of the boot disk group may require reinstallation of the system if your system uses a `root` or `/usr` file system that is defined on a volume.

V-5-1-124

```
VxVM vxconfigd ERROR V-5-1-124 Disk group group: update failed: reason
```

Description:

I/O failures have prevented `vxconfigd` from updating any active copies of the disk group configuration. This usually implies a large number of disk failures. This error will usually be followed by the error:

```
VxVM vxconfigd ERROR V-5-1-123 Disk group group: Disabled by errors
```

Recommended action:

If the underlying error resulted from a transient failure, such as a disk cabling error, then you may be able to repair the situation by rebooting. Otherwise, the disk group may have to be recreated and restored from a backup.

V-5-1-134

```
VxVM vxconfigd ERROR V-5-1-134 Memory allocation failure
```

Description:

This implies that there is insufficient memory to start VxVM.

Recommended action:

This error should not normally occur, unless your system has very small amounts of memory. Adding swap space will probably not help because this error is most likely to occur early in the boot sequence, before swap areas have been added.

V-5-1-135

```
VxVM vxconfigd FATAL ERROR V-5-1-135 Memory allocation failure during startup
```

Description:

This implies that there is insufficient memory to start up VxVM.

Recommended action:

This error should not normally occur, unless your system has very small amounts of memory. Adding swap space probably will not help, because this error is most likely to occur early in the boot sequence, before swap areas have been added.

V-5-1-148

```
VxVM vxconfigd ERROR V-5-1-148 System startup failed
```

Description:

Either the `root` or the `/usr` file system volume could not be started, rendering the system unusable. The error that resulted in this condition should appear prior to this error message.

Recommended action:

Look up other error messages appearing on the console and take the actions suggested in the descriptions of those messages.

V-5-1-169

```
VxVM vxconfigd ERROR V-5-1-169 cannot open /dev/vx/config: reason
```

```
VxVM vxconfigd ERROR V-5-1-169 Cannot open /etc/vfstab: reason
```

Description:

This message has the following variations:

- Case 1: `cannot open /dev/vx/config: reason`

The `/dev/vx/config` device could not be opened. `vxconfigd` uses this device to communicate with the Veritas Volume Manager kernel drivers. The most

likely reason is “Device is already open.” This indicates that some process (most likely `vxconfigd`) already has `/dev/vx/config` open. Less likely reasons are “No such file or directory” or “No such device or address.”

The following are likely causes:

- The Veritas Volume Manager package installation did not complete correctly.
- The device node was removed by the administrator or by an errant shell script.
- Case 2: `vxconfigd` could not open the `/etc/vfstab` file, for the reason given. The `/etc/vfstab` file is used to determine which volume (if any) to use for the `/usr` file system.

Recommended action:

For Case 1, if the reason is “Device is already open,” stop or kill the old `vxconfigd` by running the command:

```
# vxctl -k stop
```

For other failure reasons, consider re-adding the base Veritas Volume Manager package. This will reconfigure the device node and re-install the Veritas Volume Manager kernel device drivers. If you cannot re-install the package, contact Veritas Technical Support for more information.

For Case 2, this error implies that your `root` file system is currently unusable. You may be able to repair the `root` file system by mounting it after booting from a network or CD-ROM `root` file system.

See [“VxVM and boot disk failure”](#) on page 43.

V-5-1-249

VxVM vxconfigd NOTICE V-5-1-249 Volume volume entering degraded mode

Description:

Detaching a subdisk in the named RAID-5 volume has caused the volume to enter “degraded” mode. While in degraded mode, performance of the RAID-5 volume is substantially reduced. More importantly, failure of another subdisk may leave the RAID-5 volume unusable. Also, if the RAID-5 volume does not have an active log, then failure of the system may leave the volume unusable.

Recommended action:

If hot-relocation is enabled, Veritas Volume Manager objects affected by the disk failure are taken care of automatically. Mail is sent to `root` indicating what actions were taken by VxVM, and what further actions you should take.

V-5-1-480

VxVM vxconfigd ERROR V-5-1-480 Cannot reset VxVM kernel: *reason*

Description:

The `-r reset` option was specified to `vxconfigd`, but the VxVM kernel drivers could not be reset. The most common reason is “A virtual disk device is open.” This implies that a VxVM tracing or volume device is open.

Recommended action:

If you want to reset the kernel devices, track down and kill all processes that have a volume or Veritas Volume Manager tracing device open. Also, if any volumes are mounted as file systems, unmount those file systems.

Any reason other than “A virtual disk device is open” does not normally occur unless there is a bug in the operating system or in VxVM.

V-5-1-484

VxVM vxconfigd ERROR V-5-1-484 Cannot start *volume* volume, no valid complete plexes

Description:

These errors indicate that the volume cannot be started because the volume contains no valid complete plexes. This can happen, for example, if disk failures have caused all plexes to be unusable. It can also happen as a result of actions that caused all plexes to become unusable (for example, forcing the dissociation of subdisks or detaching, dissociation, or offlining of plexes).

Recommended action:

It is possible that this error results from a drive that failed to spin up. If so, rebooting may fix the problem. If that does not fix the problem, then the only recourse is to repair the disks involved with the plexes and restore the file system from a backup.

V-5-1-485

VxVM vxconfigd ERROR V-5-1-485 Cannot start *volume* volume, volume state is invalid

Description:

The volume for the `root` or `/usr` file system is in an unexpected state (not `ACTIVE`, `CLEAN`, `SYNC` or `NEEDSYNC`). This should not happen unless the system administrator circumvents the mechanisms used by VxVM to create these volumes.

Recommended action:

The only recourse is to bring up VxVM on a CD-ROM or NFS-mounted `root` file system and to fix the state of the volume.

See [“Repair of root or /usr file systems on mirrored volumes”](#) on page 61.

V-5-1-525

VxVM vxconfigd NOTICE V-5-1-525 Detached log for volume *volume*

Description:

The DRL or RAID-5 log for the named volume was detached as a result of a disk failure, or as a result of the administrator removing a disk with `vxdbg -k rmdisk`. A failing disk is indicated by a “Detached disk” message.

Recommended action:

If the log is mirrored, hot-relocation tries to relocate the failed log automatically. Use either `vxplex dis` or `vxsd dis` to remove the failing logs. Then, use `vxassist addlog` to add a new log to the volume.

See the `vxassist(1M)` manual page.

V-5-1-526

VxVM vxconfigd NOTICE V-5-1-526 Detached plex *plex* in volume *volume*

Description:

The specified plex was disabled as a result of a disk failure, or as a result of the administrator removing a disk with `vxdbg -k rmdisk`. A failing disk is indicated by a “Detached disk” message.

Recommended action:

If hot-relocation is enabled, Veritas Volume Manager objects affected by the disk failure are taken care of automatically. Mail is sent to `root` indicating what actions were taken by VxVM and what further actions the administrator should take.

V-5-1-527

VxVM vxconfigd NOTICE V-5-1-527 Detached subdisk *subdisk* in volume *volume*

Description:

The specified subdisk was disabled as a result of a disk failure, or as a result of the administrator removing a disk with `vxdbg -k rmdisk`. A failing disk is indicated by a “Detached disk” message.

Recommended action:

If hot-relocation is enabled, Veritas Volume Manager objects affected by the disk failure are taken care of automatically. Mail is sent to `root` indicating what actions were taken by VxVM and what further actions the administrator should take.

V-5-1-528

```
VxVM vxconfigd NOTICE V-5-1-528 Detached volume volume
```

Description:

The specified volume was detached as a result of a disk failure, or as a result of the administrator removing a disk with `vxdbg -k rmdisk`. A failing disk is indicated by a “Detached disk” message. Unless the disk error is transient and can be fixed with a reboot, the contents of the volume should be considered lost.

Recommended action:

Contact Veritas Technical Support.

V-5-1-543

```
VxVM vxconfigd ERROR V-5-1-543 Differing version of vxconfigd  
installed
```

Description:

A `vxconfigd` daemon was started after stopping an earlier `vxconfigd` with a non-matching version number. This can happen, for example, if you upgrade VxVM and then run `vxconfigd` without first rebooting.

Recommended action:

Reboot the system.

V-5-1-544

```
VxVM vxconfigd WARNING V-5-1-544 Disk disk in group group flagged as  
shared; Disk skipped
```

Description:

The given disk is listed as shared, but the running version of VxVM does not support shared disk groups.

Recommended action:

This message can usually be ignored. If you want to use the disk on this system, use `vxdiskadd` to add the disk. Do not do this if the disk really is shared with other systems.

V-5-1-545

```
VxVM vxconfigd WARNING V-5-1-545 Disk disk in group group locked by  
host hostid Disk skipped
```

Description:

The given disk is listed as locked by the host with the Veritas Volume Manager host ID (usually the same as the system host name).

Recommended action:

This message can usually be ignored. If you want to use the disk on this system, use `vxdiskadd` to add the disk. Do not do this if the disk really is shared with other systems.

V-5-1-546

```
VxVM vxconfigd WARNING V-5-1-546 Disk disk in group group: Disk device  
not found
```

Description:

No physical disk can be found that matches the named disk in the given disk group. This is equivalent to failure of that disk. (Physical disks are located by matching the disk IDs in the disk group configuration records against the disk IDs stored in the Veritas Volume Manager header on the physical disks.) This error message is displayed for any disk IDs in the configuration that are not located in the disk header of any physical disk. This may result from a transient failure such as a poorly-attached cable, or from a disk that fails to spin up fast enough. Alternately, this may happen as a result of a disk being physically removed from the system, or from a disk that has become unusable due to a head crash or electronics failure.

Any RAID-5 plexes, DRL log plexes, RAID-5 subdisks or mirrored plexes containing subdisks on this disk are unusable. Such disk failures (particularly on multiple disks) may cause one or more volumes to become unusable.

