SPARCstorage Array User's Guide

Sun Microsystems Computer Company
A Sun Microsystems, Inc. Business
2550 Garcia Avenue
Mountain View, CA 94043 U.S.A.
415 960-1300 FAX 415 969-9131

Part No.: 802-2042-10
Revision A, March 1995

Contents

Part 2—Using the Graphical User Interface

Part 4—Reference for the Graphical User Interface

Figures

Tables

Preface

The *SPARCstorage Array User's Guide* tells how to operate the SPARCstorage™ Array using either the graphical user interface (GUI) or the command line interface (CLI). It also gives reference information for the GUI.

Note – Read the *SPARCstorage Array Product Note* (802-2043-xx) before reading this manual to get the latest software information for the SPARCstorage Array.

Who Should Use This Book

This manual is written for an inexperienced user.

Before You Read This Book

If you are unfamiliar with the different levels of disk management offered by the SPARCstorage Array (such as mirroring and striping), then read the *SPARCstorage Array Configuration Guide* before proceeding with this manual.

How This Book Is Organized

The information in this manual is divided into five parts:

Part 1 — Description and Installation

Chapter 1, "Disk Management for the SPARCstorage Array" explains the different levels of disk management offered by the SPARCstorage Array.

Chapter 2, "Initializing the Software," tells how to install and initialize the software on the SPARCstorage Array.

Part 2 — Using the Graphical User Interface

Chapter 3, "Getting Started," gives instructions on bringing up the GUI and performing the basic tasks to get your system running.

Chapter 4, "Setting Up the Software Configuration," gives instructions on setting up your data structure to take advantage of the disk management tools available through Volume Manager using the GUI.

Chapter 5, "Changing Components of the Software Configuration," gives instructions on changing the components of the data structure once you've set it up using the GUI.

Chapter 6, "Maintaining the Data," gives instructions on performing system administration tasks using the GUI.

Chapter 7, "Performing Service-Related Software Tasks," gives the procedures you will need to perform for certain hardware service tasks using the GUI.

Part 3 — Using the Command Line Interface

Chapter 8, "Getting Started," gives the basic tasks to get your system running.

Chapter 9, "Setting Up the Software Configuration," gives instructions on setting up your data structure to take advantage of the disk management tools available through Volume Manager using the CLI.

Chapter 10, "Changing Components of the Software Configuration," gives instructions on changing the components of the data structure once you've set it up using the CLI.

Chapter 11, "Maintaining the Data," gives instructions on performing system administration tasks using the CLI.

Chapter 12, "Performing Service-Related Software Tasks," gives the procedures you will need to perform for certain hardware service tasks by using the CLI.

Part 4 — Reference for the Graphical User Interface

Chapter 13, "Reference for the Root Window," gives reference information for the root window for the GUI.

Chapter 14, "Reference for the View of SPARCstorage Array," gives reference information for the View of SPARCstorage Array of the GUI.

Chapter 15, "Reference for Volume Manager Views," gives reference information for the Volume Manager windows of the GUI.

Part 5 — Tutorial

Chapter 16, "Tutorial for the Graphical User Interface."

The *SPARCstorage Array User's Guide* also has five appendixes:

Appendix A, "Using the Volume Manager Software on Non-SPARCstorage Array Disks," gives specific instructions for using the Volume Manager software on non–SPARCstorage Array disks.

Appendix B, "GUI-Related X Resources," lists X resources that can be used to configure the GUI according to your preferences and system requirements.

Appendix C, "Volume Manager Error Messages," lists the error messages that you may see when running the Volume Manager software and what actions to take for each.

Appendix D, "SPARCstorage Array Firmware and Device Driver Error Messages," lists SPARCstorage Array-specific error messages.

Appendix E, "Recovery," tells how to recover from disk failures.

Related Documentation

Refer to the following documentation for supporting information.

Table P-1 Related Documentation

Name	Part Number
SPARCstorage Array Model 100 Series Installation Manual	801-2205-xx
SPARCstorage Array Model 100 Series Service Manual	801-2206-xx
SPARCstorage Array Model 200 Series Installation Manual	802-2027-xx
SPARCstorage Array Model 200 Series Service Manual	802-2028-xx
SPARCstorage Array Configuration Guide	802-2041-xx

What Typographic Changes and Symbols Mean

The following table describes the type changes and symbols used in this book.

Table P-2 Typographic Conventions

Typeface or Symbol	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories; on-screen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% You have mail.`
`AaBbCc123`	What you type, contrasted with on-screen computer output	`machine_name%` **`su`** `Password:`
AaBbCc123	Command-line placeholder: replace with a real name or value	To delete a file, type `rm` *filename*.
AaBbCc123	Book titles, new words or terms, or words to be emphasized	Read Chapter 6 in *User's Guide*. These are called *class* options. You *must* be root to do this.

Shell Prompts in Command Examples

The following table shows the default system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

Table P-3 Shell Prompts

Shell	Prompt
C shell prompt	`machine_name%`
C shell superuser prompt	`machine_name#`
Bourne shell and Korn shell prompt	`$`
Bourne shell and Korn shell superuser prompt	`#`

Part 1— Description and Installation

Disk Management for the SPARCstorage Array 1

Note – Refer to the *SPARCstorage Array Configuration Guide* (P/N 802-2041-xx) for more information on the hardware and software configurations for the SPARCstorage Array.

