

VERITAS Volume Manager™ 3.1

Reference Guide Solaris

July 2000
30-000001-399


VERITAS

Disclaimer

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

Copyright

Copyright © 1999-2000 VERITAS Software Corporation. All rights reserved. VERITAS is a registered trademark of VERITAS Software Corporation in the US and other countries. The VERITAS logo and VERITAS Volume Manager are trademarks of VERITAS Software Corporation. All other trademarks or registered trademarks are the property of their respective owners.

Printed in the USA, July 2000.

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



Contents

Preface	vii
Audience	vii
Organization	vii
Related Documents	vii
Conventions	viii
Chapter 1. Volume Manager Commands	9
Introduction	9
Using vxassist to Create and Change Volumes	10
Advantages of the vxassist Command	10
How vxassist Works	11
vxassist Defaults	11
Defaults File	12
Creating Volume Manager Objects With vxmake	14
Performing Volume Tasks With vxvol	15
Performing Plex Tasks With vxplex	15
Performing Subdisk Tasks With vxsd	16
Using vxdiskadm to Perform Disk Tasks	16
Adding Disks With vxdiskadd	16
Performing Disk Tasks With vxdisk	17
Performing Disk Group Tasks With vxdg	17
Moving Relocated Subdisks With vxunrelocate	17
Changing Volume Manager Objects With vxedit	17
Correcting Problems With vxmend	18



Printing Configuration Information With vxprint	18
Printing Volume Statistics With vxstat	18
Tracing Volume Tasks With vxtrace	18
Using vxctl to Manage the vxconfigd Daemon	19
Chapter 2. Volume Manager Error Messages	21
Introduction	21
Logging Error Messages	22
Volume Manager Configuration Daemon Error Messages	24
vxconfigd Usage Messages	24
vxconfigd Error Messages	25
vxconfigd Fatal Error Messages	46
vxconfigd Notice Messages	48
vxconfigd Warning Messages	50
Kernel Error Messages	76
Kernel Notice Messages	76
Kernel Warning Messages	77
Kernel Panic Messages	85
Chapter 3. Disk Array Overview	87
Introduction	87
Disk Array Overview	87
Redundant Arrays of Independent Disks (RAID)	88
RAID-0	89
RAID-1	89
RAID-2	89
RAID-3	89
RAID-4	91
RAID-5	92
Multipathed Disk Arrays	93
Active/Passive Type Disk Arrays	93



Active/Active Type Disk Arrays	94
--------------------------------------	----





Preface

The *VERITAS Volume Manager™ Reference Guide* provides information about Volume Manager commands, error messages, and other topics. For detailed information about the Volume Manager and how to use the Volume Manager, refer to the *VERITAS Volume Manager Administrator's Guide*.

Audience

This guide is intended for system administrators responsible for installing, configuring, and maintaining systems under the control of the VERITAS Volume Manager.

This guide assumes that the user has a:

- ◆ working knowledge of the UNIX operating system
- ◆ basic understanding of system administration
- ◆ basic understanding of volume management

Organization

This guide is organized as follows:

- ◆ Chapter 1 - Volume Manager Commands
- ◆ Chapter 2 - Volume Manager Error Messages
- ◆ Chapter 3 - Disk Array Overview
- ◆ Index

Related Documents

The following documents provide information related to the Volume Manager:

- ◆ *VERITAS Volume Manager Installation Guide*
- ◆ *VERITAS Volume Manager Release Note*



- ◆ *VERITAS Volume Manager Hardware Application Note*
- ◆ *VERITAS Volume Manager Administrator's Guide*
- ◆ *VERITAS Volume Manager Storage Administrator Administrator's Guide*
- ◆ *VERITAS Volume Manager manual pages*

Conventions

The following table describes the typographic conventions used in this guide.

Typeface	Usage	Examples
<code>courier</code>	Computer output; user input; names of commands, files, and directories	\$You have mail. The <code>cat</code> command displays files. \$ls -a
<i>italics</i>	New terms; document titles; words to be emphasized; variables to be substituted with a real name or value	\$cat <i>filename</i> Refer to the <i>User's Guide</i> for details.
bold	Glossary terms	

Introduction

This chapter provides brief descriptions of commonly used Volume Manager commands. For more detailed information about the commands, refer to the appropriate manual pages.

This chapter introduces the following Volume Manager commands:

- ◆ Using vxassist to Create and Change Volumes
- ◆ Creating Volume Manager Objects With vxmake
- ◆ Performing Volume Tasks With vxvol
- ◆ Performing Plex Tasks With vxplex
- ◆ Performing Subdisk Tasks With vxsd
- ◆ Using vxdiskadm to Perform Disk Tasks
- ◆ Adding Disks With vxdiskadd
- ◆ Performing Disk Tasks With vxdisk
- ◆ Performing Disk Group Tasks With vxdg
- ◆ Moving Relocated Subdisks With vxunrelocate
- ◆ Changing Volume Manager Objects With vxedit
- ◆ Correcting Problems With vxmend
- ◆ Printing Configuration Information With vxprint
- ◆ Printing Volume Statistics With vxstat
- ◆ Tracing Volume Tasks With vxtrace
- ◆ Using vxdctl to Manage the vxconfigd Daemon



Using vxassist to Create and Change Volumes

You can use the `vxassist` command to create and change volumes. `vxassist` is an automated one-step interface to Volume Manager tasks. Unlike some Volume Manager commands, `vxassist` does not require a thorough understanding of Volume Manager concepts.

`vxassist` can do tasks that would otherwise require the use of several other Volume Manager commands. `vxassist` automatically performs all underlying and related tasks that would otherwise be done by the user (in the form of other commands).

`vxassist` does not conflict with existing Volume Manager commands or preclude their use. Objects created by `vxassist` are compatible and inter-operable with objects created by other Volume Manager commands and interfaces.

`vxassist` performs these tasks:

- ◆ finds space for and creates volumes
- ◆ finds space for and creates mirrors for existing volumes
- ◆ finds space for and extends existing volumes
- ◆ shrinks existing volumes and frees unused space
- ◆ provides facilities for the online backup of existing volumes
- ◆ provides an estimate of the maximum size for a new or existing volume
- ◆ changes a volume's layout

For detailed information about how to use `vxassist`, refer to the `vxassist(1M)` manual page.

Advantages of the vxassist Command

Some of the advantages of using `vxassist` include:

- ◆ The use of `vxassist` involves only one step (command) on the part of the user.
- ◆ You are required to specify only minimal information to `vxassist`, yet you can optionally specify additional parameters to modify or control its actions.
- ◆ `vxassist` tasks result in a set of configuration changes that either succeed or fail as a group, rather than individually. Most `vxassist` tasks work so that system crashes or other interruptions do not leave intermediate states to be cleaned up. If `vxassist` finds an error or an exceptional condition, it exits without leaving partially-changed configurations. The system is left in the same state as it was prior to the attempted `vxassist` task.
- ◆ `vxassist` is required for `vxrelayout`

How vxassist Works

The `vxassist` command allows you to create and modify volumes. You specify the basic volume creation or modification requirements and `vxassist` performs the necessary tasks.

`vxassist` obtains most of the information it needs from sources other than your input. `vxassist` obtains information about the existing objects and their layouts from the objects themselves.

For tasks requiring new disk space, `vxassist` seeks out available disk space and allocates it in the configuration that conforms to the layout specifications and that offers the best use of free space.

The `vxassist` command typically takes this form:

```
vxassist keyword volume_name [attributes...]
```

where *keyword* selects the task to perform. The first argument after a `vxassist` keyword is a volume name, which is followed by a set of attributes. For details on available `vxassist` keywords and attributes, refer to the `vxassist(1M)` manual page.

`vxassist` creates and manipulates volumes based on a set of established defaults, but also allows you to supply preferences for each task.

vxassist Defaults

`vxassist` uses a set of tunable parameters that can be specified in the defaults files or at the command line. The tunable parameters default to reasonable values if they are not listed on the command line. Tunables listed on the command line override those specified elsewhere. Here are the tunable parameters:

- ◆ Internal defaults—The built-in defaults are used when the value for a particular tunable is not specified elsewhere (on the command line or in a defaults file).
- ◆ System-wide defaults file—The system-wide defaults file contains default values that you can alter. These values are used for tunables that are not specified on the command line or in an alternate defaults file.
- ◆ Alternate defaults file—A non-standard defaults file, specified with the command `vxassist -d alt_defaults_file`.
- ◆ Command line—The tunable values specified on the command line override any values specified internally or in defaults files.



Defaults File

The default behavior of vxassist is controlled by the tunables specified in the `/etc/default/vxassist` file. The format of the defaults file is a list of *attribute=value* pairs separated by new lines. These *attribute=value* pairs are the same as those specified as options on the vxassist command line (refer to the vxassist(1M) manual page for details).

Here is a sample vxassist defaults file:

```
# by default:
# create unmirrored, unstriped volumes
# allow allocations to span drives
# with RAID-5 create a log, with mirroring don't create a log
# align allocations on cylinder boundaries
  layout=nomirror,nostripe,span,nocontig,raid5log,noregionlog,
  diskalign

# use the fsgen usage type, except when creating RAID-5 volumes
  usetype=fsgen

# allow only root access to a volume
  mode=u=rw,g=,o=
  user=root
  group=root

# when mirroring, create two mirrors
  nmirror=2

# for regular striping, by default create between 2 and 8 stripe
# columns
  max_nstripe=8
  min_nstripe=2

# for RAID-5, by default create between 3 and 8 stripe columns
  max_nraid5stripe=8
  min_nraid5stripe=3

# create 1 log copy for both mirroring and RAID-5 volumes, by
# default
  nregionlog=1
  nraid5log=1

# by default, limit mirroring log lengths to 32Kbytes
  max_regionloglen=32k

# use 64K as the default stripe unit size for regular volumes
```

```
stripe_stwid=64k  
  
# use 16K as the default stripe unit size for RAID-5 volumes  
raid5_stwid=16k
```



Creating Volume Manager Objects With vxmake

You can use the `vxmake` command to add a new volume, plex, or subdisk to the set of objects managed by Volume Manager. `vxmake` adds a new record for that object to the Volume Manager configuration database. You can create records from parameters specified on the command line or by using a description file.

You can specify operands on the command line as follows:

```
vxmake -Usage_type vol volume_name len=length plex=plex_name,...
```

where:

- ◆ the first operand (keyword) determines the kind of object to be created
- ◆ the second operand is the name given to that object
- ◆ additional operands specify attributes for the object

If no operands are specified on the command line, then a description file is used to specify the records to create.

A *description file* is a file that contains plain text that describes the objects to be created with `vxmake`. A description file can contain several commands, and can be edited to perform a list of tasks. The description file is read from standard input, unless the `-d description_file` option specifies a filename. The following is a sample description file:

```
#rectyp #name #options
sd disk3-01 disk=disk3 offset=0 len=10000
sd disk3-02 disk=disk3 offset=25000 len=10480
sd disk4-01 disk=disk4 offset=0 len=8000
sd disk4-02 disk=disk4 offset=15000 len=8000
sd disk4-03 disk=disk4 offset=30000 len=4480
plex db-01 layout=STRIPE ncolumn=2 stwidth=16k
sd=disk3-01:0/0,disk3-02:0/10000,disk4-01:1/0,\
disk4-02:1/8000, disk4-03:1/16000
sd ramd1-01 disk=ramd1 len=640
comment="Hot spot for dbvol
plex db-02 sd=ramd1-01:40320
vol db usetype=gen plex=db-01,db-02
readpol=prefer pefname=db-02
comment="Uses mem1 for hot spot in last 5m
```

This description file specifies a volume with two plexes. The first plex has five subdisks on physical disks. The second plex is preferred and has one subdisk on a volatile memory disk.

For detailed information about how to use `vxmake`, refer to the `vxmake(1M)` manual page.



Performing Volume Tasks With vxvol

The `vxvol` command performs Volume Manager tasks on volumes. You can use `vxvol` to:

- ◆ initialize a volume
- ◆ start a volume
- ◆ stop a volume
- ◆ establish the read policy for a volume

Starting a volume changes its kernel state from `DISABLED` or `DETACHED` to `ENABLED`. Stopping a volume changes its state from `ENABLED` or `DETACHED` to `DISABLED` (however, it is seldom useful to stop a volume). For more information about volume states, see the “Recovery” chapter in the *VERITAS Volume Manager Administrator’s Guide*.

You can use `vxvol` to specify one of these read policies:

- ◆ `round` reads each plex in turn in “round-robin” fashion for each nonsequential I/O detected. Sequential access causes only one plex to be accessed. This takes advantage of the drive or controller read-ahead caching policies.
- ◆ `prefer` reads first from a plex that has been named as the preferred plex.
- ◆ `select` chooses a default policy based on plex associations to the volume. If the volume has an enabled striped plex, `select` defaults to preferring that plex; otherwise, it defaults to round-robin.