Recommended action:

If hot-relocation is enabled, Veritas Volume Manager objects affected by the disk failure are taken care of automatically. Mail is sent to `root` indicating what actions were taken by VxVM, and what further actions you should take.

V-5-1-554

VxVM vxconfigd WARNING V-5-1-554 Disk *disk* names group *group*, but group ID differs

Description:

As part of a disk group import, a disk was discovered that had a mismatched disk group name and disk group ID. This disk is not imported. This can only happen if two disk groups have the same name but have different disk group ID values. In such a case, one group is imported along with all its disks and the other group is not. This message appears for disks in the un-selected group.

Recommended action:

If the disks should be imported into the group, this must be done by adding the disk to the group at a later stage, during which all configuration information for the disk is lost.

V-5-1-557

VxVM vxconfigd ERROR V-5-1-557 Disk *disk*, group *group*, device *device*: not updated with new host ID Error: *reason*

Description:

This can result from using `vxctl hostid hostid` to change the Veritas Volume Manager host ID for the system. The error indicates that one of the disks in a disk group could not be updated with the new host ID. This usually indicates that the disk has become inaccessible or has failed in some other way.

Recommended action:

Try running the following command to determine whether the disk is still operational:

```
# vxdisk check device
```

If the disk is no longer operational, `vxdisk` should print a message such as:

```
device: Error: Disk write failure
```

This will result in the disk being taken out of active use in its disk group, if it has not already been taken out of use. If the disk is still operational, which should not be the case, `vxdisk` prints:

```
device: Okay
```

If the disk is listed as “Okay,” try running `vxctl hostid hostid` again. If it still results in an error, contact Veritas Technical Support.

V-5-1-568

VxVM vxconfigd WARNING V-5-1-568 Disk group *group* is disabled, disks not updated with new host ID

Description:

As a result of failures, the named disk group has become disabled. Earlier error messages should indicate the cause. This message indicates that disks in that disk group were not updated with a new Veritas Volume Manager host ID. This warning message should result only from a `vxctl hostid` operation.

Recommended action:

Typically, unless a disk group was disabled due to transient errors, there is no way to repair a disabled disk group. The disk group may have to be reconstructed from scratch. If the disk group was disabled due to a transient error such as a cabling problem, then a future reboot may not automatically import the named disk group, due to the change in the system's Veritas Volume Manager host ID. In such a case, import the disk group directly using `vxvg import` with the `-C` option.

V-5-1-569

VxVM vxconfigd ERROR V-5-1-569 Disk group *group*, Disk *disk*: Cannot auto-import group: *reason*

Description:

On system startup, `vxconfigd` failed to import the disk group associated with the named disk. A message related to the specific failure is given in *reason*. Additional error messages may be displayed that give more information on the specific error. In particular, this is often followed by:

VxVM vxconfigd ERROR V-5-1-579 Disk group *group*: Errors in some configuration copies: Disk *device*, copy number: Block *bno*: error ...

The most common reason for auto-import failures is excessive numbers of disk failures, making it impossible for VxVM to find correct copies of the disk group configuration database and kernel update log. Disk groups usually have enough copies of this configuration information to make such import failures unlikely.

A more serious failure is indicated by errors such as:

```
Configuration records are inconsistent
Disk group has no valid configuration copies
Duplicate record in configuration
Format error in configuration copy
```

Invalid block number
Invalid magic number

These errors indicate that all configuration copies have become corrupt (due to disk failures, writing on the disk by an application or the administrator, or bugs in VxVM).

Some correctable errors may be indicated by other error messages that appear in conjunction with the auto-import failure message. Look up those other errors for more information on their cause.

Failure of an auto-import implies that the volumes in that disk group will not be available for use. If there are file systems on those volumes, then the system may yield further errors resulting from inability to access the volume when mounting the file system.

Recommended action:

If the error is clearly caused by excessive disk failures, then you may have to recreate the disk group configuration, and restore the contents of any volumes from a backup.

See [“Restoring a disk group configuration”](#) on page 91.

There may be other error messages that appear which provide further information. See those other error messages for more information on how to proceed. If those errors do not make it clear how to proceed, contact Veritas Technical Support.

V-5-1-571

VxVM vxconfigd ERROR V-5-1-571 Disk group *group*, Disk *disk*: Skip disk group with duplicate name

Description:

Two disk groups with the same name are tagged for auto-importing by the same host. Disk groups are identified both by a simple name and by a long unique identifier (disk group ID) assigned when the disk group is created. Thus, this error indicates that two disks indicate the same disk group name but a different disk group ID.

VxVM does not allow you to create a disk group or import a disk group from another machine, if that would cause a collision with a disk group that is already imported. Therefore, this error is unlikely to occur under normal use.

The error can occur in the following cases:

- A disk group cannot be auto-imported due to some temporary failure. If you create a new disk group with the same name as the failed disk group and reboot, the new disk group is imported first. The auto-import of the older disk group

fails because more recently modified disk groups have precedence over older disk groups.

- A disk group is deported from one host using the `-h` option to cause the disk group to be auto-imported on reboot from another host. If the second host was already auto-importing a disk group with the same name, then reboot of that host will yield this error.

Recommended action:

If you want to import both disk groups, then rename the second disk group on import.

See the `vxdbg(1M)` manual page.

V-5-1-577

```
VxVM vxconfigd WARNING V-5-1-577 Disk group group: Disk group log
may be too small Log size should be at least number blocks
```

Description:

The log areas for the disk group have become too small for the size of configuration currently in the group. This message only occurs during disk group import; it can only occur if the disk was inaccessible while new database objects were added to the configuration, and the disk was then made accessible and the system restarted. This should not normally happen without first displaying a message about the database area size.

Recommended action:

Reinitialize the disks in the group with larger log areas. Note that this requires that you restore data on the disks from backups.

See the `vxdisk(1M)` manual page.

To reinitialize all of the disks, detach them from the group with which they are associated, reinitialize and re-add them. Then deport and re-import the disk group to effect the changes to the log areas for the group.

V-5-1-579

```
VxVM vxconfigd ERROR V-5-1-579 Disk group group: Errors in some
configuration copies: Disk disk, copy number: [Block number]:
reason...
```

Description:

During a failed disk group import, some of the configuration copies in the named disk group were found to have format or other types of errors which make those

copies unusable. This message lists all configuration copies that have uncorrected errors, including any appropriate logical block number. If no other reasons are displayed, then this may be the cause of the disk group import failure.

Recommended action:

If some of the copies failed due to transient errors (such as cable failures), then a reboot or re-import may succeed in importing the disk group. Otherwise, the disk group configuration may have to be restored.

See [“Restoring a disk group configuration”](#) on page 91.

V-5-1-583

```
VxVM vxconfigd ERROR V-5-1-583 Disk group group: Reimport of disk  
group failed: reason
```

Description:

After `vxconfigd` was stopped and restarted (or disabled and then enabled), VxVM failed to recreate the import of the indicated disk group. The reason for failure is specified. Additional error messages may be displayed that give further information describing the problem.

Recommended action:

A major cause for this kind of failure is disk failures that were not addressed before `vxconfigd` was stopped or disabled. If the problem is a transient disk failure, then rebooting may take care of the condition. The error may be accompanied by messages such as “Disk group has no valid configuration copies.” This indicates that the disk group configuration copies have become corrupt (due to disk failures, writing on the disk by an application or the administrator, or bugs in VxVM).

Use the `vxconfigrestore` utility to restore or recreate a disk group from its configuration backup.

See [“Restoring a disk group configuration”](#) on page 91.

V-5-1-587

```
VxVM vxdg ERROR V-5-1-587 disk group groupname: import failed: reason
```

Description:

The import of a disk group failed for the specified reason.

Recommended action:

The action to be taken depends on the reason given in the error message:

```
Disk is in use by another host
```

```
No valid disk found containing disk group
```

The first message indicates that disks have been moved from a system that has crashed or that failed to detect the group before the disk was moved. The locks stored on the disks must be cleared.

The second message indicates that the disk group does not contain any valid disks (not that it does not contain any disks). The disks may be considered invalid due to a mismatch between the host ID in their configuration copies and that stored in the `/etc/vx/volboot` file.

To clear locks on a specific set of devices, use the following command:

```
# vxdisk clearimport devicename...
```

To clear the locks during import, use the following command:

```
# vxdg -C import diskgroup
```

Warning: Be careful when using the `vxdisk clearimport` or `vxdg -C import` command on systems that have dual-ported disks. Clearing the locks allows those disks to be accessed at the same time from multiple hosts and can result in corrupted data.

An `import` operation fails if some disks for the disk group cannot be found among the disk drives attached to the system.

```
Disk for disk group not found
```

```
Disk group has no valid configuration copies
```

The first message indicates a recoverable error.

The second message indicates a fatal error that requires hardware repair or the creation of a new disk group, and recovery of the disk group configuration and data:

If some of the disks in the disk group have failed, you can force the disk group to be imported with this command:

```
# vxdg -f import diskgroup
```

Warning: Be careful when using the `-f` option. It can cause the same disk group to be imported twice from different sets of disks. This can cause the disk group configuration to become inconsistent.

As using the `-f` option to force the import of an incomplete disk group counts as a successful import, an incomplete disk group may be imported subsequently without this option being specified. This may not be what you expect.

These operations can also be performed using the `vxdiskadm` utility. To deport a disk group using `vxdiskadm`, select the menu item `Remove access to (deport)` a disk group. To import a disk group, select the menu item `Enable access to (import)` a disk group. The `vxdiskadm import` operation checks for host import locks and prompts to see if you want to clear any that are found. It also starts volumes in the disk group.