The SPARCstorage Array takes advantage of several different levels of disk management, some available through the standard SPARCstorage Array software and some available through the Volume Manager software.

1.1 Disk Management With the Standard SPARCstorage Array Software

The SPARCstorage Array offers several disk management features, including fast write capability and reserve/release capability.

1.1.1 Description of Fast Write

Usually when a system goes through a synchronous write for data without using the fast write capability, it goes through the following process:

1. The program issues the write command.
2. The command and data are sent to the SPARCstorage Array controller.
3. The SPARCstorage Array controller sends the command and data to the disk drive.
4. The drive executes the write command.
5. The drive tells the SPARCstorage Array controller that the write was completed successfully.
6. The SPARCstorage Array controller tells the host system that the write was completed successfully.
7. The program issues the next command.

This process can take anywhere from a few milliseconds to tens of milliseconds, because the host system must wait for confirmation from the drive that the write was executed successfully before the program can issue the next command.

If you use the fast write option, however, the data is stored on the NVRAM on the SPARCstorage Array controller for a short period of time instead of writing it directly to the disk. The SPARCstorage Array sends the status back to the system saying that the data was written to the disk (even though it really wasn't), which clears the way for the system to send more information over to the SPARCstorage Array. The SPARCstorage Array continues to store the data on the NVRAM and then flushes all the data from the NVRAM to the disk periodically.

So using the fast write option on a synchronous write for data results in a process similar to the following:

1. The program issues the write command.
2. The command and data are sent to the SPARCstorage Array controller.
3. The SPARCstorage Array controller tells the host system that the write was completed successfully.
4. The program issues the next command.
5. The SPARCstorage Array controller sends the command and data to the disk drive.
6. The drive executes the write command.
7. The drive tells the SPARCstorage Array controller that the write was completed successfully.

In the fast write scenario, if the SPARCstorage Array experiences a power failure after the SPARCstorage Array controller has told the host system that the write was completed (Step 3) and before the drive has told the SPARCstorage Array controller that the write was completed (Step 7), the SPARCstorage Array controller firmware will flush the write to the disk when the array is powered up again.

If the SPARCstorage Array is about to be powered off or if a disk drive is about to be made inaccessible while data is still sitting in the NVRAM, the SPARCstorage Array allows you to *flush* any outstanding writes from the NVRAM to the drives manually instead of waiting for the array to do it.

In some situations, though, flushing the writes to the drives won't work because the disk drive has failed. If flushing the data to the drive won't work, you will have to *purge* the data from the NVRAM. The instructions for flushing and purging writes are given later in this manual.

The fast write option is beneficial because it:

- decreases latency for individual commands;
- tends to provide a more consistent response time for write commands;
- can significantly improve performance for applications that do a large amount of synchronous writes, as long as the amount of data being written doesn't overwhelm the capacity of the NVRAM cache.

There are several potential drawbacks to the fast write option, however:

- Management of the fast write NVRAM adds overhead to the SPARCstorage Array controller. So maximum total throughput with fast writes enabled will be lower than it is when fast writes are disabled. (This is true with any write-caching product.)
- If the write to the drive fails for some reason (for example, if the drive fails), the application that issued the write will not know that the write failed, since it had been told already that the write succeeded (an error message is displayed on the system console when this happens, however). For this reason, it is often recommended that you use mirroring or RAID-5 so that individual drive failures can be survived without loss of data.

1.1.2 Description of Reserve/Release

The reserve command allows a host system to reserve individual drives or all the drives in a SPARCstorage Array so that no other host systems can use those drives. This is useful if you have more than one host system connected to a SPARCstorage Array but you want only one system to be able to access certain drives. The release command releases these drives from their reserved state.

1.2 Disk Management With the SPARCstorage Volume Manager

The SPARCstorage Array is supplied with a disk management software tool called the SPARCstorage Volume Manager. The Volume Manager can be used to increase data availability and improve disk I/O performance. It builds transparent layers called *volumes* on top of physical disks, which contain data managed by a Solaris® file system, a database, or other applications; disk I/O commands then operate on the volumes rather than the physical disks. Once volumes are created, you can optimize performance, change volume size, add mirrors, and perform backups or other administrative tasks on one or more physical disks without interrupting system users on the remaining disks.

Note – You can operate the SPARCstorage Array without the Volume Manager and treat it as a subsystem of independent SCSI disks, or you can use a different logical volume management tool instead of the SPARCstorage Volume Manager. However, because the SPARCstorage Volume Manager provides a rich set of features that enhances performance and makes disk administration easier, you may prefer to use the SPARCstorage Volume Manager for most applications.

In order to understand what Volume Manager does, you must understand the relationships between physical objects and Volume Manager objects.

1.2.1 Physical Objects

For performing disk management tasks using Volume Manager, you need to understand two physical objects:

- Physical disks
- Partitions

1.2.1.1 Physical Disks

A physical disk can be accessed using a device name, or *c#t#d#*, where *c#* is the controller, *t#* is the target ID, and *d#* is the disk number. The disk in Figure 1-1 is disk number 0 with a target ID of 0, and it is connected to controller number 0 in the system.

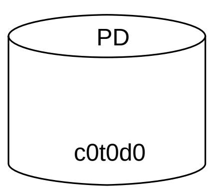

Figure 1-1 Physical Disk (Example)

1.2.1.2 Partitions

A physical disk can be divided into one or more *partitions*. The partition number, or *s#*, is given at the end of the device name. Note that a partition can take up an entire physical disk, such as the partition shown in Figure 1-2.