For detailed information about how to use `vxvol`, refer to the `vxvol(1M)` manual page.

Performing Plex Tasks With vxplex

The `vxplex` command performs Volume Manager tasks on a plex or on volume-and-plex combinations. The first operand is a keyword that specifies the task to perform. The remaining operands specify the objects to which the task is to be applied.

You can use the `vxplex` command to:

- ◆ Attach or detach a plex and a volume. A detached plex does not share in I/O activity to the volume, but remains associated with the volume. A detached plex is reattached when a volume is next started.
- ◆ Dissociate a plex from the associated volume. When a plex is dissociated, its relationship to the volume is broken. The plex is then available for other uses and can be associated with a different volume. This is useful as part of a backup procedure.



- ◆ Copy the contents of the specified volume to the named plex(es). This task makes a copy of a volume (for backup purposes) without mirroring the volume in advance.
- ◆ Move the contents of one plex to a new plex. This is useful for moving a plex on one disk to a different location.

For detailed information about how to use `vxplex`, refer to the `vxplex(1M)` manual page.

Performing Subdisk Tasks With vxsd

The `vxsd` command performs Volume Manager tasks on subdisks. `vxsd` maintains subdisk-mirror associations. You can use `vxsd` to:

- ◆ associate or dissociate a subdisk from its associated mirror
- ◆ move the contents of a subdisk to another subdisk
- ◆ split one subdisk into two subdisks that occupy the same space as the original
- ◆ join two contiguous subdisks into one.

For detailed information about how to use `vxsd`, refer to the `vxsd(1M)` manual page.

Using vxdiskadm to Perform Disk Tasks

`vxdiskadm` is the Volume Manager Support Operations menu interface. This command provides a menu of disk operations. Each entry in the main menu leads you through a particular task by providing you with information and prompts. Default answers are provided for many questions so you can easily select typical answers. Refer to the `vxdiskadm(1M)` manual page for information on how to use `vxdiskadm`.

Adding Disks With vxdiskadd

The `vxdiskadd` command adds standard disks to the Volume Manager. `vxdiskadd` leads you through the process of initializing a new disk by displaying information and prompts. See the `vxdiskadd(1M)` manual page for information on how to use `vxdiskadd`.

Performing Disk Tasks With `vxdisk`

The `vxdisk` command performs Volume Manager tasks on disks under VxVM[®] control. `vxdisk` defines special disk devices, initializes information stored on disks (that the Volume Manager uses to identify and manage disks), and performs additional special operations. See the `vxdisk(1M)` manual page for information on how to use `vxdisk`.

Performing Disk Group Tasks With `vx dg`

The `vx dg` command performs Volume Manager tasks on disk groups. `vx dg` is used for operating on disk groups. `vx dg` creates new disk groups, and administers existing disk groups. See the `vx dg(1M)` manual page for information on how to use `vx dg`.

Moving Relocated Subdisks With `vxunreloc`

The `vxunreloc` command moves subdisks that have been relocated by the hot-relocation feature back to their original disks. This allows you to restore the system back to the configuration that existed before the disk failure. See the `vxunreloc(1M)` manual page for information on how to use `vxunreloc`.

Changing Volume Manager Objects With `vxedit`

The `vxedit` command sets and changes attributes for Volume Manager configuration records. `vxedit` has two functions:

- ◆ `vxedit` can remove or rename Volume Manager objects.
- ◆ `vxedit` allows you to modify certain records in the volume management databases. Only fields that are not volume usage-type-dependent can be modified.

Volume Manager objects that are associated with other objects are not removable by `vxedit`. This means that `vxedit` cannot remove a subdisk that is associated with a plex or a plex that is associated with a volume.

Using the recursive suboption (`-r`) to the removal option of `vxedit` removes all objects from the specified object downward. In this way, a plex and its associated subdisks, or a volume and its associated plexes and their associated subdisks, can be removed by a single `vxedit` command.

For detailed information about how to use `vxedit`, refer to the `vxedit(1M)` manual page.



Correcting Problems With vxmend

The `vxmend` command performs Volume Manager usage-type-specific tasks on volumes, plexes, and subdisks. These tasks fix simple problems in configuration records (such as clearing utility fields, changing volume or plex states, and offlining or onlining volumes or plexes).

`vxmend` is used primarily to escape from a state that was accidentally reached. The offline and online functions can be performed with disk-related commands.

For detailed information about how to use `vxmend`, refer to the `vxmend(1M)` manual page.

Printing Configuration Information With vxprint

The `vxprint` command displays information from records in a Volume Manager configuration database. You can use this command to display partial or complete information about any or all Volume Manager objects. The format can be hierarchical to clarify relationships between Volume Manager objects. UNIX system utilities such as `awk`, `sed`, or `grep` can also use `vxprint` output.

For detailed information about how to use `vxprint`, refer to the `vxprint(1M)` manual page.

Printing Volume Statistics With vxstat

The `vxstat` command prints statistics about Volume Manager objects and block devices under Volume Manager control. `vxstat` reads the summary statistics for Volume Manager objects and formats them to the standard output. These statistics represent Volume Manager activity from the time the system initially booted or from the last time statistics were cleared. If no Volume Manager object name is specified, statistics from all volumes in the configuration database are reported.

For detailed information about how to use `vxstat`, refer to the `vxstat(1M)` manual page.

Tracing Volume Tasks With vxtrace

The `vxtrace` command prints kernel I/O error or I/O error trace event records on the standard output or writes them to a file in binary format. Binary trace records written to a file can also be read back and formatted by `vxtrace`.

If no operands are given, then either all error trace data or all I/O trace data on all virtual disk devices are reported. With error trace data, you can select all accumulated error trace data, wait for new error trace data, or both (the default). Selection can be limited to a specific disk group, to specific Volume Manager kernel I/O object types, or to particular named objects or devices.

For detailed information about how to use `vxtrace`, refer to the `vxtrace(1M)` manual page.

Using vxctl to Manage the vxconfigd Daemon

The volume configuration daemon (`vxconfigd`) is the interface between the Volume Manager commands and the kernel device drivers. The `vxctl` command is the interface to `vxconfigd`.

`vxctl` is used for:

- ◆ performing tasks related to the state of the `vxconfigd` daemon
- ◆ managing boot information and Volume Manager root configuration initialization
- ◆ manipulating the contents of the `volboot` file that contains a list of disks containing root configuration databases (this is not normally necessary, as Volume Manager automatically locates all disks on the system)
- ◆ reconfiguring the DMP database (if used on your system) to reflect new disk devices attached to the system, and removal of any disk devices from the system
- ◆ creating DMP (if used on your system) device nodes in the device directories `/dev/vx/dmp` and `/dev/vx/rdmp`
- ◆ reflecting the change in path type into the DMP database for active/passive disk arrays. You can change the path type from primary to secondary and vice-versa through the utilities provided by disk array vendors

The `config` device is a special device file created by the Volume Manager that interacts with `vxctl` to make system configuration changes. Some `vxctl` tasks involve modifications to the `volboot` file that indicates the location of some root configuration copies.

For detailed information about how to use `vxctl`, refer to the `vxctl(1M)` manual page.





Introduction

This chapter provides information on error messages associated with the Volume Manager configuration daemon (`vxconfigd`), the kernel, and other utilities. It covers most informational, failure, and error messages displayed (on the console) by `vxconfigd` and the kernel driver. These include some errors that are infrequently encountered and difficult to troubleshoot.

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

Clarifications are included to elaborate on the situation or problem that generated a particular message. Wherever possible, a recovery procedure (Action) is provided to help you to locate and correct the problem.

The following sections are included in this chapter:

- ◆ Logging Error Messages
- ◆ Volume Manager Configuration Daemon Error Messages
 - ◆ `vxconfigd` Usage Messages
 - ◆ `vxconfigd` Error Messages
 - ◆ `vxconfigd` Fatal Error Messages
 - ◆ `vxconfigd` Notice Messages
 - ◆ `vxconfigd` Warning Messages
- ◆ DMP Error Messages
- ◆ Cluster Error Messages
- ◆ Kernel Error Messages
 - ◆ Kernel Notice Messages
 - ◆ Kernel Warning Messages
 - ◆ Kernel Panic Messages



Logging Error Messages

The Volume Manager provides the option of logging console output 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). `vxconfigd` controls whether such logging is turned on or off. If enabled, the default log file is `/var/adm/vxconfigd.log`.

`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 (into a private log file), along with distributed logging through `syslogd`.

Both `syslog()` and log file logging are disabled by default.

To enable logging of console output to a file, you can either invoke `vxconfigd` as follows or edit Volume Manager startup scripts (described later):

```
vxconfigd -x log
```

`-x [1-9]` is required to get any debug output. So if `-x 9 -x log` is mentioned, then messages will get logged to `/var/adm/configd.log`.

To enable `syslog()` logging of console output, you can either invoke `vxconfigd` as follows or edit Volume Manager startup scripts (described later):

```
vxconfigd -x syslog
```

When `-x syslog` is used, **Debug** messages are not logged. **Only Error, Fatal Error, Warning, and Notice** messages are logged.

If you specify, `-x 9 -x log -x syslog`, then all the **Debug** messages only go to `/var/adm/vxconfigd.log`, but error messages go to both `/var/adm/vxconfigd.log` and to `/var/adm/syslog/syslog.log`.

To enable log file and/or `syslog()` logging, you can also edit the following portion of the `/etc/init.d/vxvm-sysboot` startup script:

```
# 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
                                # /var/vxvm/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
```

Uncomment the line(s) corresponding to the feature(s) that you want enabled at startup. For example, to set up `vxconfigd` to automatically use file logging, uncomment the `opts="$opts -x log"` string.

For more information on logging options available through `vxconfigd`, refer to the `vxconfigd(1M)` manual page.



Volume Manager Configuration Daemon Error Messages

The Volume Manager is fault-tolerant and resolves most problems without system administrator intervention. If the Volume Manager configuration daemon (`vxconfigd`) recognizes what actions are necessary, it will queue up the transactions that are required. Volume Manager provides atomic changes of system configurations; either a transaction completes fully or the system appears as though the transaction was never attempted. When `vxconfigd` is unable to recognize and fix system problems, the system administrator needs to handle the task of problem solving.

The following sections cover the error messages associated with the Volume Manager configuration daemon.

vxconfigd Usage Messages

The following are usage messages associated with `vxconfigd`.

Usage: **vxconfigd - long**

```
Usage: vxconfigd [-dkf] [-r reset] [-m mode] [-x level]
```

Recognized options:

```
-d      set initial mode to disabled for transactions
-k      kill the existing configuration daemon process
-f      operate in foreground; default is to operate in background
-r reset reset kernel state; requires 'reset' option argument
-m mode set vold's operating mode
        modes: disable, enable, bootload, bootstart
-x debug set debugging level to <debug>, 0 turns off debugging
-R file  set filename for client request rendezvous
-D file  set filename for client diag request rendezvous
```

❖ Clarification

This is the full usage message for `vxconfigd`, which results from entering the command `vxconfigd help`.

Usage: **vxconfigd - short**

```
Usage: vxconfigd [-dkf] [-r reset] [-m mode] [-x level]
For detailed help use: vxconfigd help
```

❖ Clarification

This is the standard `vxconfigd` usage error message. Appearance of this message implies that some option was supplied incorrectly.



❖ Action

If you need help in using `vxconfigd`, try using the command `vxconfigd help`.

For more detailed information, see the `vxconfigd(1M)` manual page.

`-r` must be followed by 'reset'

`-r` must be followed by 'reset'

❖ Clarification

This is a usage error. The `-r` option requires an option argument consisting of the string `reset`.

❖ Action

Either don't use the `-r` option, or supply the `reset` option argument.

`-x` argument: invalid debug string

`-x` argument: invalid debug string

❖ Clarification

An unrecognized string was specified as an argument to the `-x` option.

❖ Action

See `vxconfigd(1M)` for a list of valid arguments to `-x`.

`-x devprefix=device_prefix: prefix too long`

`-x devprefix=device_prefix: prefix too long`

❖ Clarification

The `-x devprefix=device_prefix` option was used to define a prefix path for the `/dev/dsk` and `/dev/rdsk` directories, and that prefix was too long.

❖ Action

Use a shorter prefix.

vxconfigd Error Messages

The following are general error messages associated with `vxconfigd`.



signal_name [core **dumped**]

```
vxvm:vxconfigd: ERROR: signal_name [ - core dumped ]
```

❖ **Clarification**

The vxconfigd daemon encountered an unexpected signal while starting up. The specific signal is indicated by *signal_name*. If the signal caused the vxconfigd process to dump core, then that will be indicated. This could be caused by a bug in vxconfigd, particularly if *signal_name* is "Segmentation fault." Alternately, this could have been caused by a user sending vxconfigd a signal with the kill utility.