V-5-1-663

```
VxVM vxconfigd WARNING V-5-1-663 Group group: Duplicate virtual device
number(s): Volume volume remapped from major,minor to major,minor
...
```

Description:

The configuration of the named disk group includes conflicting device numbers. A disk group configuration lists the recommended device number to use for each volume in the disk group. If two volumes in two disk groups happen to list the same device number, then one of the volumes must use an alternate device number. This is called device number remapping. Remapping is a temporary change to a volume. If the other disk group is deported and the system is rebooted, then the volume that was remapped may no longer be remapped. Also, volumes that are remapped once are not guaranteed to be remapped to the same device number in further reboots.

Recommended action:

Use the `vx dg remminor` command to renumber all volumes in the offending disk group permanently.

See the `vx dg(1M)` manual page.

V-5-1-737

```
VxVM vxconfigd ERROR V-5-1-737 Mount point path: volume not in bootdg
disk group
```

Description:

The volume device listed in the `/etc/vfstab` file for the given mount-point directory (normally `/usr`) is listed as in a disk group other than the boot disk group. This error should not occur if the standard Veritas Volume Manager procedures are used for encapsulating the disk containing the `/usr` file system.

Recommended action:

Boot VxVM from a network or CD-ROM mounted root file system. Then, start up VxVM using `fixmountroot` on a valid mirror disk of the root file system. After starting VxVM, mount the `root` file system volume and edit the `/etc/vfstab` file. Change the file to use a direct partition for the file system. There should be a comment in the `/etc/vfstab` file that indicates which partition to use.

V-5-1-768

VxVM vxconfigd NOTICE V-5-1-768 Offlining config copy *number* on *diskdisk*: Reason: *reason*

Description:

An I/O error caused the indicated configuration copy to be disabled. This is a notice only, and does not normally imply serious problems, unless this is the last active configuration copy in the disk group.

Recommended action:

Consider replacing the indicated disk, since this error implies that the disk has deteriorated to the point where write errors cannot be repaired automatically. The error can also result from transient problems with cabling or power.

V-5-1-809

VxVM vxplex ERROR V-5-1-809 Plex *plex* in volume *volume* is locked by another utility.

Description:

The `vxplex` command fails because a previous operation to attach a plex did not complete. The `vxprint` command should show that one or both of the temporary and persistent utility fields (`TUTIL0` and `PUTIL0`) of the volume and one of its plexes are set.

Recommended action:

If the `vxtask list` command does not show a task running for the volume, use the `vxmend` command to clear the `TUTIL0` and `PUTIL0` fields for the volume and all its components for which these fields are set:

```
# vxmend -g diskgroup clear all volume plex ...
```

V-5-1-923

VxVM vxplex ERROR V-5-1-923 Record *volume* is in disk group *diskgroup1*
plex is in group *diskgroup2*.

Description:

An attempt was made to snap back a plex from a different disk group.

Recommended action:

Move the snapshot volume into the same disk group as the original volume.

V-5-1-946

VxVM vxconfigd FATAL ERROR V-5-1-946 bootdg cannot be imported during
boot

Description:

This message usually means that multi-pathing is misconfigured.

Recommended action:

Possible causes and solutions are similar to the message V-5-1-5929.

See “[V-5-1-5929](#)” on page 155.

V-5-1-1049

VxVM vxconfigd ERROR V-5-1-1049 System boot disk does not have a
valid rootvol plex Please boot from one of the following disks:

DISK	MEDIA	DEVICE	BOOT	COMMAND
<i>diskname</i>	<i>device</i>		<i>boot</i>	<i>vx-diskname...</i>

Description:

The system is configured to use a volume for the `root` file system, but was not booted on a disk containing a valid mirror of the `root` volume. Disks containing valid `root` mirrors are listed as part of the error message. A disk is usable as a boot disk if there is a `root` mirror on that disk which is not stale or offline.

Recommended action:

Try to boot from one of the named disks using the associated boot command that is listed in the message.

V-5-1-1063

VxVM vxconfigd ERROR V-5-1-1063 There is no volume configured for the root device

Description:

The system is configured to boot from a `root` file system defined on a volume, but there is no `root` volume listed in the configuration of the boot disk group.

The following are possible causes of this error:

- Case 1: The `/etc/system` file was erroneously updated to indicate that the `root` device is `/pseudo/vxio@0:0`. This can happen only as a result of direct manipulation by the administrator.
- Case 2: The system somehow has a duplicate boot disk group, one of which contains a `root` file system volume and one of which does not, and `vxconfigd` somehow chose the wrong one. Since `vxconfigd` chooses the more recently accessed version of the boot disk group, this error can happen if the system clock was updated incorrectly at some point (reversing the apparent access order of the two disk groups). This can also happen if some disk group was deported and assigned the same name as the boot disk group with locks given to this host.

Recommended action:

The following are the recommended actions for each case:

- Case 1: Boot the system on a CD-ROM or networking-mounted root file system, directly mount the disk partition of the root file system, and remove the following lines from `/etc/system`:

```
rootdev:/pseudo/vxio@0:0
set vxio:vol_rootdev_is_volume=1
```

- Case 2: Either boot the system with all drives in the offending version of the boot disk group turned off, or import and rename the offending boot disk group from another host.

See the `vxvg(1M)` manual page.

If you turn off the drives, run the following command after booting:

```
# vxvg flush bootdg
```

This updates time stamps on the imported version of the specified boot disk group, `bootdg`, which should make the correct version appear to be the more recently accessed. If this does not correct the problem, contact Veritas Technical Support.

V-5-1-1171

VxVM vxconfigd ERROR V-5-1-1171 Version number of kernel does not match vxconfigd

Description:

The release of `vxconfigd` does not match the release of the Veritas Volume Manager kernel drivers. This should happen only as a result of upgrading VxVM, and then running `vxconfigd` without a reboot.

Recommended action:

Reboot the system. If that does not cure the problem, re-add the VxVM packages .

V-5-1-1186

VxVM vxconfigd ERROR V-5-1-1186 Volume *volume* for mount point `/usr` not found in *bootdg* disk group

Description:

The system is configured to boot with `/usr` mounted on a volume, but the volume associated with `/usr` is not listed in the configuration of the boot disk group.

The following are possible causes of this error:

- Case 1: The `/etc/vfstab` file was erroneously updated to indicate the device for the `/usr` file system is a volume, but the volume named is not in the boot disk group. This should happen only as a result of direct manipulation by the administrator.
- Case 2: The system somehow has a duplicate boot disk group, one of which contains the `/usr` file system volume and one of which does not (or uses a different volume name), and `vxconfigd` somehow chose the wrong boot disk group. Since `vxconfigd` chooses the more recently accessed version of the boot disk group, this error can happen if the system clock was updated incorrectly at some point (causing the apparent access order of the two disk groups to be reversed). This can also happen if some disk group was deported and assigned the same name as the boot disk group with locks given to this host.

Recommended action:

The following are recommended actions:

- Case 1: Boot the system on a CD-ROM or networking-mounted `root` file system. If the `root` file system is defined on a volume, then start and mount the `root` volume. If the `root` file system is not defined on a volume, mount the `root` file

system directly. Edit the `/etc/vfstab` file to correct the entry for the `/usr` file system.

- **Case 2:** Either boot with all drives in the offending version of the boot disk group turned off, or import and rename the offending boot disk group from another host.

See the `vxvg(1M)` manual page.

If you turn off drives, run the following command after booting:

```
# vxvg flush bootdg
```

This updates time stamps on the imported version of the boot disk group, *bootdg*, which should make the correct version appear to be the more recently accessed. If this does not correct the problem, contact Veritas Technical Support.

V-5-1-1589

```
VxVM vxconfigd ERROR V-5-1-1589 enable failed: aborting
```

```
VxVM vxconfigd ERROR V-5-1-1589 enable failed: Error check group
configuration copies. Database file not found
```

```
VxVM vxconfigd ERROR V-5-1-1589 enable failed: transactions are
disabled
```

Description:

Regular startup of `vxconfigd` failed. This error can also result from the command `vxctl enable`. This message has several variations, described below:

```
VxVM vxconfigd ERROR V-5-1-1589 enable failed: aborting
```

The failure was fatal and `vxconfigd` was forced to exit. The most likely cause is that the operating system is unable to create interprocess communication channels to other utilities.

```
VxVM vxconfigd ERROR V-5-1-1589 enable failed: Error check group
configuration copies. Database file not found
```

The directory `/var/vxvm/tempdb` is inaccessible. This may be because of `root` file system corruption, a full `root` file system, or if `/var` is a separate file system, because it has become corrupted or has not been mounted.

```
VxVM vxconfigd ERROR V-5-1-1589 enable failed: transactions are
disabled
```

`vxconfigd` continues to run, but no configuration updates are possible until the error condition is repaired.

Additionally, this may be followed with this message:

```
VxVM vxconfigd ERROR V-5-1-579 Disk group group: Errors in some
configuration copies: Disk device, copy number: Block bno: error ...
```

Other error messages may be displayed that further indicate the underlying problem.

Recommended action:

```
VxVM vxconfigd ERROR V-5-1-1589 enable failed: aborting
```

The failure was fatal and `vxconfigd` was forced to exit. The most likely cause is that the operating system is unable to create interprocess communication channels to other utilities.

```
VxVM vxconfigd ERROR V-5-1-1589 enable failed: Error check group
configuration copies. Database file not found
```

If the `root` file system is full, increase its size or remove files to make space for the `tempdb` file.

If `/var` is a separate file system, make sure that it has an entry in `/etc/vfstab`. Otherwise, look for I/O error messages during the boot process that indicate either a hardware problem or misconfiguration of any logical volume management software being used for the `/var` file system. Also verify that the encapsulation (if configured) of your boot disk is complete and correct.

```
VxVM vxconfigd ERROR V-5-1-1589 enable failed: transactions are
disabled
```

Evaluate the error messages to determine the root cause of the problem. Make changes suggested by the errors and then try rerunning the command.