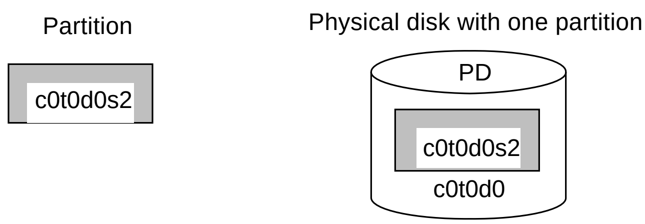

Figure 1-2 Partition (Example)

Up to 16 VTOC (volume table of contents) partitions (also called slices) can be created on the physical disk with a Solaris 2.x software environment. These partitions are named, in order, 0 through 15. Partition 2 is reserved to indicate the entire disk.

There are two ways that you can bring a disk under Volume Manager control: *initialization* and *encapsulation*. When you *initialize* a disk, all data on the disk is destroyed, and the Volume Manager creates and manages VTOC partitions itself. It creates two partitions on each physical disk: a small partition called the *private* region, in which it keeps its disk label and other administrative data, and a large partition called the *public* region that covers the remainder of the disk.

If you have data on a disk that you don't want to lose, you can choose to *encapsulate* the disk. However, in order for Volume Manager to encapsulate the disk, it must have two free slots in the VTOC and two free cylinders. To view a disk's VTOC, enter the following at the prompt:

```
prtvtoc device_name
```

where `device_name` is given in the form *c#t#d#s#*.

Note that you can encapsulate the root disk either during the `vxinstall` process while installing Volume Manager (which is explained in Chapter 2, "Initializing the Software,") or from the `vxdiskadm` menus after Volume Manager has been installed.

The relationship between physical objects and Volume Manager objects is established when you bring a partition from a physical disk under Volume Manager control.

Note – The procedure for bringing a physical disk under Volume Manager control is given in the following sections:

- Using the GUI — Section 3.3 on page 3-13
- Using the CLI — Section 8.2 on page 8-6

1.2.2 *Volume Manager Objects*

You must understand several Volume Manager objects before you can use Volume Manager to perform disk management tasks (note that these Volume Manager objects are used for disk management using either the graphical user interface [GUI] or the command line interface [CLI]):

- VM disks
- Disk groups
- Subdisks
- Plexes
- Volumes

1.2.2.1 *VM Disks*

When you place a partition from a physical disk under Volume Manager control, a *VM disk* is assigned to the partition. The Volume Manager allocates storage from VM disks. A VM disk has a one-to-one relationship with a partition. A VM disk is accessed using a *disk media name*, which you can supply (the default disk media name used by the Volume Manager is *disk##*). Figure 1-3 shows a VM disk with a disk media name of disk01 that is assigned to the partition c0t0d0s2.

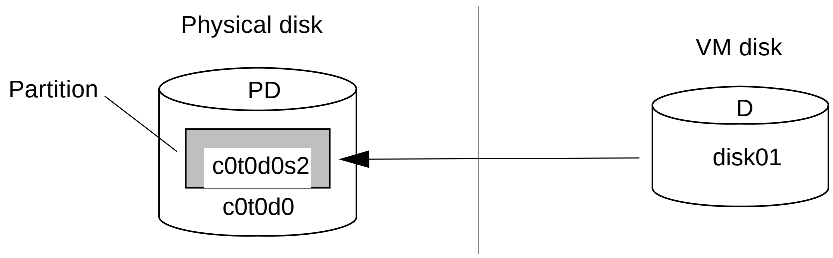

Figure 1-3 VM Disk (Example)

Because applications access volumes (created on VM disks) rather than partitions, the easiest way to use Volume Manager is to have only one partition for each physical disk that you want to be under Volume Manager control.

1.2.2.2 *Disk Groups*

When you assign a VM disk to a partition, the VM disk also becomes part of a *disk group*. A disk group is a collection of VM disks that have something in common. Disk groups allow you to group disks into logical collections for administrative convenience. For example, you could:

- place all the disks within a single SPARCstorage Array into a single disk group,
- split all the disks within a single SPARCstorage Array into several disk groups, or,
- combine disks from several SPARCstorage Arrays into a single disk group.

A VM disk needs to belong to a disk group before it can be used by the Volume Manager. However, VM disks do not have to be added to disk groups – they can be initialized and left unadded. This is useful for disk replacement. A VM disk that is initialized but not added to a disk group can be used immediately as a replacement if another disk fails.

Every disk group must have at least one VM disk in it, so you cannot remove the last VM disk from a disk group. Every disk group on a system must also have a unique name. You can create your own disk groups or use the `rootdg` disk group available through the Volume Manager. The main function of the `rootdg` disk group is to house the boot disk for the host system; therefore, there must one, and only one, `rootdg` disk group for every host system attached to the SPARCstorage Array. Because every host system must have one `rootdg` disk group and every disk group in a system must have a unique name, you cannot move the `rootdg` disk group from one host system to another.

Having excessively large disk groups may cause the private region to fill. In the case of larger disk groups, disks should be set up with larger private areas to log in with. Alternatively, large disk groups can periodically be split off into smaller disk groups.

About 80 percent of a disk's private region consists of a disk group configuration database containing configuration records for each Volume Manager object in that disk group. Since each configuration record takes up 256 bytes (or half a block), the number of configuration records that can be created in a disk group is twice the configuration database copy size (this can be obtained from the output of the command `vxdg list` *`diskgroupname`*).