❖ **Action**

Contact Customer Support.

Unrecognized operating mode

```
vxvm:vxconfigd: ERROR: mode_name: Unrecognized operating mode
```

❖ **Clarification**

An invalid string was specified as an argument to the -m option. Valid strings are: enable, disable, and boot.

❖ **Action**

Supply a correct option argument.

vxconfigd cannot boot-start RAID-5 volumes

```
vxvm:vxconfigd: ERROR: volume_name: vxconfigd cannot boot-start RAID-5 volumes
```

❖ **Clarification**

A volume that vxconfigd should start immediately upon booting the system (i.e., 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.

❖ **Action**

It is likely that the only recovery for this is to boot the Volume Manager 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.

Cannot get all disk groups from the kernel

```
vxvm:vxconfigd: ERROR: Cannot get all disk groups from the kernel: reason
```

❖ **Clarification**

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ **Action**

Contact Customer Support for more information.

Cannot get all disks from the kernel

```
vxvm:vxconfigd: ERROR: Cannot get all disks from the kernel: reason
```

❖ **Clarification**

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ **Action**

Contact Customer Support for more information.

Cannot get kernel transaction state

```
vxvm:vxconfigd: ERROR: Cannot get kernel transaction state: reason
```

❖ **Clarification**

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ **Action**

Contact Customer Support for more information.

Cannot get private storage from kernel

```
vxvm:vxconfigd: ERROR: Cannot get private storage from kernel: reason
```

❖ **Clarification**

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.



❖ Action

Contact Customer Support for more information.

Cannot get private storage size from kernel

```
vxvm:vxconfigd: ERROR: Cannot get private storage size from kernel: reason
```

❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Cannot get record from the kernel

```
vxvm:vxconfigd: ERROR: Cannot get record record_name from the kernel: reason
```

❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Cannot kill existing daemon, pid=process-ID

```
vxvm:vxconfigd: ERROR: Cannot kill existing daemon, pid=process-ID
```

❖ Clarification

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, then the above error message will be displayed.

❖ 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 appears again, contact Customer Support.

Cannot make directory

```
vxvm:vxconfigd: ERROR: Cannot make directory directory_path: reason
```

❖ Clarification

`vxconfigd` failed to create a directory that it expects to be able to create. Directories that `vxconfigd` might try to create are: `/dev/vx/dsk`, `/dev/vx/rdsk`, and `/var/vxvm/tempdb`. Also, for each disk group, `/dev/vx/dsk/diskgroup` and `/dev/vx/rdsk/diskgroup` directories are created. The system error related to the failure is given in *reason*. A system error of “No such file or directory” indicates that one of the prefix directories (for example, `/var/vxvm`) does not exist.

This type of error normally implies that the Volume Manager packages were installed incorrectly. Such an error can also occur if alternate file or directory locations are specified on the command line, using the `-x` option. The `_VXVM_ROOT_DIR` environment variable may also relocate to a directory that lacks a `var/vxvm` subdirectory.

❖ Action

Try to create the directory manually and then issue the command `vxdctl enable`. If the error is due to incorrect installation of the Volume Manager packages, try to add the Volume Manager packages again.

Cannot open `/etc/vfstab`

```
vxvm:vxconfigd: ERROR: Cannot open /etc/vfstab: reason
```

❖ Clarification

`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. If the `/etc/vfstab` file cannot be opened, `vxconfigd` prints the above error message and exits.

❖ Action

This error implies that your root file system is currently unusable. You may be able to repair your root file system by mounting the root file system after booting from a network or CD-ROM root file system. If the root file system is defined on a volume, then see the procedures defined for recovering from a failed root file system in the “Recovery” chapter of the Administrator’s Guide.



Cannot recover operation in progress

```
vxvm:vxconfigd: ERROR: Cannot recover operation in progress  
Failed to get group group from the kernel: error
```

❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Cannot reset VxVM kernel

```
vxvm:vxconfigd: ERROR: Cannot reset VxVM kernel: reason
```

❖ Clarification

The `-r` reset option was specified to `vxconfigd`, but the VxVM kernel drivers could not be reset. The most common reason for this is “A virtual disk device is open.” That error implies that a VxVM tracing device or volume device is open.

❖ Action

If, for some reason, you really want to reset the kernel devices, you will need to track down and kill all processes that have a volume or VxVM tracing device open. Also, if any volumes are mounted as file systems, unmount those file systems.

An error reason other than “A virtual disk device is open” should not normally occur unless there is a bug in the operating system or in the Volume Manager.

Cannot start volume, no valid complete plexes

```
vxvm:vxconfigd: ERROR: Cannot start volume volume, no valid complete plexes
```

❖ Clarification

This error indicates that the volume for either the root or `/usr` file system cannot be started because the volume contains no 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).

❖ 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 restore the root or /usr file system or to reinstall the system. Restoring the root or /usr file system requires that you have a valid backup. See the “Recovery” chapter of the Administrator’s Guide for information on how to fix problems with root or /usr file system volumes.

Cannot start volume, no valid plexes

```
vxvm:vxconfigd: ERROR: Cannot start volume volume, no valid plexes
```

❖ Clarification

This error indicates that the volume for either the root or /usr file system cannot be started because the volume contains no 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, dissociating, or offlining plexes).

❖ 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 restore the root or /usr file system or to reinstall the system. Restoring the root or /usr file system requires that you have a valid backup. See the “Recovery” chapter of the Administrator’s Guide for information on how to fix problems with root or /usr file system volumes.

Cannot start volume, volume state is invalid

```
vxvm:vxconfigd: ERROR: Cannot start volume volume, volume state is invalid
```

❖ Clarification

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 the Volume Manager to create these volumes.

❖ Action

The only recourse is to bring up the Volume Manager on a CD-ROM or NFS-mounted root file system and to fix the state of the volume. See the “Recovery” chapter of the Administrator’s Guide for further information.

Cannot store private storage into the kernel

```
vxvm:vxconfigd: ERROR: Cannot store private storage into the kernel: error
```



❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Differing version of vxconfigd installed

```
vxvm:vxconfigd: ERROR: Differing version of vxconfigd installed
```

❖ Clarification

A vxconfigd daemon was started after the stopping of an earlier vxconfigd with a non-matching version number. This can happen, for example, if you upgrade from an earlier release of Volume Manager to VxVM 3.0 and run vxconfigd without a reboot.

❖ Action

To fix, reboot the system.

Disk, group, device: not updated with new host ID

```
vxvm:vxconfigd: ERROR: Disk disk, group group, device device: not updated with  
new host ID  
Error: reason
```

❖ Clarification

This can result from using vxctl hostid to change the 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. Most likely, this indicates that the given disk has become inaccessible or has failed in some other way.

❖ Action

Try running the following 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 been taken out of use already. If the disk is still operational (which should not be the case), vxdisk will print:



device: Okay

If the disk is listed as Okay, try `vxctl hostid` again. If it still results in an error, contact Customer Support.

Disk group, Disk: Cannot auto-import group

```
vxvm:vxconfigd: ERROR: Disk group group, Disk disk: Cannot auto-import group:  
reason
```

❖ Clarification

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: 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 the Volume Manager 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 error types of:

```
Format error in configuration copy  
Invalid magic number  
Invalid block number  
Duplicate record in configuration  
Configuration records are inconsistent
```

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 the Volume Manager).

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.

❖ Action

If the error is clearly caused by excessive disk failures, then you may have to recreate the disk group and restore contents of any volumes from a backup. 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 Customer Support.



Disk group, Disk: Group name collides with record in rootdg

vxvm:vxconfigd: ERROR: Disk group *group*, Disk *device*: Group name collides with record in rootdg

❖ Clarification

The name of a disk group that is being imported conflicts with the name of a record in the `rootdg` disk group. Volume Manager does not allow this kind of conflict because of the way the `/dev/vx/dsk` directory is organized: devices corresponding to records in the root disk group share this directory with subdirectories for each disk group.

❖ Action

Either remove or rename the conflicting record in the root disk group, or rename the disk group on import. See the `vxdg(1M)` manual page for information on how to use the `import` operation to rename a disk group.

Disk group, Disk: Skip disk group with duplicate name

vxvm:vxconfigd: ERROR: Disk group *group*, Disk *device*: Skip disk group with duplicate name

❖ Clarification

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.

The Volume Manager 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. However, this error can occur in the following two 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, then the new disk group will be imported first, and the auto-import of the older disk group will fail with `group with duplicate name` (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.

❖ Action



If you want to import both disk groups, then rename the second disk group on import. See the `vxchg(1M)` manual page for information on how to use the `import` operation to rename a disk group.

Disk group: Cannot recover temp database

```
vxvm:vxconfigd: ERROR: Disk group group: Cannot recover temp
database: reason
```

Consider use of "vxconfigd -x cleartempdir" [see `vxconfig(1M)`].

❖ Clarification

This can happen if you kill and restart `vxconfigd` or you if you disable and enable it with `vxctl disable` and `vxctl enable`. This error indicates a failure related to reading the file `/var/vxvm/tempdb/groupname`. 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.

❖ Action

If you can reboot, do so. If you do not want to reboot, then do the following:

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. Run the command:

```
vxconfigd -x cleartempdir 2> /dev/console
```

This will recreate the temporary database files for all imported disk groups.

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.

Disk group: Disabled by errors

```
vxvm:vxconfigd: ERROR: Disk group group: Disabled by errors
```



❖ Clarification

This message indicates that some error condition has made it impossible for Volume Manager 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 `rootdg` disk group, then the following additional error should be displayed:

```
vxvm:vxconfigd: ERROR: 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 `rootdg`.

❖ 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. Failure of the `rootdg` disk group may require reinstallation of the system if your system uses a `root` or `/usr` file system defined on a volume.

```
Disk group: Errors in some configuration copies: Disk, copy
```

```
vxvm:vxconfigd: ERROR: Disk group group: Errors in some configuration copies:  
Disk disk, copy number: [Block number]:  
reason ...
```

❖ Clarification

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.

❖ Action

If some of the copies failed due to transient errors (such as cable failures), then a reboot or reimport may succeed in importing the disk group. Otherwise, the disk group may have to be recreated from scratch.

```
Disk group: Reimport of disk group failed
```

```
vxvm:vxconfigd: ERROR: Disk group group: Reimport of disk group failed: reason
```

❖ Clarification



After `vxconfigd` was stopped and restarted (or disabled and then enabled), the Volume Manager 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.

❖ 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.

Disk group: update failed

```
vxvm:vxconfigd: ERROR: Disk group group: update failed: reason
```

❖ Clarification

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: Disk group group: Disabled by errors
```

❖ 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.

Failed to store commit status list into kernel

```
vxvm:vxconfigd: ERROR: Failed to store commit status list into kernel: reason
```

❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

GET_VOLINFO ioctl failed

```
vxvm:vxconfigd: ERROR: GET_VOLINFO ioctl failed: reason
```



❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Get of current rootdg failed

```
vxvm:vxconfigd: ERROR: Get of current rootdg failed: reason
```

❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Memory allocation failure

```
vxvm:vxconfigd: ERROR: Memory allocation failure
```

❖ Clarification

This implies that there is insufficient memory to start up the Volume Manager and to get the volumes for the root and /usr file systems running.

❖ Action

This error should not normally occur, unless your system has very small amounts of memory. Adding just 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.

Mount point: volume not in rootdg disk group

```
vxvm:vxconfigd: ERROR: Mount point path: volume not in rootdg disk group
```

❖ Clarification

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 rootdg. This error should not occur if the standard Volume Manager procedures are used for encapsulating the disk containing the /usr file system.



❖ Action

You will need to boot the Volume Manager from a network or CD-ROM mounted root file system. Then, start up the Volume Manager using `fixmountroot` on a valid mirror disk of the root file system. After starting Volume Manager, 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, for example:

```
#NOTE: volume usr (/usr) encapsulated partition
c0t3d0s5 (or c0b0t3d0s5 if your system uses a bus).
```

No convergence between root disk group and disk list

```
vxvm:vxconfigd: ERROR: No convergence between root disk group and disk list
Disks in one version of rootdg:
  device type=device_type info=devinfo ...
Disks in alternate version of rootdg:
  device type=device_type info=devinfo ...
```

❖ Clarification

This message can appear when `vxconfigd` is not running in autoconfigure mode (see the `vxconfigd(1M)` manual page) and when, after several retries, it can not resolve the set of disks belonging to the root disk group. The algorithm for non-autoconfigure disks is to scan disks listed in the `/etc/vx/volboot` file and then examine the disks to find a database copy for the `rootdg` disk group. The database copy is then read to find the list of disk access records for disks contained in the group. These disks are then examined to ensure that they contain the same database copy. As such, this algorithm expects to gain convergence on the set of disks and the database copies contained on them. If a loop is entered and convergence cannot be reached, then this message will appear and the root disk group importation will fail.