If the “Errors in some configuration copies” error occurs again, that may indicate the real problem lies with the configuration copies in the disk group.

See [“Restoring a disk group configuration”](#) on page 91.

V-5-1-2020

```
VxVM vxconfigd ERROR V-5-1-2020 Cannot kill existing daemon,
pid=process_ID
```

Description:

The `-k` (kill existing `vxconfigd` process) option was specified, but a running configuration daemon process could not be killed. A configuration daemon process, for purposes of this discussion, is any process that opens the `/dev/vx/config` device (only one process can open that device at a time). If there is a configuration

daemon process already running, then the `-k` option causes a `SIGKILL` signal to be sent to that process. If, within a certain period of time, there is still a running configuration daemon process, the error message is displayed.

Recommended action:

This error can result from a kernel error that has made the configuration daemon process unkillable, from some other kind of kernel error, or from some other user starting another configuration daemon process after the `SIGKILL` signal. This last condition can be tested for by running `vxconfigd -k` again. If the error message reappears, contact Veritas Technical Support.

V-5-1-2197

VxVM vxconfigd ERROR V-5-1-2197 node *N*: missing vxconfigd

Description:

The `vxconfigd` daemon is not running on the indicated cluster node.

Recommended action:

Restart the `vxconfigd` daemon.

V-5-1-2198

VxVM vxconfigd ERROR V-5-1-2198 node *N*: vxconfigd not ready

Description:

The `vxconfigd` daemon is not responding properly in a cluster.

Recommended action:

Stop and restart the `vxconfigd` daemon on the node indicated.

V-5-1-2274

VxVM vxconfigd ERROR V-5-1-2274 volume:vxconfigd cannot boot-start
RAID-5 volumes

Description:

A volume that `vxconfigd` should start immediately upon booting the system (that is, the volume for the `/usr` file system) has a RAID-5 layout. The `/usr` file system should never be defined on a RAID-5 volume.

Recommended action:

It is likely that the only recovery for this is to boot VxVM from a network-mounted root file system (or from a CD-ROM), and reconfigure the `/usr` file system to be defined on a regular non-RAID-5 volume.

V-5-1-2290

VxVM vxddmpadm ERROR V-5-1-2290 Attempt to enable a controller that is not available

Description:

This message is returned by the `vxddmpadm` utility when an attempt is made to enable a controller that is not working or is not physically present.

Recommended action:

Check hardware and see if the controller is present and whether I/O can be performed through it.

V-5-1-2353

VxVM vxconfigd ERROR V-5-1-2353 Disk group *group*: Cannot recover temp database: *reason* Consider use of "`vxconfigd -x cleartempdir`" [see `vxconfigd(1M)`].

Description:

This error can happen if you kill and restart `vxconfigd`, or if you disable and enable `vxconfigd` with `vxctl disable` and `vxctl enable`. The error indicates a failure related to reading the file `/var/vxvm/tempdb/group`. This is a temporary file used to store information that is used when recovering the state of an earlier `vxconfigd`. The file is recreated on a reboot, so this error should never survive a reboot.

Recommended action:

If you can reboot the system, do so. If you do not want to reboot, then use the following procedure.

To correct the error without rebooting

- 1 Ensure that no `vxvol`, `vxplex`, or `vxsd` processes are running.

Use `ps -e` to search for such processes, and use `kill` to kill any that you find. You may have to run `kill` twice to make these processes go away. Killing utilities in this way may make it difficult to make administrative changes to some volumes until the system is rebooted.

- 2 Recreate the temporary database files for all imported disk groups using the following command:

```
# vxconfigd -x cleartempdir 2> /dev/console
```

The `vxvol`, `vxplex`, and `vxsd` commands make use of these `tempdb` files to communicate locking information. If the file is cleared, then locking information can be lost. Without this locking information, two utilities can end up making incompatible changes to the configuration of a volume.

V-5-1-2524

VxVM vxconfigd ERROR V-5-1:2524 VOL_IO_DAEMON_SET failed: daemon count must be above *N* while cluster

Description:

The number of Veritas Volume Manager kernel daemons (`vxiod`) is less than the minimum number needed to join a cluster.

Recommended action:

Increase the number of daemons using `vxiod`.

V-5-1-2630

VxVM vxconfigd WARNING V-5-1-2630 library and vxconfigd disagree on existence of client *number*

Description:

This warning may safely be ignored.

Recommended action:

No recovery procedure is required.

V-5-1-2824

VxVM vxconfigd ERROR V-5-1-2824 Configuration daemon error 242

Description:

A node failed to join a cluster, or a cluster join is taking too long. If the join fails, the node retries the join automatically.

Recommended action:

No action is necessary if the join is slow or a retry eventually succeeds.

V-5-1-2829

VxVM vxdg ERROR V-5-1-2829 *diskgroup*: Disk group version doesn't support feature; see the vxdg upgrade command

Description:

The version of the specified disk group does not support disk group move, split or join operations.

Recommended action:

Use the `vx dg upgrade diskgroup` command to update the disk group version.

V-5-1-2830

VxVM vxconfigd ERROR V-5-1-2830 Disk reserved by other host

Description:

An attempt was made to online a disk whose controller has been reserved by another host in the cluster.

Recommended action:

No action is necessary. The cluster manager frees the disk and VxVM puts it online when the node joins the cluster.

V-5-1-2841

VxVM vxconfigd ERROR V-5-1-2841 enable failed: Error in disk group configuration copies Unexpected kernel error in configuration update; transactions are disabled.

Description:

Usually means that multi-pathing is misconfigured.

Recommended action:

Possible causes and solutions are similar to the message V-5-1-5929.

See “[V-5-1-5929](#)” on page 155.

V-5-1-2860

VxVM vxdg ERROR V-5-1-2860 Transaction already in progress

Description:

One of the disk groups specified in a disk group move, split or join operation is currently involved in another unrelated disk group move, split or join operation (possibly as the result of recovery from a system failure).

Recommended action:

Use the `vxprint` command to display the status of the disk groups involved. If `vxprint` shows that the TUTILO field for a disk group is set to MOVE, and you are certain that no disk group move, split or join should be in progress, use the `vxdg` command to clear the field.

See [“Recovering from an incomplete disk group move”](#) on page 29.

Otherwise, retry the operation.

V-5-1-2862

VxVM vxdg ERROR V-5-1-2862 *object*: Operation is not supported

Description:

DCO and snap objects dissociated by Persistent FastResync, and VVR objects cannot be moved between disk groups.

Recommended action:

No action is necessary. The operation is not supported.

V-5-1-2866

VxVM vxdg ERROR V-5-1-2866 *object*: Record already exists in disk group

Description:

A disk group join operation failed because the name of an object in one disk group is the same as the name of an object in the other disk group. Such name clashes are most likely to occur for snap objects and snapshot plexes.

Recommended action:

Use the following command to change the object name in either one of the disk groups:

```
# vxedit -g diskgroup rename old_name new_name
```

See the `vxedit(1M)` manual page.

V-5-1-2870

`VxVM vxdg ERROR V-5-1-2870 volume: Volume or plex device is open or mounted`

Description:

An attempt was made to perform a disk group move, split or join on a disk group containing an open volume.

Recommended action:

It is most likely that a file system configured on the volume is still mounted. Stop applications that access volumes configured in the disk group, and unmount any file systems configured in the volumes.

V-5-1-2879

`VxVM vxdg ERROR V-5-1-2879 subdisk: Record is associated`

Description:

The named subdisk is not a top-level object.

Recommended action:

Objects specified for a disk group move, split or join must be either disks or top-level volumes.

V-5-1-2907

`VxVM vxdg ERROR V-5-1-2907 diskgroup: Disk group does not exist`

Description:

The disk group does not exist or is not imported

Recommended action:

Use the correct name, or import the disk group and try again.

V-5-1-2908

`VxVM vxdg ERROR V-5-1-2908 diskdevice: Request crosses disk group boundary`

Description:

The specified disk device is not configured in the source disk group for a disk group move or split operation.

Recommended action:

Correct the name of the disk object specified in the disk group move or split operation.

V-5-1-2911

VxVM vxdg ERROR V-5-1-2911 *diskname*: Disk is not usable

Description:

The specified disk has become unusable.

Recommended action:

Do not include the disk in any disk group move, split or join operation until it has been replaced or repaired.

V-5-1-2922

VxVM vxconfigd ERROR V-5-1-2922 Disk group exists and is imported

Description:

A slave tried to join a cluster, but a shared disk group already exists in the cluster with the same name as one of its private disk groups.

Recommended action:

Use the `vxdg -n newname import diskgroup` operation to rename either the shared disk group on the master, or the private disk group on the slave.

V-5-1-2928

VxVM vxdg ERROR V-5-1-2928 *diskgroup*: Configuration too large for configuration copies

Description:

The disk group's configuration database is too small to hold the expanded configuration after a disk group move or join operation.

Recommended action:

No action is required.

V-5-1-2933

VxVM vxdg ERROR V-5-1-2933 *diskgroup*: Cannot remove last disk group configuration copy

Description:

The requested disk group move, split or join operation would leave the disk group without any configuration copies.

Recommended action:

No action is required. The operation is not supported.

V-5-1-2935

VxVM vxassist ERROR V-5-1-2935 No more space in disk group configuration.

Description:

There is no more space in the disk group's configuration database for VxVM object records.

Recommended action:

Copy the contents of several volumes to another disk group and then delete the volumes from this disk group, or use the disk group split/join feature to move the volumes to another disk group. To avoid the problem in the future, do not create more than a few hundred volumes in a disk group, or specify a larger size for the private region when adding disks to a new disk group.

V-5-1-3009

VxVM vxdbg ERROR V-5-1-3009 *object:* Name conflicts with imported diskgroup

Description:

The target disk group of a split operation already exists as an imported disk group.