1.2.2.3 Subdisks

A VM disk can be divided into one or more *subdisks*. Subdisks are the basic units in which the Volume Manager allocates disk space. Since the default name for a VM disk is *disk##* (such as disk01), the default name for a subdisk is *disk##-##* . So, for example, disk01-01 would be the name of the first subdisk on the VM disk named disk01.

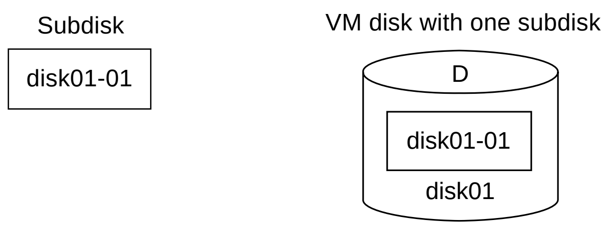

Figure 1-4 Subdisk (Example)

Volume Manager also uses *log subdisks* to log disk activity for Dirty Region Logging, or DRL (DRL is explained in greater detail in Section 1.2.5, "Dirty Region Logging"). A log subdisk icon has double borders to distinguish them from regular subdisk icons (see Figure 1-5).

Figure 1-5 Log Subdisk (Example)

A VM disk may contain more than one subdisk. The example given in Figure 1-6 shows a VM disk with three subdisks.

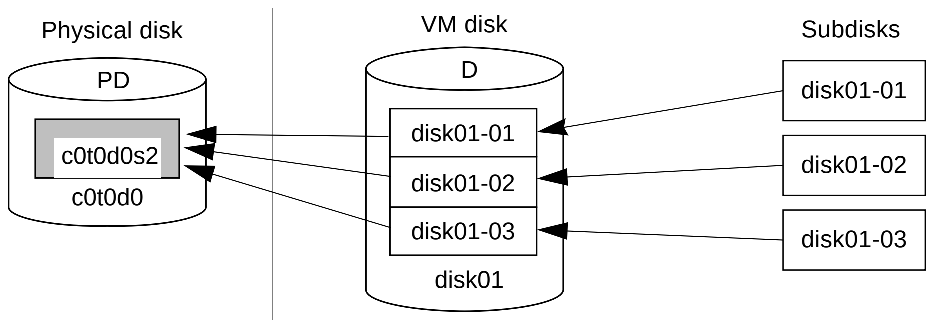

Figure 1-6 VM Disk with Three Subdisks (Example)

1.2.2.4 Plexes

The Volume Manager uses subdisks to build virtual devices, which are called *plexes*. A plex consists of one or more subdisks that can be located on one or more physical disks. There are four ways that data can be allocated on the subdisks that constitute a plex:

- simple
- concatenation
- striping
- RAID 5

Simple

A simple plex has only one subdisk. Since a subdisk cannot span multiple physical disks, a simple plex has all its storage on a single physical disk. Addresses are mapped linearly from the beginning of the subdisk to the end.

Figure 1-7 illustrates a simple plex.

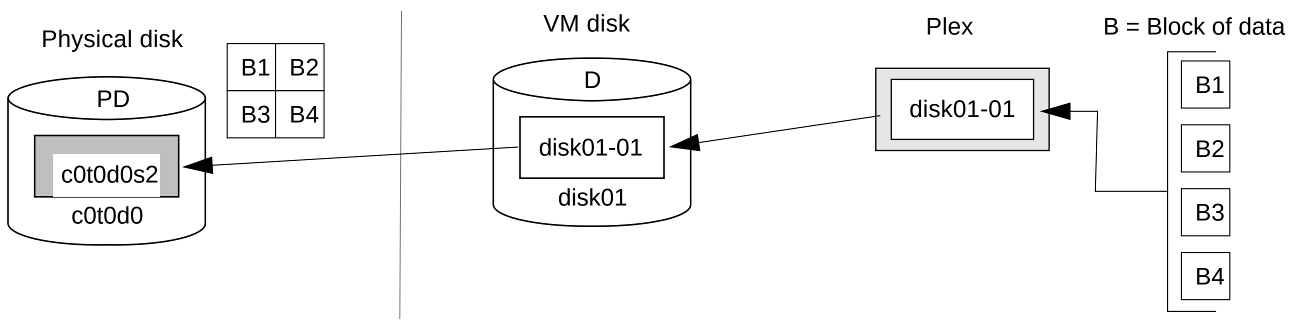

Figure 1-7 Simple Plex (Example)

Concatenation

Concatenation maps data in a linear manner onto one or more subdisks in a plex. If you were to access all the data in a concatenated plex sequentially, you would first access the data in the first subdisk from beginning to end, then access the data in the second subdisk from beginning to end, and so forth until the end of the last subdisk.

The subdisks in a concatenated plex do not have to be physically contiguous. The subdisks can also be located on more than one physical disk. Concatenation using subdisks that reside on more than one physical disk is also called *spanning*. Concatenation with more than one subdisk is useful when there is insufficient contiguous space for the plex on any one disk.

Figure 1-8 shows how data would be spread over two subdisks in a spanned plex.