❖ Action

Reorganizing the physical locations of the devices attached to the system may break the deadlock. Failing this, contact Customer Support.

Open of directory failed

```
vxvm:vxconfigd: ERROR: Open of directory directory failed: reason
```

❖ Clarification

An open failed for the `/dev/vx/dsk` or `/dev/vx/rdsk` directory (or a subdirectory of either of those directories). The only likely cause of such a failure should be that the directory was removed by the administrator or by an errant program. For this case, the *reason* should be "No such file or directory." An alternate possible cause is an I/O failure.



❖ Action

If the error was “No such file or directory,” then create the directory (using `mkdir`). Then run the command `vxctl enable`.

If the error was an I/O error, then there may be other serious damage to the root file system. You may need to reformat your root disk and restore the root file system from backup. Contact your system vendor or consult your system documentation.

Read of directory failed

```
vxvm:vxconfigd: ERROR: Read of directory directory failed: reason
```

❖ Clarification

There was a failure in reading the `/dev/vx/dsk` or `/dev/vx/rdisk` directory (or a subdirectory of either of those directories). The only likely cause of this error is an I/O failure on the root file system.

❖ Action

If the error was an I/O error, then there may be other serious damage to the root file system. You may need to reformat your root disk and restore the root file system from backup. Contact your system vendor or consult your system documentation.

System boot disk does not have a valid root plex

```
vxvm:vxconfigd: ERROR: System boot disk does not have a valid root plex  
Please boot from one of the following disks:  
Disk: diskname Device: device ...
```

❖ Clarification

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.

❖ Action

Try to boot from one of the disks named in the error message.

Under some operating systems, you may be able to boot using a device alias for one of the named disks. For example, use this command:

```
boot vx-diskname
```

System startup failed

```
vxvm:vxconfigd: ERROR: System startup failed
```

❖ Clarification

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.

❖ Action

Look up other error messages appearing on the console and take the actions suggested in the descriptions of those messages.

There is no volume configured for the root device

```
vxvm:vxconfigd: ERROR: There is no volume configured for the root device
```

❖ Clarification

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 `rootdg` disk group.

There are two 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 should happen only as a result of direct manipulation by the administrator.
- ◆ Case 2: The system somehow has a duplicate `rootdg` 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 `rootdg`, 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 renamed to `rootdg` with locks given to this host.

❖ Action

In 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
```

In case 2, either boot with all drives in the offending version of `rootdg` turned off, or import and rename [see `vxchg(1M)`] the offending `rootdg` disk group from another host. In the case of turning off drives, run the following command after booting:

```
vxchg flush rootdg
```



This will update time stamps on the imported version of `rootdg`, which should make the correct version appear to be the more recently accessed. If this does not correct the problem, then contact Customer Support.

Unexpected configuration tid for group found in kernel

```
vxvm:vxconfigd: ERROR: Unexpected configuration tid for group group found in kernel
```

❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Unexpected error during volume reconfiguration

```
vxvm:vxconfigd: ERROR: Unexpected error during volume volume reconfiguration: reason
```

❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Unexpected error fetching disk for volume

```
vxvm:vxconfigd: ERROR: Unexpected error fetching disk for disk volume: reason
```

❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.



Unexpected values stored in the kernel

```
vxvm:vxconfigd: ERROR: Unexpected values stored in the kernel
```

❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Version number of kernel does not match vxconfigd

```
vxvm:vxconfigd: ERROR: Version number of kernel does not match vxconfigd
```

❖ Clarification

The release of `vxconfigd` does not match the release of the Volume Manager kernel drivers. This should happen only as a result of upgrading Volume Manager, and then running `vxconfigd` without a reboot.

❖ Action

Reboot the system. If that does not cure the problem, then add the VxVM packages again.

Volume for mount point /usr not found in rootdg disk group

```
vxvm:vxconfigd: ERROR: Volume volume for mount point /usr not found in rootdg disk group
```

❖ Clarification

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 `rootdg` disk group. There are a couple of possible causes of this error:

- ◆ 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 `rootdg` disk group. This should happen only as a result of direct manipulation by the administrator.
- ◆ The system somehow has a duplicate `rootdg` 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 `rootdg`. Since `vxconfigd` chooses the more recently accessed version of `rootdg`, 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 renamed to `rootdg` with locks given to this host.

❖ Action

In 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 using the procedures defined in the “Recovery” chapter of the Administrator’s Guide. If the root file system is not defined on a volume, then just mount the root file system directly. Edit the `/etc/vfstab` file to correct the entry for the `/usr` file system.

In case 2, either boot with all drives in the offending version of `rootdg` turned off, or import and rename [see `vxdg(1M)`] the offending `rootdg` disk group from another host. In the case of turning off drives, run the following command after booting:

```
vxdg flush rootdg
```

This will update time stamps on the imported version of `rootdg`, which should make the correct version appear to be the more recently accessed. If this does not correct the problem, then contact Customer Support.

```
cannot open /dev/vx/config
```

```
vxvm:vxconfigd: ERROR: cannot open /dev/vx/config: reason
```

❖ Clarification

The `/dev/vx/config` device could not be opened. `vxconfigd` uses this device to communicate with the Volume Manager kernel drivers. The *reason* string indicates the reason for the open failure. The most likely reason is `Device is already open`. This reason indicates that some process (most likely `vxconfigd`) already has `/dev/vx/config` open. Other less likely reasons are “No such file or directory” or “No such device or address.” For either of these two reasons, the two likely causes are:

- ◆ The Volume Manager package installation did not complete correctly.
- ◆ The device node was removed by the administrator or by an errant shell script.

❖ Action

For the reason “Device is already open,” if you really want to run `vxconfigd`, then stop or kill the old one. You can kill whatever process has `vxconfigd` open by running the command:

```
vxdctl -k stop
```

For other failure reasons, consider re-adding the base Volume Manager package. This will reconfigure the device node and re-install the Volume Manager kernel device drivers. See the *VERITAS Volume Manager Installation Guide* for information on how to add the package. If you cannot re-add the package, then contact Customer Support for more information.

enable failed

```
vxvm:vxconfigd: ERROR: enable failed: reason
```

❖ Clarification

Regular startup of `vxconfigd` failed for the stated reason. This error can also result from the command `vxctl enable`. This error may include the following additional text:

additional-reason; aborting

This message indicates that the failure was fatal and that `vxconfigd` is forced to exit. The most likely cause that results in an abort is inability to create IPC channels for communicating with other utilities.

additional-reason; transactions are disabled

This message indicates that `vxconfigd` is continuing to run, but no configuration updates are possible until the error condition is repaired.

Additionally, this may be followed with:

```
vxvm:vxconfigd: ERROR: Disk group group: Errors in some configuration copies:
```

```
Disk device, copy number: Block bno: error ...
```

Reasons for failure vary considerably. Other error messages may be displayed that further indicate the underlying problem. If the `Errors in some configuration copies error` occurs, then that may indicate the problem.

❖ Action

Evaluate other error messages occurring with this one to determine the root cause of the problem. Make changes suggested by the other errors and then retry the command.

/dev/vx/info

```
vxvm:vxconfigd: ERROR: /dev/vx/info: reason
```

❖ Clarification

The `/dev/vx/info` device could not be opened, or did not respond to a Volume Manager kernel request. This error most likely indicates one of the following:

- ◆ The Volume Manager package installation did not complete correctly.



-
- ◆ The device node was removed by the administrator or by an errant shell script.

- ❖ Action

Consider re-adding the base Volume Manager package. This will reconfigure the device node and re-install the Volume Manager kernel device drivers. See the *VERITAS Volume Manager Installation Guide* for information on how to add the package.

vxconfigd Fatal Error Messages

The following are fatal error messages associated with vxconfigd.

Disk group rootdg: Inconsistency -- Not loaded into kernel

```
vxvm:vxconfigd: FATAL ERROR: Disk group rootdg: Inconsistency -- Not loaded into kernel
```

- ❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

- ❖ Action

Contact Customer Support for more information.

Group: Cannot update kernel

```
vxvm:vxconfigd: FATAL ERROR: Group group: Cannot update kernel
```

- ❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

- ❖ Action

Contact Customer Support for more information.

Interprocess communication failure

```
vxvm:vxconfigd: FATAL ERROR: Interprocess communication failure: reason
```

- ❖ Clarification



This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Invalid status stored in kernel

```
vxvm:vxconfigd: FATAL ERROR: Invalid status stored in kernel
```

❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Memory allocation failure during startup

```
vxvm:vxconfigd: FATAL ERROR: Memory allocation failure during startup
```

❖ Clarification

This implies that there is insufficient memory to start up the Volume Manager and to get the volumes for the root and /usr file systems running.

❖ Action

This error should not normally occur, unless your system has very small amounts of memory. Adding just 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.

Rootdg cannot be imported during boot

```
vxvm:vxconfigd: FATAL ERROR: Rootdg cannot be imported during boot
```

❖ Clarification

This is an internal Volume Manager error. This error should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.



Unexpected threads failure

```
vxvm:vxconfigd: FATAL ERROR: Unexpected threads failure: reason
```

❖ Clarification

This is an unexpected operating system error. This error should not occur unless there is a bug in the Volume Manager or in the operating system multi-threading libraries.

❖ Action

Contact Customer Support for more information.

vxconfigd Notice Messages

The following are notice messages associated with `vxconfigd`.

Detached disk

```
vxvm:vxconfigd: NOTICE: Detached disk disk
```

❖ Clarification

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.

❖ Action

If hot-relocation is enabled, the Volume Manager objects affected by the disk failure may be taken care of automatically. Mail will be sent to `root` indicating what actions were taken by the Volume Manager and what further actions the administrator should take.

Detached log for volume

```
vxvm:vxconfigd: NOTICE: Detached log for volume volume
```

❖ Clarification

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 `vxchg -k rmdisk`. A failing disk is indicated by a Detached disk *disk* message.

❖ Action



If the log is mirrored, hot-relocation may automatically relocate the failed log. Remove the failing logs using either `vxplex dis` or `vxsd dis`. Then, use `vxassist addlog` [see the `vxassist(1M)` manual page] to add a new log to the volume.

Detached plex in volume

```
vxvm:vxconfigd: NOTICE: Detached plex plex in volume volume
```

❖ Clarification

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 *disk* message.

❖ Action

If hot-relocation is enabled, the Volume Manager objects affected by the disk failure may be taken care of automatically. Mail will be sent to `root` indicating what actions were taken by the Volume Manager and what further actions the administrator should take.

Detached subdisk in volume

```
vxvm:vxconfigd: NOTICE: Detached subdisk subdisk in volume volume
```

❖ Clarification

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 *disk* message.

❖ Action

If hot-relocation is enabled, the Volume Manager objects affected by the disk failure may be taken care of automatically. Mail will be sent to `root` indicating what actions were taken by the Volume Manager and what further actions the administrator should take.

Detached volume

```
vxvm:vxconfigd: NOTICE: Detached volume volume
```

❖ Clarification

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 *disk* message. Unless the disk error is transient and can be fixed with a reboot, the contents of the volume should be considered lost.



❖ Action

There is no action to be taken. Contact Customer Support for more information.

Offlining config copy on disk

```
vxvm:vxconfigd: NOTICE: Offlining config copy number on disk disk: Reason:  
reason
```

❖ Clarification

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.

❖ Action

You should consider replacing the indicated disk, since this error implies that the disk has deteriorated to the point where write errors cannot be repaired automatically. This can also result from transient errors, such as cabling problems or power problems. Check for a cabling problem.

Volume entering degraded mode

```
vxvm:vxconfigd: NOTICE: Volume volume entering degraded mode
```

❖ Clarification

The detach of a subdisk in the named RAID-5 volume has caused that volume to enter “degraded” mode. While in degraded mode, performance of the RAID-5 volume will be 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.

❖ Action

If hot-relocation is enabled, the Volume Manager objects affected by the disk failure may be taken care of automatically. Mail will be sent to `root` indicating what actions were taken by the Volume Manager and what further actions the administrator should take.

vxconfigd Warning Messages

The following are warning messages associated with `vxconfigd`.

```
Bad request: client, portal [REQUEST|DIAG], size
```



```
vxvm:vxconfigd: WARNING: Bad request number: client number, portal  
[REQUEST|DIAG], size number
```

❖ Clarification

This is a diagnostic message that indicates an invalid request generated by a utility that has connected to vxconfigd. This message indicates a bug in that connected utility.

❖ Action

If you are actually developing a new utility, then this error indicates a bug in your code. Otherwise, this error indicates a bug in the Volume Manager. Contact Customer Support for more information.