Recommended action:

Choose a different name for the target disk group.

V-5-1-3020

VxVM vxconfigd ERROR V-5-1-3020 Error in cluster processing

Description:

This may be due to an operation inconsistent with the current state of a cluster (such as an attempt to import or deport a shared disk group to or from the slave). It may also be caused by an unexpected sequence of commands from vxclust.

Recommended action:

Perform the operation from the master node.

V-5-1-3022

VxVM vxconfigd ERROR V-5-1-3022 Cannot find disk on slave node

Description:

A slave node in a cluster cannot find a shared disk. This is accompanied by the `syslog` message:

```
VxVM vxconfigd ERROR V-5-1-2173 cannot find disk disk
```

Recommended action:

Make sure that the same set of shared disks is online on both nodes. Examine the disks on both the master and the slave with the command `vxdisk list` and make sure that the same set of disks with the `shared` flag is visible on both nodes. If not, check the connections to the disks.

V-5-1-3023

VxVM vxconfigd ERROR V-5-1-3023 Disk in use by another cluster

Description:

An attempt was made to import a disk group whose disks are stamped with the ID of another cluster.

Recommended action:

If the disk group is not imported by another cluster, retry the import using the `-c` (clear import) flag.

V-5-1-3024

VxVM vxconfigd ERROR V-5-1-3024 vxclust not there

Description:

An error during an attempt to join a cluster caused `vxclust` to fail. This may be caused by the failure of another node during a join or by the failure of `vxclust`.

Recommended action:

Retry the join. An error message on the other node may clarify the problem.

V-5-1-3025

VxVM vxconfigd ERROR V-5-1-3025 Unable to add portal for cluster

Description:

`vxconfigd` was not able to create a portal for communication with the `vxconfigd` on the other node. This may happen in a degraded system that is experiencing shortages of system resources such as memory or file descriptors.

Recommended action:

If the system does not appear to be degraded, stop and restart `vxconfigd`, and try again.

V-5-1-3030

VxVM vxconfigd ERROR V-5-1-3030 Volume recovery in progress

Description:

A node that crashed attempted to rejoin the cluster before its DRL map was merged into the recovery map.

Recommended action:

Retry the join when the merge operation has completed.

V-5-1-3031

VxVM vxconfigd ERROR V-5-1-3031 Cannot assign minor *minor*

Description:

A slave attempted to join a cluster, but an existing volume on the slave has the same minor number as a shared volume on the master.

This message is accompanied by the following console message:

```
VxVM vxconfigd ERROR V-5-1-2192 minor number minor disk group group
in use
```

Recommended action:

Before retrying the join, use `vx dg remminor` (see the `vx dg(1M)` manual page) to choose a new minor number range either for the disk group on the master or for the conflicting disk group on the slave. If there are open volumes in the disk group, the `remminor` operation will not take effect until the disk group is deported and updated (either explicitly or by rebooting the system).

V-5-1-3032

VxVM vxconfigd ERROR V-5-1-3032 Master sent no data

Description:

During the slave join protocol, a message without data was received from the master. This message is only likely to be seen in the case of an internal VxVM error.

Recommended action:

Contact Veritas Technical Support.

V-5-1-3033

VxVM vxconfigd ERROR V-5-1-3033 Join in progress

Description:

An attempt was made to import or deport a shared disk group during a cluster reconfiguration.

Recommended action:

Retry when the cluster reconfiguration has completed.

V-5-1-3034

VxVM vxconfigd ERROR V-5-1-3034 Join not currently allowed

Description:

A slave attempted to join a cluster when the master was not ready. The slave will retry automatically.

Recommended action:

No action is necessary if the join eventually completes. Otherwise, investigate the cluster monitor on the master.

V-5-1-3042

VxVM vxconfigd ERROR V-5-1-3042 Clustering license restricts operation

Description:

An operation requiring a full clustering license was attempted, and such a license is not available.

Recommended action:

If the error occurs when a disk group is being activated, dissociate all but one plex from mirrored volumes before activating the disk group. If the error occurs during a transaction, deactivate the disk group on all nodes except the master.

V-5-1-3046

VxVM vxconfigd ERROR V-5-1-3046 Node activation conflict

Description:

The disk group could not be activated because it is activated in a conflicting mode on another node in a cluster.

Recommended action:

Retry later, or deactivate the disk group on conflicting nodes.

V-5-1-3049

VxVM vxconfigd ERROR V-5-1-3049 Retry rolling upgrade

Description:

An attempt was made to upgrade a cluster to a higher protocol version when a transaction was in progress.

Recommended action:

Retry the upgrade at a later time.

V-5-1-3050

VxVM vxconfigd ERROR V-5-1-3050 Version out of range for at least one node

Description:

Before trying to upgrade a cluster by running `vxdctl upgrade`, all nodes should be able to support the new protocol version. An upgrade can fail if at least one of them does not support the new protocol version.

Recommended action:

Make sure that the Veritas Volume Manager package that supports the new protocol version is installed on all nodes and retry the upgrade.

V-5-1-3091

VxVM vxdg ERROR V-5-1-3091 *diskname* : Disk not moving, but subdisks on it are

Description:

Some volumes have subdisks that are not on the disks implied by the supplied list of objects.

Recommended action:

Use the `-o expand` option to `vxvg listmove` to produce a self-contained list of objects.

V-5-1-3212

VxVM vxconfigd ERROR V-5-1-3212 Insufficient DRL log size: logging is disabled.

Description:

A volume with an insufficient DRL log size was started successfully, but DRL logging is disabled and a full recovery is performed.

Recommended action:

Create a new DRL of sufficient size.

V-5-1-3243

VxVM vxddmpadm ERROR V-5-1-3243 The VxVM restore daemon is already running. You can stop and restart the restore daemon with desired arguments for changing any of its parameters.

Description:

The `vxddmpadm start restore` command has been executed while the restore daemon is already running.

Recommended action:

Stop the restore daemon and restart it with the required set of parameters.
See the `vxddmpadm(1M)` manual page.

V-5-1-3362

VxVM vxddmpadm ERROR V-5-1-3362 Attempt to disable controller failed. One (or more) devices can be accessed only through this controller. Use the `-f` option if you still want to disable this controller.

Description:

Disabling the controller could lead to some devices becoming inaccessible.

Recommended action:

To disable the only path connected to a disk, use the `-f` option.

V-5-1-3486

VxVM vxconfigd ERROR V-5-1-3486 Not in cluster

Description:

Checking for the current protocol version (using `vxctl protocol version`) does not work if the node is not in a cluster.

Recommended action:

Bring the node into the cluster and retry.

V-5-1-3689

```
VxVM vxassist ERROR V-5-1-3689 Volume record id rid is not found in the configuration.
```

Description:

An error was detected while reattaching a snapshot volume using `snapback`. This happens if a volume's record identifier (rid) changes as a result of a disk group split that moved the original volume to a new disk group. The snapshot volume is unable to recognize the original volume because its record identifier has changed.

Recommended action:

Use the following command to perform the `snapback`:

```
# vxplex [-g diskgroup] -f snapback volume plex
```

V-5-1-3828

```
VxVM vxconfigd ERROR V-5-1-3828 upgrade operation failed: Already at highest version
```

Description:

An upgrade operation has failed because a cluster is already running at the highest protocol version supported by the master.

Recommended action:

No further action is possible as the master is already running at the highest protocol version it can support.

V-5-1-3848

```
VxVM vxconfigd ERROR V-5-1-3848 Incorrect protocol version (number) in volboot file
```

Description:

A node attempted to join a cluster where VxVM software was incorrectly upgraded or the `volboot` file is corrupted, possibly by being edited manually. The `volboot`

file should contain a supported protocol version before trying to bring the node into the cluster.

Recommended action:

Verify the supported cluster protocol versions using the `vxctl protocolversion` command. The `volboot` file should contain a supported protocol version before trying to bring the node into the cluster. Run `vxctl init` to write a valid protocol version to the `volboot` file. Restart `vxconfigd` and retry the join.

V-5-1-4220

```
VxVM vxconfigd ERROR V-5-1-4220 DG move: can't import diskgroup,
giving up
```

Description:

The specified disk group cannot be imported during a disk group move operation. (The disk group ID is obtained from the disk group that could be imported.)

Recommended action:

The disk group may have been moved to another host. One option is to locate it and use the `vxvg recover` command on both the source and target disk groups. Specify the `-o clean` option with one disk group, and the `-o remove` option with the other disk group.

See [“Recovering from an incomplete disk group move”](#) on page 29.

V-5-1-4267

```
VxVM vxassist WARNING V-5-1-4267 volume volume already has at least
one snapshot plex Snapshot volume created with these plexes will have
a dco volume with no associated dco plex.
```

Description:

An error was detected while adding a DCO object and DCO volume to a mirrored volume. There is at least one snapshot plex already created on the volume. Because this snapshot plex was created when no DCO was associated with the volume, there is no DCO plex allocated for it.

Recommended action:

For information on adding DCO volumes and volume snapshots, see the *Veritas Volume Manager Administrator's Guide*.

V-5-1-4277

VxVM vxconfigd ERROR V-5-1-4277 cluster_establish: CVM protocol version out of range

Description:

When a node joins a cluster, it tries to join at the protocol version that is stored in its `volboot` file. If the cluster is running at a different protocol version, the master rejects the join and sends the current protocol version to the slave. The slave re-tries with the current version (if that version is supported on the joining node), or the join fails.

Recommended action:

Make sure that the joining node has a Veritas Volume Manager release installed that supports the current protocol version of the cluster.

V-5-1-4551

VxVM vxconfigd ERROR V-5-1-4551 dg_move_recover: can't locate disk(s), giving up

Description:

Disks involved in a disk group move operation cannot be found, and one of the specified disk groups cannot be imported.