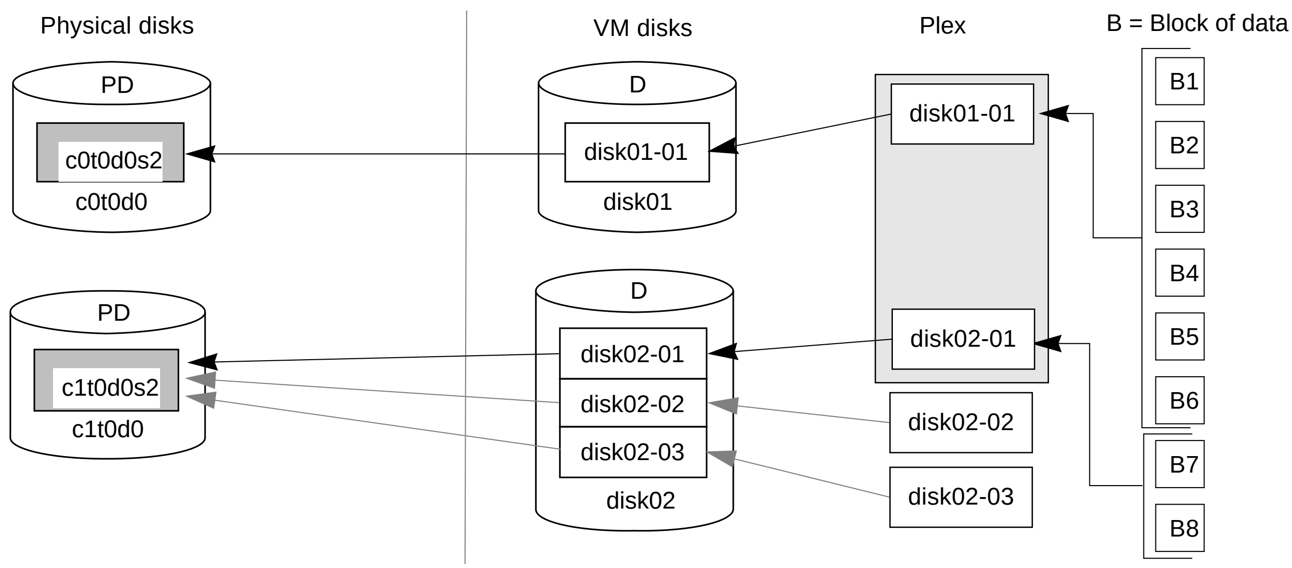

Figure 1-8 Spanned Plex (Example)

Note that since the first six blocks of data (B1 through B6) took up most or all of the room on the partition that VM disk01 is assigned to, subdisk disk01-01 is alone on VM disk disk01. However, the last two blocks of data, B7 and B8, take up only a portion of the room on the partition that VM disk02 is assigned to. That means that the remaining free space on VM disk02 can be put to other uses. In this example, subdisks disk02-02 and disk02-03 are used in other plexes.

Striping

Striping maps data so that it is interleaved among two or more physical disks. More specifically, a striped plex contains two or more subdisks, spread out over two or more physical disks. The subdisks are grouped into "columns," with one column per physical disk. Each column contains one or more subdisks and can be derived from one or more physical disks. The number and sizes of subdisks per column can vary. Additional subdisks can be added to columns, as necessary.

Data is allocated in equal sized blocks ("stripe blocks") that are interleaved between the columns. For example, if there are two columns in the striped plex and six blocks of data, the first, third, and fifth block of data would be allocated in column 1, while the second, fourth and sixth block would be allocated in column 2. Viewed in sequence, if you were striping data over three physical disks, the first block of data would go in the first column, the second block would go in the second column, and the third block would go in the third column. The fourth block of data would then go back to the first column, the fifth block would go in the second column, and the sixth block would go in the third column.

Striping is useful if you need large amounts of data to be written to or read from the physical disks quickly by using parallel data transfer to multiple disks. Striping is also a way to balance the I/O load from multi-user applications across multiple disks.

Figure 1-9 shows a simple striped plex with three equal sized, single-subdisk columns. There is one column per physical disk.