```
Cannot change disk group record in kernel
```

```
vxvm:vxconfigd: WARNING: Cannot change disk group record in kernel: reason
```

❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

```
Cannot create device
```

```
vxvm:vxconfigd: WARNING: Cannot create device device_path: reason
```

❖ Clarification

vxconfigd cannot create a device node either under /dev/vx/dsk or under /dev/vx/rdisk. This should happen only if the root file system has run out of inodes.

❖ Action

Try removing some files from the root file system. Then, regenerate the device node with the command:

```
vxctl enable
```

```
Cannot exec /usr/bin/rm to remove directory
```

```
vxvm:vxconfigd: WARNING: Cannot exec /usr/bin/rm to remove directory: reason
```



❖ Clarification

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.

❖ 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, you should restore it from somewhere.

Cannot fork to remove directory

```
vxvm:vxconfigd: WARNING: Cannot fork to remove directory directory: reason
```

❖ Clarification

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.

❖ Action

If your system is this low on memory or paging space, then your overall system performance is probably being substantially effected. Consider adding more memory or paging space.

Cannot issue internal transaction

```
vxvm:vxconfigd: WARNING: Cannot issue internal transaction: reason
```

❖ Clarification

This problem usually occurs only if there is a Volume Manager bug. However, it may occur in cases where memory is low.

❖ Action

Contact Customer Support for more information.



Cannot open log file

```
vxvm:vxconfigd: WARNING: Cannot open log file log_filename: reason
```

❖ Clarification

The `vxconfigd` console output log file could not be opened for the given reason. A log file is opened if `-x log` is specified, or if a log file is specified with `-x logfile=file`. The default log file is `/var/vxvm/vxconfigd.log`. The most likely cause for failure is “No such file or directory,” which indicates that the directory containing the log file does not exist.

❖ Action

Create any needed directories, or use a different log file path name.

Detaching plex from volume

```
vxvm:vxconfigd: WARNING: Detaching plex plex from volume volume
```

❖ Clarification

The given plex is being detached from the given volume as part of starting the volume. This error only happens for volumes that are started automatically by `vxconfigd` at system startup (i.e., for the root and `/usr` file system volumes). The plex is being detached as a result of an I/O failure, a disk failure during startup or prior to the last system shutdown or crash, or a disk removal prior to the last system shutdown or crash.

❖ Action

If you want to ensure that the root or `/usr` file system retains the same number of active mirrors, then remove the given plex and add a new mirror using the `vxassist mirror` operation. You might also consider replacing any bad disks before using `vxassist mirror`.

Disk in group flagged as shared; Disk skipped

```
vxvm:vxconfigd: WARNING: Disk disk in group group flagged as shared; Disk skipped
```

❖ Clarification

The given disk is listed as shared, but the running version of Volume Manager does not support shared disk groups. This message can usually be ignored.

❖ Action



There is no action to take. If you want to use the disk on this system, then use `vxdiskadd` to add the disk for use by the local system. However, do not do that if the disk really is in a shared disk group that is in use by other systems that are sharing this disk.

Disk in group locked by host Disk skipped

```
vxvm:vxconfigd: WARNING: Disk disk in group group locked by host hostid Disk skipped
```

❖ Clarification

The given disk is listed as locked by the host with the listed Volume Manager `hostid` (usually the same as the system hostname). This message can usually be ignored.

❖ Action

There is no action to take. If you want to use the disk on this system, then use `vxdiskadd` to add the disk for use by the local system. However, *do not* do that if the disk really is in a disk group that is in use by another system that is sharing this disk.

Disk in group: Disk device not found

```
vxvm:vxconfigd: WARNING: Disk disk in group group: Disk device not found
```

❖ Clarification

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 disk IDs stored in the Volume Manager header on a disk and disk IDs stored in the disk group configuration. The configuration contains the official list of disk IDs for all disks in a disk group (the IDs are contained in disk media configuration records). The physical disks are then scanned to match that list against the disk IDs stored in disk headers. 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 failed 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 or DRL log plexes on this disk will be unusable; any RAID-5 subdisks or mirrored plexes containing subdisks on this disk will also be unusable. These disk failures (particularly multiple disk failures) may cause one or more volumes to become unusable.

❖ Action



If hot-relocation is enabled, the Volume Manager objects affected by the disk failure may be taken care of automatically. Mail will be sent to `root` indicating what actions were taken by the Volume Manager and what further actions the administrator should take.

Disk in kernel is not a recognized type

```
vxvm:vxconfigd: WARNING: Disk disk in kernel is not a recognized type
```

❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Disk names group, but group ID differs

```
vxvm:vxconfigd: WARNING: Disk disk names group group, but group ID differs
```

❖ Clarification

As part of a disk group import, a disk was discovered that had a mismatched disk group name and disk group ID. This disk will not have been imported. This can only happen if two disk groups of the same name exist that have different disk group ID values. In that case, one group will be imported along with all its disks and the other group will not. This message will appear for disks in the un-selected group.

❖ Action

If it turns out that the disk should be imported into the group, then this will have to be done by adding the disk to the group at a later stage. It will not happen automatically as part of the import. All configuration information for the disk will also be lost.

Disk group is disabled, disks not updated with new host ID

```
vxvm:vxconfigd: WARNING: Disk group group is disabled, disks not updated with new host ID
```

❖ Clarification

As a result of failures, the named disk group has become disabled. Earlier error messages should indicate the cause of this. This warning message indicates that disks in that disk group were not updated with a new Volume Manager host ID.



This warning message should result only from a `vxctl hostid` operation.

❖ 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 Volume Manager host ID for the system. In that case, the disk group should be imported directly using `vxvg import` with the `-C` option.

```
Disk group: Disk group log may be too small
```

```
vxvm:vxconfigd: WARNING: Disk group group: Disk group log may be too small  
Log size should be at least number blocks
```

❖ Clarification

The log areas for the disk group have become too small for the size of configuration currently in the group. This should normally never happen without first displaying a message about the database area size. 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.

❖ Action

If this situation does occur, then the disks in the group will have to be explicitly reinitialized with larger log areas (which would require data to be restored from backup). See the `vxdisk(1M)` manual page. To reinitialize all of the disks, they must be detached from the group with which they are associated and then reinitialized and re-added. The disk group should then be deported and re-imported for the changes to the log areas for the group to take effect.

```
Disk group: Errors in some configuration copies: Disk, copy
```

```
vxvm:vxconfigd: WARNING: Disk group group: Errors in some configuration  
copies: Disk disk, copy number: [Block number]: reason ...
```

❖ Clarification

During a 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.

❖ Action



There are usually enough configuration copies for any disk group to ensure that these errors do not become a serious problem. No action is usually necessary.

Error in volboot file

```
vxvm:vxconfigd: WARNING: Error in volboot file: reason Entry: disk device disk_type
disk_info
```

❖ Clarification

The `/etc/vx/volboot` file includes an invalid disk entry. This error should occur only if the file was edited directly (for example, using the `vi` editor).

❖ Action

This is just a warning message. The offending entry can be removed using the command:

```
vxctl rm disk device
```

Failed to store commit status list into kernel

```
vxvm:vxconfigd: WARNING: Failed to store commit status list into kernel: reason
```

❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Failed to update voldinfo area in kernel

```
vxvm:vxconfigd: WARNING: Failed to update voldinfo area in kernel: reason
```

❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.



Field too long in volboot file

```
vxvm:vxconfigd: WARNING: Field too long in volboot file:  
Entry: disk device disk_type disk_info
```

❖ Clarification

The `/etc/vx/volboot` file includes a disk entry with a field that is larger than the size the Volume Manager supports. This error should occur only if the file was edited directly (for example, using the `vi` editor).

❖ Action

This is just a warning message. The offending entry can be removed using the command:

```
vxctl rm disk device
```

Get of record from kernel failed

```
vxvm:vxconfigd: WARNING: Get of record record_name from kernel failed: reason
```

❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Group: Duplicate virtual device number(s)

```
vxvm:vxconfigd: WARNING: Group group: Duplicate virtual device number(s):  
Volume volume remapped from major,minor to major,minor ...
```

❖ Clarification

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.

❖ Action



You should use the `vxchg remirror` operation to renumber all volumes in the offending disk group permanently. See the `vxchg(1M)` manual page for more information.

Internal transaction failed

```
vxvm:vxconfigd: WARNING: Internal transaction failed: reason
```

❖ Clarification

This problem usually occurs only if there is a Volume Manager bug. However, it may occur in cases where memory is low.

❖ Action

Contact Customer Support for more information.

cannot remove group from kernel

```
vxvm:vxconfigd: WARNING: cannot remove group group from kernel: reason
```

❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

client not recognized by VXVM library

```
vxvm:vxconfigd: WARNING: client number not recognized by VXVM library
```

❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

client not recognized

```
vxvm:vxconfigd: WARNING: client number not recognized
```



❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

library and vxconfigd disagree on existence of client

```
vxvm:vxconfigd: WARNING: library and vxconfigd disagree on existence of client number
```

❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

library specified non-existent client

```
vxvm:vxconfigd: WARNING: library specified non-existent client number
```

❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

response to client failed

```
vxvm:vxconfigd: WARNING: response to client number failed: reason
```

❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action



Contact Customer Support for more information.

vold_turnclient failed

```
vxvm:vxconfigd: WARNING: vold_turnclient(number) failed reason
```

❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

DMP Error Messages

Path Failure detected by vxdmp driver

```
vxvm:vxdmp:NOTICE: Path failure on <major>/<minor>
```

❖ Clarification

Appears when a path under the control of the DMP driver fails. The device number of the failed path is part of the message.

❖ Action

None.

Load of sd driver failed

```
vxvm:vxdmp:NOTICE: Could not load sd driver
```

❖ Clarification

Appears when a path under the control of the DMP driver fails. The device number of the failed path is part of the message.

❖ Action

None.

Install of sd driver failed

```
vxvm:vxdmp:NOTICE: Could not install sd driver
```



❖ Clarification

During initialization, if the `vxdmp` driver tries to load the `sd` driver. If the attempt fails, this message appears.

❖ Action

None.

Can't lock sd driver

`vxvm:vxdmp:NOTICE: could not lock sd driver`

❖ Clarification

The `sd` driver is locked during `vxdmp` driver initialization to avoid unloading of the driver. The message is printed when the `sd` driver cannot be locked.

❖ Action

None.

Load of ssd driver failed

`vxvm:vxdmp:NOTICE: Could not load ssd driver`

❖ Clarification

Appears when a path under the control of the DMP driver fails. The device number of the failed path is part of the message.

❖ Action

None.

Install of ssd driver failed

`vxvm:vxdmp:NOTICE: Could not install ssd driver`

❖ Clarification

During initialization, the `vxdmp` driver tries to load the `ssd` driver. If the attempt fails, this message appears.

❖ Action

None.

Can't lock `ssd` driver

`vxvm:vxndmp:NOTICE: could not lock ssd driver`

❖ Clarification

The `ssd` driver is locked during `vxndmp` driver initialization to avoid unloading of the driver. The message is printed when the `ssd` driver cannot be locked.

❖ Action

None.

Attempt to disable controller failed

`vxvm:vxndmp:NOTICE:Attempt to disable controller controller_name failed. Rootdisk has just one enabled path.`

❖ Clarification

There is only one remaining active path to the root disk. This cannot be disabled. The user is trying to disable a controller accessible through this path.

❖ Action

Do not try to disable this controller. This is not allowed.

Disabled path belonging to `dmpnode`

`vxvm:vxndmp:NOTICE:disabled path path device number belonging to dmpnode dmpnode device number`

❖ Clarification

The path with the device number indicated by the message, has been marked disabled in the DMP database. This path is controlled by the DMP node indicated by the specified device number. This could have happened due to a hardware failure.

❖ Action

Check the underlying hardware in case you want to recover the desired path.

Enabled path belonging to `dmpnode`

`vxvm:vxndmp:NOTICE:enabled path path device number belonging to dmpnode dmpnode device number`

❖ Clarification



The path with the device number indicated by the message, has been marked enabled in the DMP database. This path is controlled by the DMP node indicated by the specified device number. This can happen if a previously disabled path has been repaired and the user has reconfigured the DMP database using the `vxctl(1M)` command or, automatically.

❖ Action

None.

Disabled dmpnode

```
vxvm:vxldmp:NOTICE:disabled dmpnode dmpnode device number
```

❖ Clarification

The dmpnode with the device number indicated in the message, has been marked disabled in the DMP database. It will no longer be accessible to further IOs. It happens when all paths controlled by a DMP node are in the disabled state, and therefore inaccessible.

❖ Action

Check hardware or enable the appropriate controllers in order to get at least one path under this dmpnode in the enabled state. This will enable the dmpnode specified.

Enabled dmpnode

```
vxvm:vxldmp:NOTICE:enabled dmpnode dmpnode device number
```

❖ Clarification

The dmpnode with the device number indicated by the message has been marked enabled in the DMP database. It will now allow IOs. This happens when at least one path controlled by this dmpnode has been enabled.

❖ Action

None.

Disabled controller connected to a disk array