Recommended action:

Manual use of the `vx dg recover` command may be required to clean the disk group to be imported.

See [“Recovering from an incomplete disk group move”](#) on page 29.

V-5-1-4620

VxVM vxassist WARNING V-5-1-4620 Error while retrieving information from SAL

Description:

The `vxassist` command does not recognize the version of the SAN Access Layer (SAL) that is being used, or detects an error in the output from SAL.

Recommended action:

If a connection to SAL is desired, ensure that the correct version of SAL is installed and configured correctly. Otherwise, suppress communication between `vxassist` and SAL by adding the following line to the `vxassist` defaults file (usually `/etc/default/vxassist`):

```
salcontact=no
```

V-5-1-4625

```
VxVM vxassist WARNING V-5-1-4625 SAL authentication failed...
```

Description:

The SAN Access Layer (SAL) rejects the credentials that are supplied by the `vxassist` command.

Recommended action:

If connection to SAL is desired, use the `vxspcshow` command to set a valid user name and password. Otherwise, suppress communication between `vxassist` and SAL by adding the following line to the `vxassist` defaults file (usually `/etc/default/vxassist`):

```
salcontact=no
```

V-5-1-5150

```
VxVM vxassist ERROR V-5-1-5150 Insufficient number of active snapshot mirrors in snapshot_volume.
```

Description:

An attempt to snap back a specified number of snapshot mirrors to their original volume failed.

Recommended action:

Specify a number of snapshot mirrors less than or equal to the number in the snapshot volume.

V-5-1-5160

```
VxVM vxplex ERROR V-5-1-5160 Plex plex not associated to a snapshot volume.
```

Description:

An attempt was made to snap back a plex that is not from a snapshot volume.

Recommended action:

Specify a plex from a snapshot volume.

V-5-1-5161

```
VxVM vxplex ERROR V-5-1-5161 Plex plex not attached.
```

Description:

An attempt was made to snap back a detached plex.

Recommended action:

Reattach the snapshot plex to the snapshot volume.

V-5-1-5162

VxVM vxplex ERROR V-5-1-5162 Plexes do not belong to the same snapshot volume.

Description:

An attempt was made to snap back plexes that belong to different snapshot volumes.

Recommended action:

Specify the plexes in separate invocations of `vxplex snapback`.

V-5-1-5929

VxVM vxconfigd NOTICE V-5-1-5929 Unable to resolve duplicate diskid.

Description:

VxVM has detected disks with duplicate disk identifiers. Arrays with mirroring capability in hardware are particularly susceptible to such data corruption, but other causes are possible as explained below.

In releases prior to 3.5, VxVM selected the first disk that it found if the selection process failed. From release 3.5, the default behavior of VxVM was to avoid the selection of the wrong disk as this could lead to data corruption.

If VxVM could not determine which disk was the original, it would not import the disks until they were reinitialized with a new disk ID.

From release 5.0, VxVM checks the unique disk identifier (UDID) value that is known to the Device Discovery Layer (DDL) against the UDID value that is set in the disk's private region. The `udid_mismatch` flag is set on the disk if the values differ. If set, this flag is displayed in the output from the `vxdisk list` command.

A new set of `vxdisk` and `vxdbg` operations are provided to handle such disks; either by writing the DDL value of the UDID to a disk's private region, or by tagging a disk and specifying that it is a cloned disk to the `vxdbg import` operation.

Recommended action:

User intervention is required in the following cases:

- Case 1: Some arrays such as EMC and HDS provide mirroring in hardware. When a LUN pair is split, depending on how the process is performed, this can result in two disks that have the same disk identifier and UDID value. See "Handling Disks with Duplicated Identifiers" in the "Creating and Administering Disk Groups" chapter of the *Veritas Volume Manager Administrator's Guide* for full details of how to deal with this condition.
- Case 2: If disks have been duplicated by using the `dd` command or any similar copying utility, you can use the following command to update the UDID for one or more disks:

```
# vxdisk [-f] updateudid disk1 ...
```

This command uses the current value of the UDID that is stored in the Device Discovery Layer (DDL) database to correct the value in the private region. The `-f` option must be specified if VxVM has not set the `udid_mismatch` flag on a disk.

For example, the following command updates the UDIDs for the disks `c2t66d0` and `c2t67d0`:

```
# vxdisk updateudid c2t66d0 c2t67d0
```

- Case 3: If DMP has been disabled to an array that has multiple paths, then each path to the array is claimed as a unique disk.
If DMP is suppressed, VxVM does not know which path to select as the true path. You must choose which path to use. Decide which path to exclude, and then select item 1 (suppress all paths through a controller from VxVM's view) or item 2 (suppress a path from VxVM's view) from `vxdiskadm option 17` (Prevent multipathing/Suppress devices from VxVM's view).
- Case 4: (Solaris 9 only) After suppressing one path to a multi-pathed array from DMP, you then have suppressed that path from VxVM using item 1 (suppress all paths through a controller from VxVM's view) from `vxdiskadm option 17` (Prevent multipathing/Suppress devices from VxVM's view). If all `bootdg` disks are from that array, `vxconfigd` does not start. (You may also see errors V-5-0-64, V-5-1-946 and V-5-1-2841.)
If only one array is connected to the controller, suppress the path from DMP using item 5 (Prevent multipathing of all disks on a controller by VxVM), and then item 1 (Suppress all paths through a controller from VxVM's view), from `vxdiskadm option 17` (Prevent multipathing/Suppress devices from VxVM's view).
If more than one array is connected to the controller, suppress the path from DMP using item 5 (Prevent multipathing of all disks on a controller by VxVM), and then item 2 (Suppress a path from VxVM's view), from

`vxdiskadm option 17` (Prevent multipathing/Suppress devices from VxVM's view).

V-5-2-2400

VxVM `vxdisksetup` NOTICE V-5-2-2400 *daname*: Duplicate DA records encountered for this device.

Description:

This may occur if a disk has been replaced in a Sun StorEdge™ A5x00 (or similar) array without using the `luxadm` command to notify the operating system. Duplicated disk access record entries similar to the following appear in the output from the `vxdisk list` command:

```
c1t5d0s2      sliced      c1t5d0s2      -      error
c1t5d0s2      sliced      c1t5d0s2      -      online|error
```

The status of the second entry may be `online` or `error`.

Recommended action:

To correct the problem

- 1 Use the following command for each of the duplicated disk access names to remove them all from VxVM control:

```
# vxdisk rm daname
```

Run the `vxdisk list` command to ensure that you have removed all entries for the disk access name.

- 2 Use the Solaris `luxadm` command to obtain the A5K array name and slot number of the disk, and then use these with `luxadm` to remove the disk:

```
# luxadm disp /dev/rdsk/daname
# luxadm remove_device array_name,slot_number
```

- 3 Pull out the disk as instructed to by `luxadm`.
- 4 Run the following commands:

```
# devfsadm -C
# vxctl enable
```

This completes the removal of device entries corresponding to the physical disk.

- 5 As in step 1, run the `vxdisk list` and `vxdisk rm` commands to list any remaining entries for the disk access name and remove these from VxVM control.
- 6 Use the following `luxadm` command to take the disk offline on all the paths to it. The following example shows how to take a disk offline that is connected to controllers `c1` and `c2`:

```
# luxadm -e offline /dev/dsk/c1t5d0s2
# luxadm -e offline /dev/dsk/c2t5d0s2
```

- 7 Repeat step 5 and step 6 until no entries remain for the disk in the output from `vxdisk list`. This completes the removal of stale device entries for the disk from VxVM.
- 8 Finally, replace the failed or removed disk.
See the *Veritas Volume Manager Administrator's Guide*.

V-5-1-15728

VxVM command ERROR V-5-1-15728 `vxconfigd` died during command shipping. Check the status of command execution manually.

Description:

During execution of a shared disk group configuration change command that is shipped from slave on master, the `vxconfigd` process exited on the slave node.

Recommended action:

Verify the state of the VxVM objects using query commands such as `vxprint` or `vxdisk list`. Since the command exited during execution, the command might have left some stale entries. Clear the stale entries using commands such as `vxedit` or `vxmend`. Verify the running state of the `vxconfigd` process. If necessary, restart the `vxconfigd` process.

V-5-1-15861

VxVM command ERROR V-5-1-15861 Command is not supported for command shipping. Operation must be executed on master.

Description:

On the slave, when you run a command that changes the configuration for a shared disk group, CVM ships the command to the master for execution. The message occurs when you run a command on the slave node that is not supported for

shipping to the master. The commands that are not supported are listed in the file `/etc/vx/vxcommandship.exclude`.

For a list of CVM commands that are supported for executing on the slave node, see the *Veritas Volume Manager Administrator's Guide*.

Recommended action:

Run the command on the CVM master node.

V-5-1-15917

VxVM command INFO V-5-1-15917 Command is executing on master through command shipping. Attempting to kill remote command with client id:`clid`

Description:

On the slave node, if you run a command that involves configuration changes for a shared disk group, CVM ships the command to the master node and executes the command on the master. The message occurs on the slave node, when you issue CTRL-C to terminate a shipped command that is currently executing on the master node.

Recommended action:

Verify the state of the VxVM objects using query commands such as `vxprint` or `vxdisk list`. Since the command exited during execution, the command might have left some stale entries. Clear the stale entries using commands such as `vxedit` or `vxmend`.

V-5-1-0

VxVM command ERROR V-5-1-0 Command terminated during command shipping. Check status manually.

Description:

The message occurs during execution of a shared disk group configuration change command that is shipped from slave on master. The message indicates that either the `vxconfigd` process exited on the master node, the master node left the cluster, or the slave node that issued the command left the cluster.