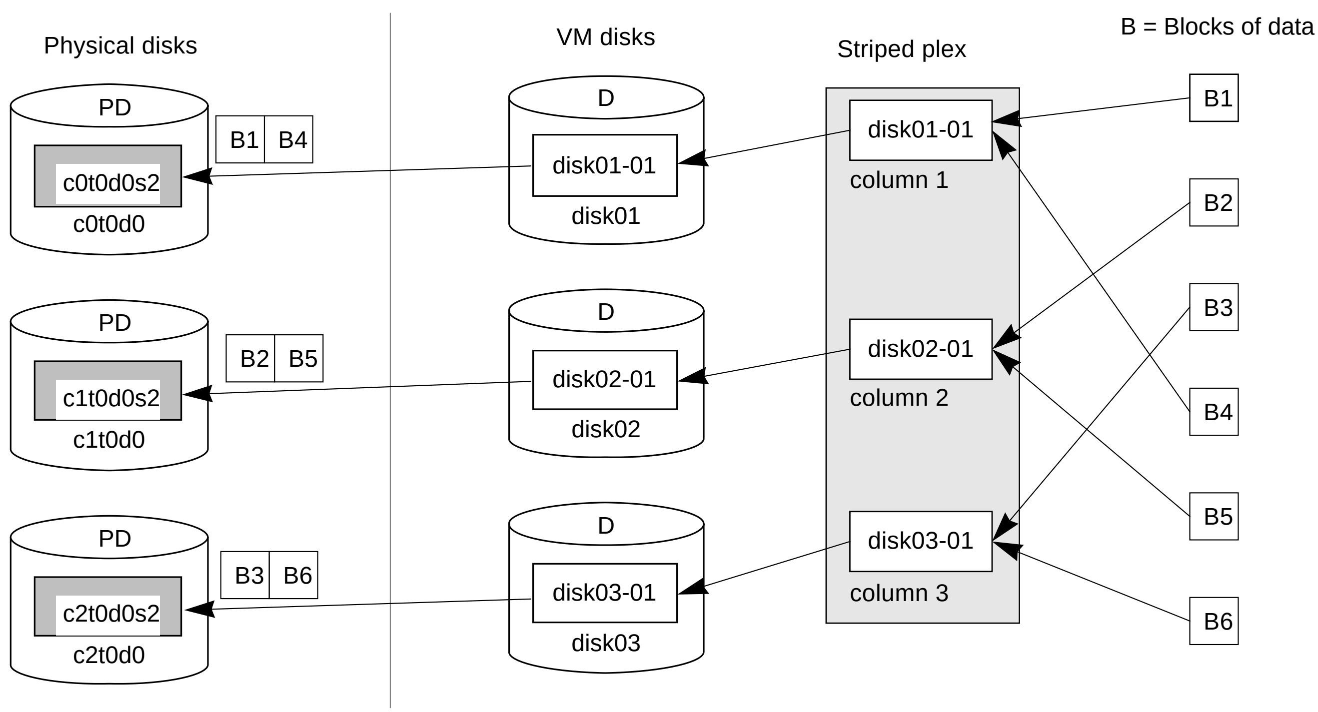

Figure 1-9 Striped Plex with One Subdisk per Column (Example)

Note that while the example in Figure 1-9 shows three subdisks that take up all of the VM disks, it is also possible for each subdisk in a striped plex to take up only a portion of the VM disk, thereby leaving free space for other subdisks.

Figure 1-10 shows a striped plex with three columns containing subdisks of different sizes. There is one column per physical disk. Each column in a striped plex must have the same number of disk sectors.

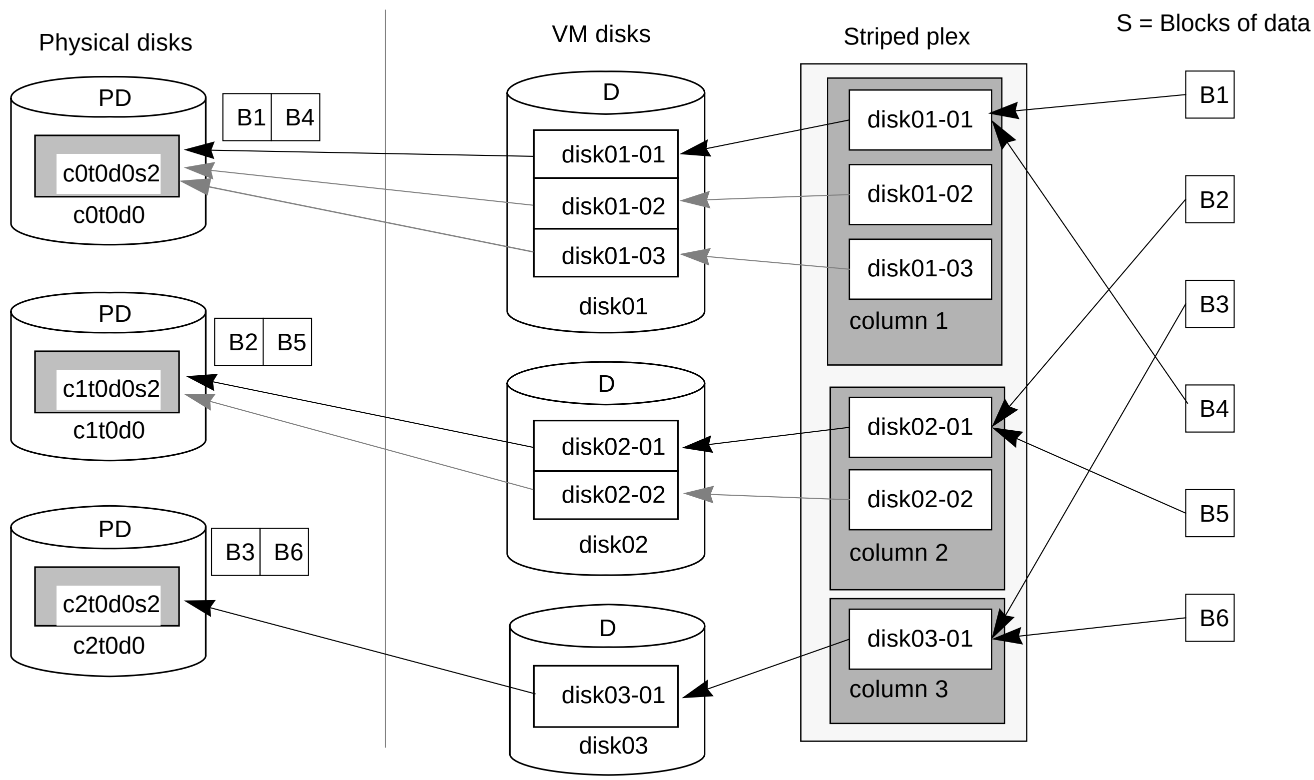

Figure 1-10 Striped Plex with Multiple Subdisks per Column (Example)

One disadvantage of striping is that some configuration changes are harder to perform on striped plexes. For example, it is not possible to move an individual subdisk of a striped plex. However, striping does offer the advantage that load balancing can be achieved easily. Figure 1-11 gives an example of a single file system that has been identified as being a data access bottleneck. This file system was striped across four disks, leaving the remainder of those four disks free for use by less-heavily used file systems.