```
vxvm:vxldmp:NOTICE:disabled controller controller name connected to disk array disk array serial number
```

❖ Clarification

All paths that go through the specified controller connected to the specified disk array, are in disabled state. This happens user chooses to disable a particular controller for maintenance tasks.

❖ Action

None.

Enabled controller connected to a disk array

```
vxvm:vxvxdmp:NOTICE:enabled controller controller name connected to disk array disk array serial number
```

❖ Clarification

All paths that go through the specified controller connected to the specified disk array, have been put into enabled state. This happens when the user chooses to enable a particular controller.

❖ Action

None.

Removed disk array

```
vxvm:vxvxdmp:NOTICE:removed disk array disk array serial number
```

❖ Clarification

The specified disk array has been disconnected from the host or some hardware failure has resulted in the disk array becoming inaccessible to the host.

❖ Action

None.

Added disk array

```
vxvm:vxvxdmp:NOTICE:added disk array disk array serial number
```

❖ Clarification

The new disk array has been added to the host.

❖ Action

None.

Attempt to enable a controller that is not available

```
vxvm:vxvxdmpadm:ERROR:Attempt to enable a controller that is not available
```

❖ Clarification



This message is returned by the `vxdmpadm` utility when an attempt to enable a controller that is not working or physically present is made. To enable a controller, it should be visible to the OS and I/Os should be possible through it.

❖ **Action**

Check hardware and see if this controller is present and I/Os can be done through it

Attempt to enable a controller that is not available

```
vxvm:vxdmpadm:ERROR:Attempt to enable a controller that is not available
```

❖ **Clarification**

This message is returned by the `vxdmpadm` utility when an attempt to enable a controller that is not working or physically present is made. To enable a controller, it should be visible to the OS and I/Os should be possible through it.

❖ **Action**

Check hardware and see if this controller is present and I/Os can be done through it

Attempt to start the restore daemon that is already running

```
vxvm:vxdmpadm: ERROR:The VxVM restore daemon is already running. To change any of its parameters, stop and restart the restore daemon with the new set of arguments.
```

❖ **Clarification**

`vxdmpadm` utility returns this error message when the `vxdmpadm start restore` command is executed while the restore daemon is running.

❖ **Action**

Stop the restore daemon. Restart the restore daemon with the required set of parameters.

Attempt to disable the last enabled path to a device

```
vxvm:vxdmpadm: ERROR: 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.
```

❖ **Clarification**

Disabling this controller could lead to some devices being inaccessible.

❖ **Action**



To disable the only path connected to a disk, use the `-f` option.

Cluster Error Messages

This section presents error messages that may occur with the Volume Manager in a cluster environment. Each message is accompanied by an explanation and a suggested user action.

Note Some of these messages may appear on the console; others are returned by `vxclust`.

Error in cluster processing

❖ Clarification

This may be due to an operation inconsistent with the current state of the cluster (such as an attempt to import or deport a shared disk group from the slave). It may also be caused by an unexpected sequence of commands from `vxclust`.

❖ Action

Make sure that the operation can be performed in the current environment.

Cannot find disk on slave node

❖ Clarification

A slave node cannot find a shared disk. This is accompanied by the `syslog` message:

```
vxvm:vxconfigd cannot find disk disk
```

❖ 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 connections to the disks.

Disk in use by another cluster

❖ Clarification

An attempt was made to import a disk group whose disks are stamped with the ID of another cluster.

❖ Action



If the disk group is not imported by another cluster, retry the import using the `-C` (clear import) flag.

`vxclust not there`

❖ **Clarification**

An error during an attempt to join the cluster caused `vxclust` to fail. This may be caused by the failure of another node during a join or by the failure of `vxclust`.

❖ **Action**

Retry the join. An error message on the other node may clarify the problem.

`Unable to add portal for cluster`

❖ **Clarification**

`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).

❖ **Action**

If the system does not appear to be degraded, stop and restart `vxconfigd` and try again.

`Vol recovery in progress`

❖ **Clarification**

A node that crashed attempted to rejoin the cluster before its DRL map was merged into the recovery map.

❖ **Action**

Retry the join again later (when the merge operation has completed).

`Cannot assign minor #`

❖ **Clarification**

The slave attempted to join, but an existing volume on the slave has the same minor number as a shared volume on the master.

This message should be accompanied by the following console message:

Caution vxvm:vxconfigd minor number ### disk group group in use

❖ Action

Before retrying the join, use `vx dg remminor diskgroup ###` (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 through system restart.

Master sent no data

❖ Clarification

During the slave join protocol, a message without data was received. This message is only likely to be seen in the case of a programming error.

❖ Action

Contact Customer Support for more information.

Join in progress

❖ Clarification

An attempt was made to import or deport a shared disk group during a cluster reconfiguration.

❖ Action

Retry later.

Join not allowed now

❖ Clarification

A slave attempted to join the cluster when the master was not ready. The slave will retry automatically. If the retry succeeds, the following message should appear:

`vxclust: slave join complete`

❖ Action

No action is necessary if the join eventually completes. Otherwise, investigate the cluster monitor on the master.



Disk reserved by other host

❖ **Clarification**

An attempt to online a disk whose controller has been reserved by another host will fail with this error.

❖ **Action**

No action is necessary. The cluster manager will free the disk and the Volume Manager will online it when the node joins the cluster.

Group exists

```
vxvm:vxconfigd: group group exists
```

❖ **Clarification**

The slave tried to join the cluster, but there is already a shared disk group in the cluster with the same name as one of its private disk groups.

❖ **Action**

Use the `vxchg newname` operation to rename either the shared disk group on the master or the private disk group on the slave.

Plex detached from volume

```
WARNING: vxvm:vxio: Plex plex detached from volume volume  
NOTICE: vol_kmsg_send_wait_callback: got error 22  
NOTICE: commit: NOTE: Reason found for abort: code=6
```

❖ **Clarification**

These messages may appear during a plex detach operation on a slave.

❖ **Action**

These messages provide information and require no user action.

Read error on Plex of shared volume; Plex detached from volume

```
WARNING: vxvm:vxio: read error on Plex plex of shared volume volume offset 10  
length 1  
WARNING: vxvm:vxio: Plex plex detached from volume volume  
NOTICE: commit: NOTE: Reason found for abort: code=2
```

```
NOTICE: ktcvm_check: sent to slave node: node=1 mid=196
```

❖ **Clarification**

These messages may appear during a plex detach operation on the master.

❖ **Action**

These messages provide information and require no user action.

```
Return from cluster_establish is Configuration daemon error 242
```

❖ **Clarification**

This error may occur when a node fails to join the cluster or when a cluster join takes a long time. If the join fails, the node should retry the join automatically.

❖ **Action**

No action is necessary if the join is slow or a retry eventually succeeds.

```
Missing vxconfigd
```

❖ **Clarification**

The vxconfigd daemon is not running.

❖ **Action**

Restart the vxconfigd daemon.

```
vxconfigd not ready
```

```
node #: vxconfigd is not communicating properly
```

❖ **Clarification**

The vxconfigd daemon is not responding properly.

❖ **Action**

Stop and restart the vxconfigd daemon.

```
vxiod count must be above # to join cluster
```

```
ERROR: vxiod count must be above 5 to join cluster
```



❖ Clarification

The number of Volume Manager kernel daemons (`vxiod`) is less than the minimum number needed to join the cluster.

❖ Action

Increase the number of daemons using `vxiod`.

Insufficient DRL log size: logging is disabled.

❖ Clarification

A volume with an insufficient DRL log size was started. The volume gets started successfully, but DRL logging is disabled and a full recovery is performed.

❖ Action

Create a new DRL of sufficient size.

This node was running different CM. Please Reboot.

❖ Clarification

VxVM supports clustering under the control of various cluster managers. However, once a node joins the cluster under a particular cluster manager, it cannot be restarted under a different cluster manager until it is rebooted.

❖ Action

Reboot the host machine if the cluster must be started under a different cluster manager.

Clustering license restricts operation

❖ Clarification

An operation requiring a full clustering license was attempted, and such a license is not available.

❖ 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.

Node activation conflict

❖ Clarification



The disk group could not be activated because it is activated in a conflicting mode on another node in the cluster.

❖ Action

Retry later, or deactivate the disk group on conflicting nodes.

Incorrect protocol version *num* in volboot file

❖ Clarification

This message may appear when a node attempts to join the cluster if VxVM software was incorrectly upgraded or the volboot file is corrupted.

❖ Action

Verify the supported cluster protocol versions using `vxctl protocolversion`, and reinstall the Volume Manager if necessary.

Retry rolling upgrade

❖ Clarification

An attempt was made to upgrade the cluster to a higher protocol version when a transaction was in progress.

❖ Action

Retry later.

Version out of range for at least one node

❖ Clarification

One or more nodes in the cluster do not support the protocol version that would result from a protocol upgrade.

❖ Action

Make sure that the latest version of VxVM is installed on all nodes in the cluster.

Upgrade operation failed: Version out of range for at least one node

❖ Clarification



Before trying to upgrade a cluster by running `vxdctl upgrade`, all the nodes should be able to support the new protocol version. If at least one of them doesn't support the the new protocol version, an upgrade can fail.

❖ **Action**

Make sure that the VxVM package that supports the new protocol version is installed on all nodes and retry the upgrade.

Upgrade operation failed: Error in cluster processing

❖ **Clarification**

The cluster protocol upgrade must be done on the master and cannot be done from a slave node.

❖ **Action**

Retry the `vxdctl upgrade` command on the master node.

CVM protocol version out of range

❖ **Clarification**

When a node joins the cluster, it tries to join at a 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.

❖ **Action**

Make sure that the joining node has a VxVM release installed that supports the current protocol version of the cluster.

ERROR: upgrade operation failed: Already at highest version

❖ **Clarification**

The message conveys that the upgrade operation has failed because the cluster is already running at the highest protocol version that the master supports.

❖ **Action**

No further action is necessary as the master is already running at the highest protocol version it can support.

Not in cluster

❖ **Clarification**

Checking for the current protocol version (`vxctl protocolversion`) makes sense only if the node is in the cluster.

❖ **Action**

Bring the node in the cluster and retry.

Incorrect protocol version (15) in volboot file.

❖ **Clarification**

The `volboot` file contains an incorrect protocol version. It has either been corrupted or edited manually. The `volboot` file should always have a supported protocol version in it before trying to bring the node into the cluster.

❖ **Action**

Run `vxctl init`. This writes a valid protocol version in the `volboot` file. Restart `vxconfigd` and retry the join.

Upgrade operation failed: Retry rolling upgrade

❖ **Clarification**

No transactions should be in progress when an upgrade is tried.

❖ **Action**

Retry the upgrade later.



Kernel Error Messages

The following sections cover the kernel level error messages.

Kernel Notice Messages

The following are notice messages associated with the kernel.

Can't open disk in group

`vxvm:vxio:NOTICE: Can't open disk disk in group disk_group. If it is removable media (like a floppy), it may not be mounted or ready. Otherwise, there may be problems with the drive. Kernel error code number`

❖ Clarification

The named disk cannot be accessed in the named disk group.

❖ Action

Ensure that the disk exists, is powered on, and is visible to the system.

Can't close disk in group

`vxvm:vxio:NOTICE: Can't close disk disk in group disk_group. If it is removable media (like a floppy), it may have been removed. Otherwise, there may be problems with the drive. Kernel error code public_region_error/private_region_error`

❖ Clarification

This is unlikely to happen; closes cannot fail.

❖ Action

None.

Read error on object of mirror in volume corrected

`vxvm:vxio:NOTICE: read error on object subdisk of mirror plex in volume volume (start offset, length length) corrected.`

❖ Clarification

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.

❖ Action

No action is required. The problem was corrected automatically. The administrator may, however, note the failure as a reference because if the same region fails again or frequently, the error could indicate a more insidious failure and the disk should be reformatted at the next reasonable opportunity.

String on volume device in disk group

```
vxvm:vxio:NOTICE: string on volume device_# (device_name) in disk group group_name
```

❖ Clarification

An application requested message. The application running on top of the Volume Manager has requested the output of this message.

❖ Action

Refer to documentation for the appropriate application for more information.

Kernel Warning Messages

The following are warning messages associated with the kernel.

Received spurious close

```
vxvm:vxio:WARNING: Device major, minor: Received spurious close
```

❖ Clarification

This message happens if a close was received for an object that was previously not opened. This will only happen if the operating system is not correctly tracking opens and closes.

❖ Action

No action is necessary; the system will continue.

Failed to log the detach of the DRL volume

```
vxvm:vxio:WARNING: Failed to log the detach of the DRL volume volume
```

❖ Clarification

An attempt to write a kernel log entry indicating the loss of a DRL volume failed. 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 will become detached.



❖ Action

Messages about log failures are often 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 (i.e. the drive cannot be fixed and brought back online without data loss), the disk group must be recreated from scratch and all of its volumes must be restored from backup. Even if the problem is transient, the system must be rebooted after correcting the problem.

If error messages were seen from the disk driver, it is likely that the last copy of the log failed due to a disk error. The failed drive in the disk group should be replaced and the log will then be re-initialized on the new drive. The failed volume can then be forced into an active state and the data recovered.

DRL volume is detached