Recommended action:

Verify the state of the VxVM objects using query commands such as `vxprint` or `vxdisk list`. Since the command exited during execution, the command might have left some stale entries. Clear the stale entries using commands such as `vxedit`

or `vxmend`. Verify the running state of the `vxconfigd` process. If necessary, restart the `vxconfigd` process.

Index

Symbols

- .cmdlog file 81
- .translog file 83
- /etc/init.d/vxvm-sysboot file 100
- /etc/system file
 - missing or damaged 59
 - restoring 59–60
- /etc/vfstab file
 - damaged 57
 - purpose 57
- /etc/vx/cbr/bk/diskgroup.dgid
 - dgid .binconfig file 91
 - dgid .cfgrec file 91
 - dgid .diskinfo file 91
 - dgid.dginfo file 91
- /etc/vx/log logging directory 81, 83
- /lib/svc/method/vxvm-sysboot file 100
- /var/log/syslog file 100
- /var/vxvm/configd.log file 99

A

- ACTIVE plex state 13
- ACTIVE volume state 23
- aliased disks 47

B

- badlog flag
 - clearing for DCO 32
- BADLOG plex state 22
- boot command
 - a flag 45, 59
 - s flag 60, 63
 - syntax 45
- boot disks
 - alternate 45
 - configurations 44
 - hot-relocation 53
 - listing aliases 47
 - re-adding 65–66
 - recovery 43

- boot disks (*continued*)
 - relocating subdisks 54
 - replacing 65
 - using aliases 46
- boot failure
 - cannot open altboot_disk 46
 - cannot open boot device 55
 - damaged /usr entry 58
 - due to stale plexes 55
 - due to unusable plexes 55
 - invalid partition 57
- booting system
 - aliased disks 47
 - recovery from failure 54
 - using CD-ROM 63

C

- CD-ROM
 - booting 63
- CLEAN plex state 13
- client ID
 - in command logging file 81
 - in transaction logging file 83
- cmdlog file 81
- commands
 - associating with transactions 85
 - logging 81
- configuration
 - backing up for disk groups 89–90
 - backup files 90
 - resolving conflicting backups 93
 - restoring for disk groups 89, 91
- copy-on-write
 - recovery from failure of 41

D

- data loss
 - RAID-5 20
- DCO
 - recovering volumes 30
 - removing badlog flag from 32

- DCO volumes
 - recovery from I/O failure on 42
- debug message
 - logging 99
- degraded mode
 - RAID-5 21
- DEGRADED volume state 21
- detached RAID-5 log plexes 25
- DETACHED volume kernel state 23
- devalias command 47
- DISABLED plex kernel state 13, 23
- disabling VxVM 64
- disk group errors
 - new host ID 125
- disk groups
 - backing up configuration of 89–90
 - configuration backup files 90
 - recovering from failed move
 - split or join 29
 - resolving conflicting backups 93
 - restoring configuration of 89, 91
- disk IDs
 - fixing duplicate 155
- disks
 - aliased 47
 - causes of failure 11
 - failing flag 18
 - failures 21
 - fixing duplicated IDs 155
 - invalid partition 57
 - reattaching 19
 - reattaching failed 19
- DMP
 - fixing duplicated disk IDs 155
- dumpadm command 54

E

- eeeprom
 - using boot disk aliases 46
- EEPROM variables
 - use-nvramrc? 46
- EMPTY plex state 13
- ENABLED plex kernel state 13
- ENABLED volume kernel state 23
- error message
 - enable failed 141
- ERROR messages 103
- error messages
 - A virtual disk device is open 121

- error messages (*continued*)
 - All transactions are disabled 117
 - Already at highest version 151
 - Attempt to disable controller failed 150
 - Attempt to enable a controller that is not available 139
 - Cannot assign minor 147
 - Cannot auto-import group 89, 126
 - Cannot create /var/adm/utmp or /var/adm/utmpx 57
 - Cannot find disk on slave node 146
 - Cannot kill existing daemon 137
 - cannot open /dev/vx/config 119
 - Cannot open /etc/vfstab 120
 - cannot open altboot_disk 46
 - Cannot recover temp database 139
 - Cannot remove last disk group configuration copy 144
 - Cannot reset VxVM kernel 121
 - Cannot start volume 116, 121
 - can't import diskgroup 152
 - Can't locate disk(s) 153
 - Can't open boot device 55
 - Clustering license restricts operation 148
 - Configuration records are inconsistent 126
 - Configuration too large for configuration copies 144
 - CVM protocol version out of range 153
 - daemon count must be above number while clustered 140
 - default log file 99
 - Device is already open 119
 - Differing version of vxconfigd installed 123
 - Disabled by errors 89, 117
 - Disk for disk group not found 129
 - Disk group does not exist 29, 143
 - Disk group errors, multiple disk failures 117
 - Disk group has no valid configuration copies 126, 129
 - Disk group version doesn't support feature 141
 - Disk in use by another cluster 146
 - Disk is in use by another host 129
 - Disk is not usable 144
 - Disk not moving but subdisks on it are 149
 - Disk reserved by another host 141
 - Disk write failure 125
 - Duplicate DA records encountered for this device 157
 - Duplicate record in configuration 126

error messages (*continued*)

- enable failed 136–137
- Error in cluster processing 145
- Error in disk group configuration copies 141
- Errors in some configuration copies 126, 128, 137
- failed write of utmpx entry 57
- File just loaded does not appear to be executable 57
- Format error in configuration copy 126
- group exists 144
- import failed 129
- Incorrect protocol version in volboot file 151
- Insufficient DRL log size logging is disabled 150
- Insufficient number of active snapshot mirrors in snapshot_volume 154
- Invalid block number 126
- Invalid magic number 126
- Join in progress 148
- Join not allowed now 148
- logging 99
- Master sent no data 147
- Memory allocation failure 118
- Missing vxconfigd 138
- Name conflicts with imported diskgroup 145
- No more space in disk group configuration 145
- No such device or address 119
- No such file or directory 119
- no valid complete plexes 121
- No valid disk found containing disk group 129
- no valid plexes 116
- Node activation conflict 149
- Not in cluster 150
- not updated with new host ID 125
- Operation is not supported 142
- Plex plex not associated with a snapshot volume 154
- Plex plex not attached 154
- Plexes do not belong to the same snapshot volume 155
- RAID-5 plex does not map entire volume length 27
- Record already exists in disk group 142
- Record is associated 143
- Record volume is in disk group diskgroup1 plex is in group diskgroup2 132–133
- Reimport of disk group failed 129
- Request crosses disk group boundary 143
- Retry rolling upgrade 149

error messages (*continued*)

- Return from cluster_establish is Configuration daemon error 140
- Skip disk group with duplicate name 127
- some subdisks are unusable and the parity is stale 27
- startup script 100
- Synchronization of the volume stopped due to I/O error 42
- System boot disk does not have a valid root plex 56
- System boot disk does not have a valid rootvol plex 133
- System startup failure 56, 119
- The VxVM restore daemon is already running 150
- There are two backups that have the same diskgroup name with different diskgroup id 93
- There is no volume configured for the root device 134
- Transaction already in progress 142
- transactions are disabled 141
- Unable to add portal for cluster 146
- Unexpected kernel error in configuration update 141
- Unrecognized operating mode 114
- update failed 118
- upgrade operation failed 151
- Version number of kernel does not match vxconfigd 135
- Version out of range for at least one node 149
- Vol recovery in progress 147
- Volume for mount point /usr not found in rootdg disk group 135
- Volume is not startable 27
- volume not in rootdg disk group 131
- Volume or plex device is open or mounted 143
- Volume record id is not found in the configuration 151
- volume state is invalid 121
- vxclust not there 146
- vxconfigd cannot boot-start RAID-5 volumes 138
- vxconfigd minor number in use 147
- vxconfigd not ready 138

F

- failing flag
 - clearing 18
- failures
 - disk 21
 - system 20
- FATAL ERROR messages 103
- fatal error messages
 - bootdg cannot be imported during boot 133
 - Memory allocation failure during startup 119
- files
 - disk group configuration backup 90

H

- hardware failure
 - recovery from 11
- hot-relocation
 - boot disks 53
 - defined 11
 - RAID-5 23
 - root disks 53

I

- INFO messages 103
- install-db file 64, 71
- IOFAIL plex state 13

L

- listing
 - alternate boot disks 45
 - unstartable volumes 12
- log file
 - default 99
 - syslog error messages 100
 - vxconfigd 99
- LOG plex state 22
- log plexes
 - importance for RAID-5 20
 - recovering RAID-5 25
- logging
 - associating commands and transactions 85
 - directory 81, 83
- logging debug error messages 99

M

- mirrored volumes
 - recovering 16

MOVE flag

- set in TUTILO field 29

N

- NEEDSYNC volume state 24
- NOTICE messages 103
- notice messages
 - added disk array 105
 - Attempt to disable controller failed 105
 - Detached disk 117
 - Detached log for volume 122
 - Detached plex in volume 122
 - Detached subdisk in volume 122
 - Detached volume 123
 - disabled controller connected to disk array 106
 - disabled dmpnode 107
 - disabled path belonging to dmpnode 107
 - enabled controller connected to disk array 108
 - enabled dmpnode 108
 - enabled path belonging to dmpnode 109
 - Offlining config copy 132
 - Path failure 112
 - read error on object 113
 - removed disk array 114
 - Rootdisk has just one enabled path 105
 - Unable to resolve duplicate diskid 155
 - Volume entering degraded mode 120

O

- OpenBoot PROMs (OPB) 45

P

- PANIC messages 102
- parity
 - regeneration checkpointing 24
 - resynchronization for RAID-5 24
 - stale 20
- partitions
 - invalid 57
- plex kernel states
 - DISABLED 13, 23
 - ENABLED 13
- plex states
 - ACTIVE 13
 - BADLOG 22
 - CLEAN 13
 - EMPTY 13
 - IOFAIL 13