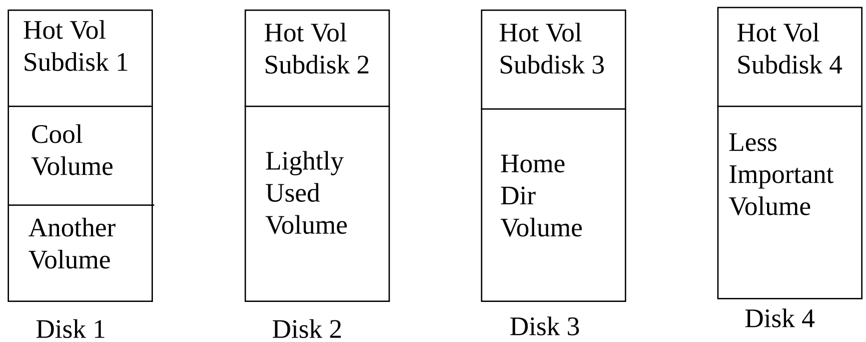

Figure 1-11 Use of Striping for Optimal Data Access

Follow these guidelines when using striping:

- Calculate stripe sizes carefully. It is often best to set the stripe width to be the length of a disk track (you can obtain the track length using the `prtvtoc` command). If it is not feasible to set the stripe width to the track size, use 64 kilobytes for the stripe width.
- For random access I/O, avoid small stripe widths. Small stripe widths can result in poor system performance unless the stripe width times the number of stripe columns exactly matches the size of the I/O requests being done at the application layer.
- For sequential access I/O, you can achieve optimal performance when the I/O size exactly matches the stripe width times the number of columns.
- Never put more than one column of a striped plex on the same physical disk.
- Typically, the greater the number of physical disks in the stripe, the greater the improvement in I/O performance (up to about six drives). However, this reduces the effective mean time between failures of the volume. If this is an issue, striping can be combined with mirroring to provide a high-performance volume with improved reliability.
- Where possible, distribute the subdisks of a striped volume across drives connected to different controllers and buses.
- You cannot include a non-SPARCstorage Array disk within a striped configuration.

Note – The procedure for setting up a striped configuration is given in the following sections:

- Using the GUI — Section 4.2 on page 4-16
- Using the CLI — Section 9.2 on page 9-6

RAID 5

A RAID 5 configuration is similar to a striped configuration in that data is spread out evenly across several disks in a system. However, a RAID 5 configuration adds a *parity* block to the stripe to provide redundancy. This parity block contains the result of an exclusive OR (XOR) procedure done on the data in the data stripes. If the data on one of the disks in the RAID 5 configuration becomes inaccessible due to a 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.

When a write is performed on disks set up in a RAID 5 configuration, the data is written to all but one of the disks; the remaining disk then gets the parity information written to it. The next write performed on the disks in a RAID 5 configuration would work the same way, except that the parity information would be written to a different disk than that used in the first write. That way, the parity information gets spread across all the disks in the configuration, so that if one disk fails, the data from that disk can be restored using the parity information written on the remaining disks in the configuration.

For example, if three disks were used in a RAID 5 configuration, data would be written to those disks in this manner: for the first set of writes, data would be written to the first two disks in the configuration and the parity information would be written to the third disk. On the second set of writes to the disks, the data would be written to the first and third disks, and the parity information would be written to the second. On the third set of writes, the data would be written to the second and third disks, and the parity information would be written to the first. The fourth write would be performed as the first write was, with data on the first and second disks and parity information on the third. Figure 1-12 shows how RAID 5 information would be written to disks using this setup.

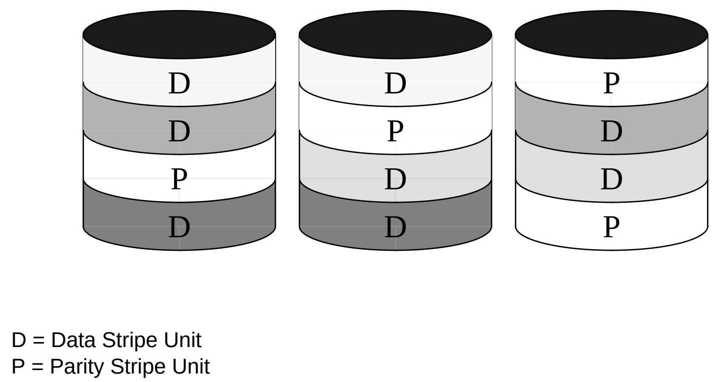

Figure 1-12 Graphical Representation of RAID 5 Writes

Logging

Without logging, if the system fails simultaneously with the failure of a disk during the write process, there is no way of knowing if the data being written to the data portions of the disks or the parity being written to the parity stripes have actually been written. Therefore, the recovery of the corrupted disk may be corrupted itself.