```
vxvm:vxio:WARNING: DRL volume volume is detached
```

❖ Clarification

A Dirty Region Logging volume became detached because a DRL log entry could not be written. This might be because of a media failure, in which case other errors may have been logged to the console.

❖ Action

The volume containing the DRL log will continue. If the system fails before the DRL can be repaired, a full recovery of the volume's contents may be necessary and will be performed automatically when the system is restarted. To recover the DRL capability, a new DRL log should be added to the volume using the `vxassist addlog` command.

Read error on mirror of volume

```
vxvm:vxio:WARNING: read error on mirror plex of volume volume offset offset  
length length
```

❖ Clarification

An error was detected while reading a mirror. This error may lead to further action shown by later error messages.

❖ Action

If the volume is mirrored, no action is necessary at this time 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 marginal region on the disk at the

position shown. It may eventually be necessary to remove data from this disk (see the `vxevac(1M)` manual page) and then to reformat the drive. In the unmirrored case, 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.

Write error on mirror of volume offset length

```
vxvm:vxio:WARNING: write error on mirror plex of volume volume offset offset  
length length
```

❖ Clarification

An error was detected while writing a mirror. This error will generally be followed by a detach message, unless the volume is un-mirrored.

❖ Action

The disk reporting the error is failing to correctly store written data. If the volume is not mirrored, consider removing the data and reformatting the disk. If the volume is mirrored, it will become detached and you should consider replacing or reformatting the disk.

If this error occurs often but never leads to a plex detach, there may be a marginal region on the disk at the position shown. It may eventually be necessary to remove data from this disk (see the `vxevac(1M)` manual page) and then to reformat the drive.

Object detached from volume

```
vxvm:vxio:WARNING: object plex detached from volume volume
```

❖ Clarification

An uncorrectable error was detected by the mirroring code and a mirror copy was detached.

❖ Action

To restore redundancy, it may be necessary to add another mirror. The disk on which the failure occurred should be evacuated and reformatted, if possible. If the drive has failed completely, it may need to be replaced.

Overlapping mirror detached from volume

```
vxvm:vxio:WARNING: Overlapping mirror plex detached from volume volume
```

❖ Clarification



An error has occurred on the last complete plex in the mirrored volume. Any sparse mirrors that map the failing region must be detached so that they cannot be accessed to satisfy that failed region inconsistently. This message indicates that such an overlapping mirror was found and is being detached.

❖ Action

No Action is directly necessary. The message indicates that the volume may have left some data inaccessible at the failing region and that it is no longer redundantly stored.

Kernel log full

```
vxvm:vxio:WARNING: Kernel log full: volume detached
```

❖ Clarification

A plex detach failed because the kernel log was full. As a result, the mirrored volume will become detached.

❖ Action

It is unlikely that this condition could ever occur. The only corrective action for the detached volume is to reboot the system.

Kernel log update failed

```
vxvm:vxio:WARNING: Kernel log update failed: volume detached
```

❖ Clarification

A plex detach failed because the kernel log could not be flushed to disk. As a result, the mirrored volume will become detached. This could be caused by all the disks containing a kernel log going bad.

❖ Action

Correct the failed disks so that kernel logging can once again function.

Detaching RAID-5 volume

```
vxvm:vxio:WARNING: detaching RAID-5 raidvol
```

❖ Clarification

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.

❖ Action

If two or more drives were lost due to a controller or power failure, then once the disks can be re-attached to the system, they should be recovered using the `vxrecover` utility. Check the console for other errors that may provide additional information as to the nature of the failure.

Object detached from RAID-5 volume

```
vxvm:vxio:WARNING: object subdisk detached from RAID-5 raidvol at column column
offset offset
```

❖ Clarification

A subdisk was detached from a RAID-5 volume at the specified column number and offset. This is caused by the failure of a disk or an uncorrectable error occurring on that disk.

❖ Action

Check the console for other error messages indicating the cause of the failure. If the disk has failed, then it should be replaced as soon as possible.

RAID-5 volume entering degraded mode operation

```
vxvm:vxio:WARNING: RAID-5 raidvol entering degraded mode operation
```

❖ Clarification

This message occurs when an uncorrectable error has forced the detach of a subdisk. At this point, not all data disks exist to provide the data upon request. Instead, parity regions are required to regenerate the data for each stripe in the array. Accesses will consequently take longer and will involve reading from all drives in the stripe.

❖ Action

Check the console for other error messages indicating the cause of the failure. If the disk has failed, it should be replaced as soon as possible.

Double failure condition detected on RAID-5 volume

```
vxvm:vxio:WARNING: Double failure condition detected on RAID-5 raidvol
```

❖ Clarification



Double-failures occur if I/O errors are received at the same altitude in the array from more than one column of the array. This could be caused by a controller failure causing more than a single drive to become unavailable; by the loss of a second drive after having run in a degraded state for significant periods of time; or by two separate disk drives failing simultaneously (which is unlikely to happen).

❖ Action

If the condition is correctable and the drives recoverable, the conditions should be corrected. The volume can then be recovered using the `vxrecover(1M)` command.

Failure in RAID-5 logging operation

```
vxvm:vxio:WARNING: Failure in RAID-5 logging operation
vxvm:vxio:WARNING: log object object_name detached from RAID-5 volume
```

❖ Clarification

These two messages will appear together when a RAID-5 log has failed and has been detached.

❖ Action

To restore RAID-5 logging to the RAID-5 volume, simply create a new log region and attach it to the volume.

Stranded ilock on object

```
vxvm:vxio:WARNING: check_ilocks: stranded ilock on object_name start offset len
length
```

❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Overlapping ilocks

```
vxvm:vxio:WARNING: check_ilocks: overlapping ilocks: offset for length, offset
for length
```

❖ Clarification



This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.

Illegal vminor encountered

```
vxvm:vxio:WARNING: Illegal vminor encountered
```

❖ Clarification

If a volume device other than the root volume device is opened before a configuration has been loaded, this message could result.

❖ Action

No action should be necessary; an attempt to access a volume device was made before the volume daemon (vxconfigd) loaded the volume configuration. Under normal startup conditions, this message should not occur. If the operation is necessary, start the Volume Manager and re-attempt the operation.

Uncorrectable read error

```
vxvm:vxio:WARNING: object_type object_name block offset: Uncorrectable read error
```

❖ Clarification

A read or write operation from the specified object failed. An error will be returned to the application.

❖ Action

This error represents lost data. The data may need to be restored and failed media may need to be repaired. Depending on the type of object failing and on the type of recovery suggested for that type, an appropriate recovery operation may be necessary.

Uncorrectable read/write error

```
vxvm:vxio:WARNING: object_type object_name block offset:  
Uncorrectable read error on object_type object_name block offset  
vxvm:vxio:WARNING: object_type object_name block offset:  
Uncorrectable write error on object_type object_name block offset
```

❖ Clarification



A read or write operation from the specified object failed. An error will be returned to the application. Although similar to the previous message, this message is able to supply more specific information about the failing object.

❖ Action

This error represents lost data. The data may need to be restored and failed media may need to be repaired. Depending on the type of object failing and on the type of recovery suggested for that type, an appropriate recovery operation may be necessary.

Root volumes are not supported on your PROM version

```
vxvm:vxio:WARNING: Root volumes are not supported on your PROM version.
```

❖ Clarification

The Volume Manager requires the ability to access the PROMs for your SPARC hardware. If the PROMs are not a recent OpenBoot PROM type, then root volumes will not be usable.

❖ Action

If you have set up a root volume, then undo the configuration (by running `vxunroot` or removing the `rootdev` line from `/etc/system`) as soon as possible and contact your hardware vendor for an upgrade to your PROM level.

Cannot find device number

```
vxvm:vxio:WARNING: Cannot find device number for boot_path
```

❖ Clarification

The supplied boot path was retrieved from the PROMs for your system. It cannot be converted to a valid device number.

❖ Action

Check your PROM settings for the correct boot string.

mod_install returned *errno*

```
vxvm:vxio:WARNING: mod_install returned errno
```

❖ Clarification

A call made to the operating system `mod_install()` function to load the `vxio` driver failed.

❖ Action

Check your console for additional 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.

```
subdisk failed in plex in volume
```

```
vxvm:vxio:WARNING: subdisk subdisk failed in plex plex in volume volume
```

❖ Clarification

The kernel has detected a subdisk failure, which may mean that the underlying disk is failing.

❖ Action

Check for obvious problems with the disk (such as a disconnected cable). If hot-relocation is enabled and the disk is failing, the subdisk failure may be taken care of automatically.

Kernel Panic Messages

The following are panic messages associated with the kernel.

Object association depth overflow