- plex states (*continued*)
 - LOG 22
 - STALE 16
- plexes
 - defined 13
 - displaying states of 13
 - in RECOVER state 17
 - mapping problems 27
 - recovering mirrored volumes 16
- primary boot disk failure 45
- process ID
 - in command logging file 81
 - in transaction logging file 83
- PROMs
 - boot 45

R

- RAID-5
 - detached subdisks 21
 - failures 20
 - hot-relocation 23
 - importance of log plexes 20
 - parity resynchronization 24
 - recovering log plexes 25
 - recovering volumes 23
 - recovery process 23
 - stale parity 20
 - starting forcibly 28
 - starting volumes 27
 - startup recovery process 23
 - subdisk move recovery 26
 - unstartable volumes 27
- reattaching disks 19
- reconstructing-read mode
 - stale subdisks 21
- RECOVER state 17
- recovery
 - disk 19
- reinstalling entire system 68
- replacing
 - boot disks 65
- REPLAY volume state 23
- restarting disabled volumes 17
- resynchronization
 - RAID-5 parity 24
- root disks
 - booting alternate 45
 - configurations 44
 - hot-relocation 53

- root disks (*continued*)
 - re-adding 65–66
 - recovery 43
 - repairing 61
 - replacing 65
- root file system
 - backing up 62
 - configurations 44
 - damaged 68
 - restoring 63

S

- snapshot resynchronization
 - recovery from errors during 42
- stale parity 20
- states
 - displaying for volumes and plexes 13
- subdisks
 - marking as non-stale 28
 - recovering after moving for RAID-5 26
 - stale
 - starting volume 28
 - unrelocating to replaced boot disk 54
- swap space
 - configurations 44
- SYNC volume state 23–24
- syslog
 - error log file 100
- system
 - reinstalling 68
- system failures 20

T

- transactions
 - associating with commands 85
 - logging 83
- translog file 83
- TUTIL0 field
 - clearing MOVE flag 29

U

- udid_mismatch flag 155
- ufsdump 62
- ufsrestore
 - restoring a UFS file system 63
- use-nvramrc? 46
- usr file system
 - backing up 62

usr file system *(continued)*
 configurations 44
 repairing 61
 restoring 63

V

V-5-0-106 106
 V-5-0-108 106
 V-5-0-110 106
 V-5-0-111 107
 V-5-0-112 107
 V-5-0-144 107
 V-5-0-145 108
 V-5-0-146 108
 V-5-0-147 108
 V-5-0-148 109
 V-5-0-164 109
 V-5-0-166 109
 V-5-0-168 110
 V-5-0-181 110
 V-5-0-194 110
 V-5-0-196 111
 V-5-0-2 104
 V-5-0-207 111
 V-5-0-216 111
 V-5-0-237 111
 V-5-0-243 112
 V-5-0-244 112
 V-5-0-249 112
 V-5-0-251 113
 V-5-0-252 113
 V-5-0-258 114
 V-5-0-281 114
 V-5-0-34 105
 V-5-0-35 105
 V-5-0-386 114
 V-5-0-4 104
 V-5-0-55 105
 V-5-0-64 106
 V-5-1-1049 56, 133
 V-5-1-1063 134
 V-5-1-111 116
 V-5-1-116 116
 V-5-1-117 116
 V-5-1-1171 135
 V-5-1-1186 135
 V-5-1-121 117
 V-5-1-122 117
 V-5-1-123 89, 117
 V-5-1-1236 27
 V-5-1-1237 27
 V-5-1-124 118
 V-5-1-134 118
 V-5-1-135 119
 V-5-1-148 119
 V-5-1-1589 136
 V-5-1-169 119
 V-5-1-2020 137
 V-5-1-2173 146
 V-5-1-2192 147
 V-5-1-2197 138
 V-5-1-2198 138
 V-5-1-2274 138
 V-5-1-2290 139
 V-5-1-2353 139
 V-5-1-249 120
 V-5-1-2524 140
 V-5-1-2630 140
 V-5-1-2824 140
 V-5-1-2829 141
 V-5-1-2830 141
 V-5-1-2841 141
 V-5-1-2860 142
 V-5-1-2862 142
 V-5-1-2866 142
 V-5-1-2870 143
 V-5-1-2879 143
 V-5-1-2907 29, 143
 V-5-1-2908 143
 V-5-1-2911 144
 V-5-1-2922 144
 V-5-1-2928 144
 V-5-1-2933 144
 V-5-1-2935 145
 V-5-1-3009 145
 V-5-1-3020 145
 V-5-1-3022 146
 V-5-1-3023 146
 V-5-1-3024 146
 V-5-1-3025 146
 V-5-1-3030 147
 V-5-1-3031 147
 V-5-1-3032 147
 V-5-1-3033 148
 V-5-1-3034 148
 V-5-1-3042 148
 V-5-1-3046 149
 V-5-1-3049 149

- V-5-1-3050 149
- V-5-1-3091 149
- V-5-1-3212 150
- V-5-1-3243 150
- V-5-1-3362 150
- V-5-1-3486 150
- V-5-1-3689 151
- V-5-1-3828 151
- V-5-1-3848 151
- V-5-1-4220 152
- V-5-1-4267 152
- V-5-1-4277 153
- V-5-1-4551 153
- V-5-1-4620 153
- V-5-1-4625 154
- V-5-1-480 121
- V-5-1-484 121
- V-5-1-485 121
- V-5-1-5150 154
- V-5-1-5160 154
- V-5-1-5161 154
- V-5-1-5162 155
- V-5-1-525 122
- V-5-1-526 122
- V-5-1-527 122
- V-5-1-528 123
- V-5-1-543 123
- V-5-1-544 123
- V-5-1-545 124
- V-5-1-546 124
- V-5-1-554 125
- V-5-1-557 125
- V-5-1-568 126
- V-5-1-569 89, 126
- V-5-1-571 127
- V-5-1-577 128
- V-5-1-579 128, 137
- V-5-1-583 129
- V-5-1-587 129
- V-5-1-5929 155
- V-5-1-6012 93
- V-5-1-663 131
- V-5-1-6840 42
- V-5-1-737 131
- V-5-1-768 132
- V-5-1-90 114
- V-5-1-91 115
- V-5-1-92 115
- V-5-1-923 132–133
- V-5-1-946 133
- V-5-2-2400 157
- VM disks
 - aliased 47
- volume kernel states
 - DETACHED 23
 - ENABLED 23
- volume states
 - ACTIVE 23
 - DEGRADED 21
 - NEEDSYNC 24
 - REPLAY 23
 - SYNC 23–24
- volumes
 - displaying states of 13
 - listing unstartable 12
 - RAID-5 data loss 20
 - recovering for DCO 30
 - recovering mirrors 16
 - recovering RAID-5 23
 - restarting disabled 17
 - stale subdisks
 - starting 28
- vxcmdlog
 - controlling command logging 81
- vxconfigbackup
 - backing up disk group configuration 90
- vxconfigd
 - log file 99
- vxconfigd.log file 99
- vxconfigrestore
 - restoring a disk group configuration 91
- vxdco
 - removing badlog flag from DCO 32
- vxctl
 - changing level of debug logging 99
- vxdg
 - recovering from failed disk group move
 - split or join 29
- vxdisk
 - updating the disk identifier 155
- vxedit
 - clearing a disk failing flag 18
- vxinfo command 12
- vxmend command 16
- vxplex command 25
- vxprint
 - displaying volume and plex states 13

- vxreattach
 - reattaching failed disks 19
- vxsnap make
 - recovery from failure of 38
- vxsnap prepare
 - recovery from failure of 37
- vxsnap reattach
 - recovery from failure of 40
- vxsnap refresh
 - recovery from failure of 40
- vxsnap restore
 - recovery from failure of 40
- vxtranslog
 - controlling transaction logging 83
- vxunreloc command 54
- VxVM
 - disabling 64
 - RAID-5 recovery process 23
 - recovering configuration of 71
- vxvol recover command 26
- vxvol resync command 24
- vxvol start command 16

W

- WARNING messages 103
- warning messages
 - Cannot create device 115
 - Cannot exec /usr/bin/rm to remove directory 115
 - Cannot find device number 105
 - Cannot fork to remove directory 116
 - cannot log commit record for Diskgroup bootdg 106
 - Cannot open log file 116
 - corrupt label_sdo 57
 - Detaching plex from volume 117
 - detaching RAID-5 106
 - Disk device not found 124
 - Disk group is disabled 126
 - Disk group log may be too small 128
 - Disk in group flagged as shared 123
 - Disk in group locked by host 124
 - Disk names group but group ID differs 125
 - Disk skipped 123
 - disks not updated with new host ID 126
 - Double failure condition detected on RAID-5 107
 - Duplicate virtual device number(s) 131
 - error 28 106

- warning messages (*continued*)
 - Error while retrieving information from SAL 153
 - Failed to join cluster 109
 - Failed to log the detach of the DRL volume 109
 - Failure in RAID-5 logging operation 110
 - Illegal vminor encountered 110
 - Kernel log full 110
 - Kernel log update failed 111
 - library and vxconfigd disagree on existence of client 140
 - log object detached from RAID-5 volume 111
 - Log size should be at least 128
 - mod_install returned errno 111
 - object detached from RAID-5 volume 111
 - object plex detached from volume 104
 - Overlapping mirror plex detached from volume 112
 - Plex for root volume is stale or unusable 56
 - RAID-5 volume entering degraded mode operation 112
 - read error on mirror plex of volume 113
 - Received spurious close 106
 - Root volumes are not supported on your PROM version 114
 - SAL authentication failed 154
 - subdisk failed in plex 114
 - unable to read label 57
 - Uncorrectable read error 104
 - Uncorrectable write error 104
 - volume already has at least one snapshot plex 152
 - volume is detached 108
 - Volume remapped 131