```
vxvm:vxio:PANIC: Object association depth overflow
```

❖ Clarification

This is an internal Volume Manager problem. This warning should not occur unless there is a bug in the Volume Manager.

❖ Action

Contact Customer Support for more information.





Introduction

This chapter describes traditional disk arrays and provides a general overview of arrays. The following topics are discussed in this chapter:

- ◆ Disk Array Overview
- ◆ Multipathed Disk Arrays

Disk Array Overview

This section provides an overview of traditional disk arrays.

Performing I/O to disks is a slow process because disks are physical devices that require time to move the heads to the correct position on the disk before reading or writing. If all of the read or write operations are done to individual disks, one at a time, the read-write time can become unmanageable. Performing these operations on multiple disks can help to reduce this problem.

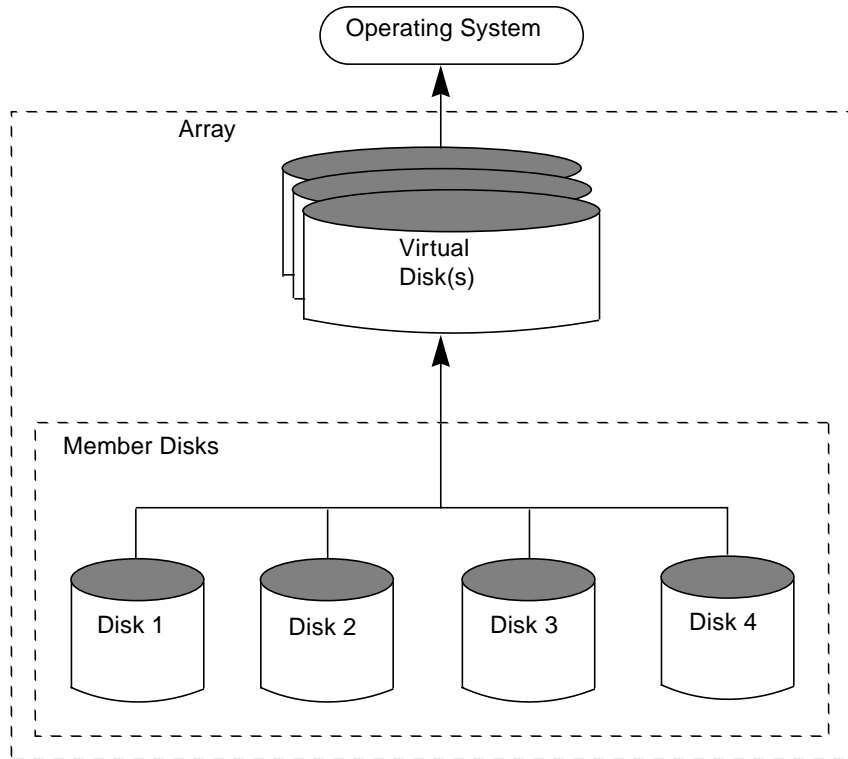
A *disk array* is a collection of disks that appears to the system as one or more virtual disks (also referred to as *volumes*). The virtual disks created by the software controlling the disk array look and act (to the system) like physical disks. Applications that interact with physical disks should work exactly the same with the virtual disks created by the array.

Data is spread across several disks within an array, which allows the disks to share I/O operations. The use of multiple disks for I/O improves I/O performance by increasing the data transfer speed and the overall throughput for the array.

Figure 1 shows a standard disk array.



Figure 1. Standard Disk Array



Redundant Arrays of Independent Disks (RAID)

A *Redundant Array of Independent Disks* (RAID) is a disk array set up so that part of the combined storage capacity is used for storing duplicate information about the data stored in the array. The duplicate information allows you to regenerate the data in case of a disk failure.

Several levels of RAID exist. These are introduced in the following sections.

Note The Volume Manager supports RAID levels 0, 1, and 5 only.

For information on the Volume Manager's implementations of RAID, refer to "Volume Manager and RAID-5." in the *VERITAS Volume Manager Administrator's Guide*.

RAID-0

Although it does not provide redundancy, striping is often referred to as a form of RAID, known as RAID-0. The Volume Manager's implementation of striping is described in "Striping (RAID-0)," in Chapter 1 of the *VERITAS Volume Manager Administrator's Guide*. RAID-0 offers a high data transfer rate and high I/O throughput, but suffers lower reliability and availability than a single disk.

RAID-1

Mirroring is a form of RAID, which is known as RAID-1. The Volume Manager's implementation of mirroring is described in the *VERITAS Volume Manager Administrator's Guide*. Mirroring uses equal amounts of disk capacity to store the original plex and its mirror. Everything written to the original plex is also written to any mirrors. RAID-1 provides redundancy of data and offers protection against data loss in the event of physical disk failure.

RAID-2

RAID-2 uses bitwise striping across disks and uses additional disks to hold Hamming code check bits. RAID-2 is described in a University of California at Berkeley research paper entitled *A Case for Redundant Arrays of Inexpensive Disks (RAID)*, by David A. Patterson, Garth Gibson, and Randy H. Katz (1987).

RAID-2 deals with error detection, but does not provide error correction. RAID-2 also requires large system block sizes, which limits its use.

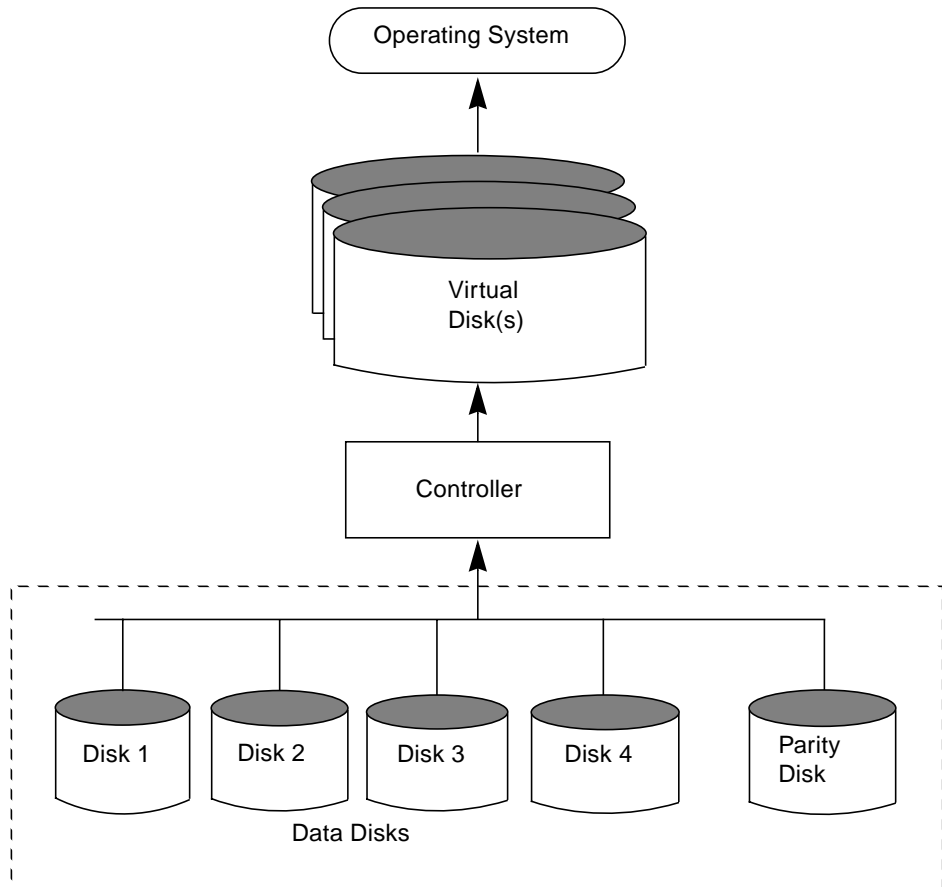
RAID-3

RAID-3 uses a *parity disk* to provide redundancy. RAID-3 distributes the data in stripes across all but one of the disks in the array. It then writes the parity in the corresponding stripe on the remaining disk. This disk is the parity disk.



Figure 2 shows a RAID-3 disk array.

Figure 2. RAID-3 Disk Array

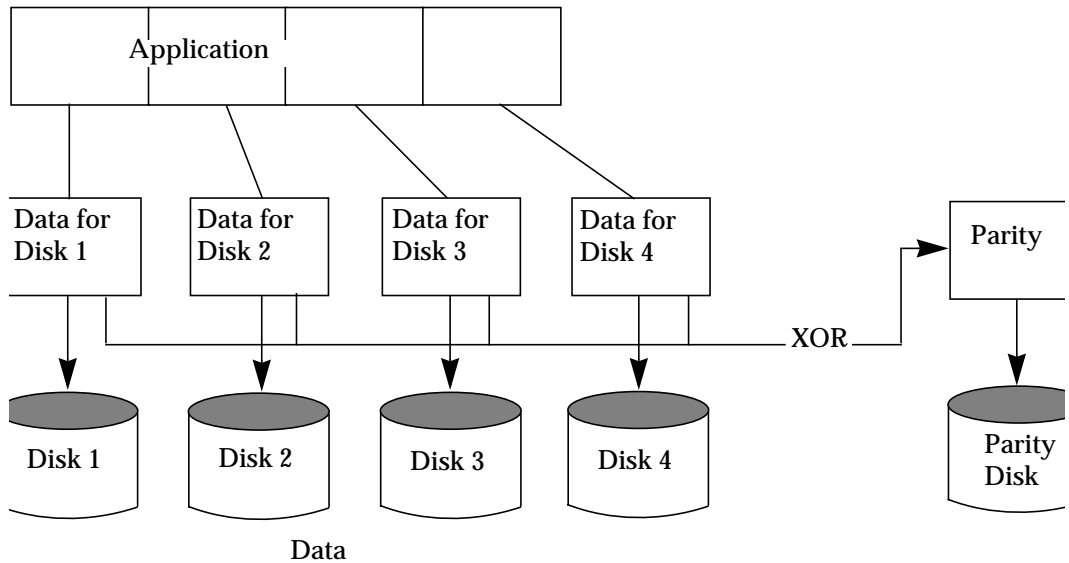


The user data is striped across the data disks. Each stripe on the parity disk contains the result of an exclusive OR (XOR) procedure done on the data in the data disks. If the data on one of the disks is inaccessible due to hardware or software failure, data can be restored by XORing the contents of the remaining data disks with the parity disk. The data on the failed disk can be rebuilt from the output of the XOR process.

RAID-3 typically uses a very small stripe unit size (also historically known as a *stripe width*), sometimes as small as one byte per disk (which requires special hardware) or one sector (block) per disk.

Figure 3 shows a data write to a RAID-3 array.

Figure 3. Data Writes to RAID-3



The parity disk model uses less disk space than mirroring, which uses equal amounts of storage capacity for the original data and the copy.

The RAID-3 model is often used with synchronized spindles in the disk devices. This synchronizes the disk rotation, providing constant rotational delay. This is useful in large parallel writes.

RAID-3 type performance can be emulated by configuring RAID-5 (described later) with very small stripe units.

RAID-4

RAID-4 introduces the use of independent-access arrays (also used by RAID-5). With this model, the system does not typically access all disks in the array when executing a single I/O procedure. This is achieved by ensuring that the stripe unit size is sufficiently large that the majority of I/Os to the array will only affect a single disk (for reads).

An array attempts to provide the highest rate of data transfer by spreading the I/O load as evenly as possible across all the disks in the array. In RAID-3, the I/O load is spread across the data disks, as shown in Figure 3, and each write is executed on all the disks in the array. The data in the data disk is XORed and the parity is written to the parity disk.



RAID-4 maps data and uses parity in the same manner as RAID-3, by striping the data across all the data disks and XORing the data for the information on the parity disk. The difference between RAID-3 and RAID-4 is that RAID-3 accesses all the disks at one time and RAID-4 accesses each disk independently. This allows the RAID-4 array to execute multiple I/O requests simultaneously (provided they map to different member disks), while RAID-3 can only execute one I/O request at a time.

RAID-4 read performance is much higher than its write performance. It performs well with applications requiring high read I/O rates. RAID-4 performance is not as high in small, write-intensive applications.

The parity disk can cause a bottleneck in the performance of RAID-4. This is because all the writes that are taking place simultaneously on the data disks must each wait its turn to write to the parity disk. The transfer rate of the entire RAID-4 array in a write-intensive application is limited to the transfer rate of the parity disk.

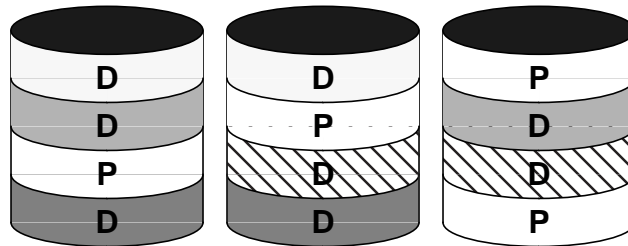
Since RAID-4 is limited to parity on one disk only, it is less useful than RAID-5.

RAID-5

RAID-5 is similar to RAID-4, using striping to spread the data over all the disks in the array and using independent access. However, RAID-5 differs from RAID-4 in that the parity is striped across all the disks in the array, rather than being concentrated on a single parity disk. This breaks the write bottleneck caused by the single parity disk write in the RAID-4 model.

Figure 4 shows parity locations in a RAID-5 array configuration. Every stripe has a column containing a parity stripe unit and columns containing data. The parity is spread over all of the disks in the array, reducing the write time for large independent writes because the writes do not have to wait until a single parity disk can accept the data.

Figure 4. Parity Locations in a RAID-5 Model



D = Data Stripe

For additional information on RAID-5 and how it is implemented by the Volume Manager, refer to “Volume Manager and RAID-5.” in the *VERITAS Volume Manager Administrator’s Guide*.

Multipathed Disk Arrays

Some disk arrays provide multiple ports to access their disk devices. These ports, coupled with the HBA (Host Bus Adaptor) controller and any array-local data bus and I/O processor, make up multiple hardware paths to access the disk devices. Such disk arrays are called *multipathed disk arrays*. This type of disk array can be connected to host systems in many different configurations, (such as multiple ports connected to different controllers on a single host, chaining of the ports through a single controller on a host, or ports connected to different hosts simultaneously).

Multipathed disk arrays can be classified as either Active/Active type, or Active/Passive type.

Active/Passive Type Disk Arrays

This type of disk array designates one of the multiple paths to a disk device as the primary path, and the others as secondary paths. Access to a disk is enabled through its primary path. If the primary path fails, one of the secondary paths is made the new primary path to the disk device, either automatically, or through administrator intervention. Access to the disk device through a secondary path may be disabled, or may degrade system performance severely.



In this type of disk array, a disk device is bound to one of the redundant hardware components local to the disk array, (for example, the I/O bus, I/O controller, cache and access port). This disk is then said to be owned by the port, and the I/O path through that port is designated as the active path

Some disk arrays have a mode setting, called autotresspass mode, in which the disk arrays automatically mark a path as primary path when I/O is attempted through that path. This switch of primary path is an expensive operation, and degrades the performance of the disk array severely if different paths are used to access the disk alternately, by the same or different hosts.

Active/Active Type Disk Arrays

This type of disk array allows access to the disk devices simultaneously (at any time) through all the paths that are available, without significant performance degradation. Thus, all the paths are active all the time, except for failed paths.

Index

A

- active/active disk arrays 94
- active/passive disk arrays 93
- arrays
 - disk 87

D

- daemons
 - configuration 19, 24
- defaults file
 - vxassist 12
- description file 14
- disk arrays 87
 - active/active 94
 - active/passive 93
- DMP configuration 19

M

- mirroring 89

P

- plexes
 - attach 15
 - detach 15

R

- RAID 88
- RAID-0 89
- RAID-1 89
- RAID-2 89
- RAID-3 89
- RAID-4 91
- RAID-5 92

S

- spindles, synchronized 91
- striping 89

U

- utility descriptions
 - vxassist 10

- vxctl 19
- vxedit 17
- vxmake 14
- vxmend 18
- vxplex 15
- vxprint 18
- vxsd 16
- vxstat 18
- vxvol 15

V

- volumes
 - kernel state 15
 - operations 19
- vxassist 10, 11
 - defaults 11
 - description of 10
- vxconfigd 19, 24
- vxctl 19
 - description of 19
- vxdg 17
- vxdisk 17
- vxdiskadd 16
- vxdiskadm 16
- vxedit
 - description of 17
- vxmake 14
 - description of 14
- vxmend 18
- vxplex 15
 - description of 15
- vxprint 18
 - description of 18
- vxsd 16
 - description of 16
- vxstat 18
 - description of 18
- vxtrace 18
- vxvol 15



description of 15

W
writes
parallel 91