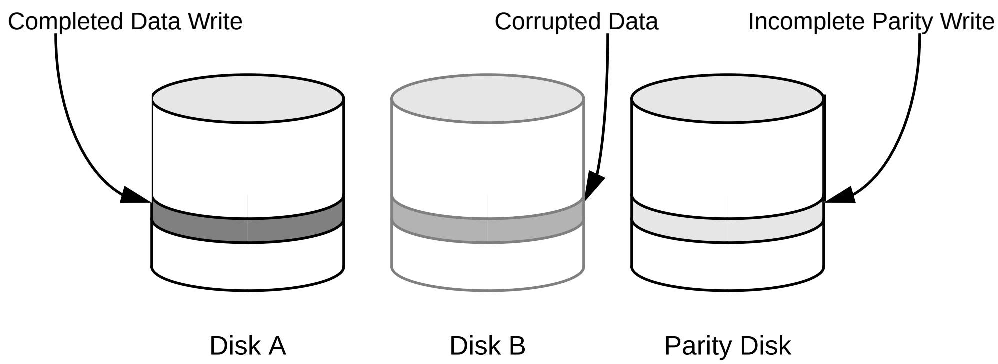

Figure 1-13 Incomplete Write

Logging is used to prevent this corruption of recovery data. A log of the new data and parity is made on a persistent, device (such as a disk-resident volume). The new data and parity are then written to the disks. Logging of RAID-5 volumes is optional, but it is very strongly recommended.

Read-Modify-Write

When you write to drives in a RAID-5 configuration, the following steps may be followed for each stripe involved in the I/O:

1. The data stripe units to be updated with new write data are accessed and read into internal buffers. The parity stripe unit is read into internal buffers.
2. The parity is updated to reflect the contents of the new data region. First, the contents of the old data undergo an exclusive OR (XOR) with the parity (logically removing the old data). The new data is then XORed into the parity (logically adding the new data). The new data and new parity is written to a log.
3. The new parity is written to the parity stripe unit. The new data is written to the data stripe units. The entire stripe is written in a single write.

This process is known as a *read-modify-write* cycle. If a disk fails, both data and parity stripe units on that disk become unavailable. The disk array is then said to be operating in a *degraded* mode.

The read-modify write sequence is illustrated in Figure 1-14.

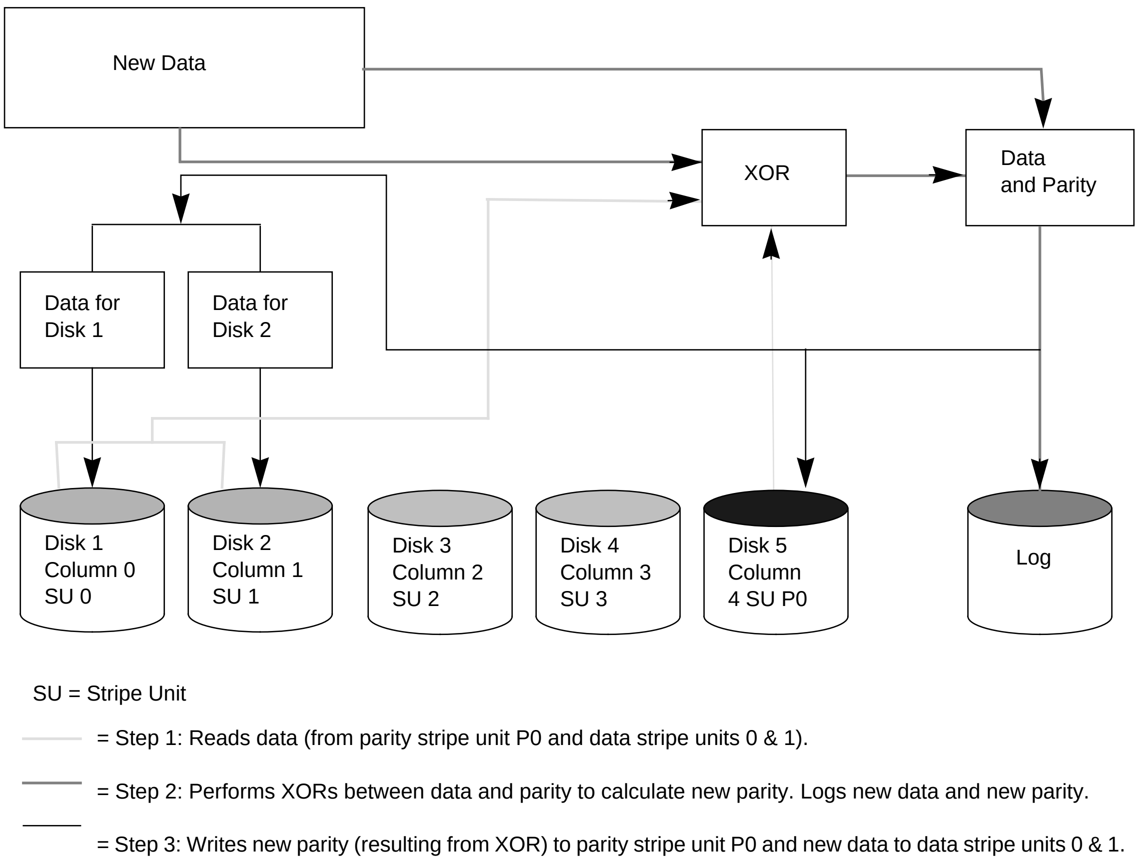

Figure 1-14 Read-Modify-Write

Full-Stripe Writes

When large writes (writes that cover an entire data stripe) are issued, the read-modify-write procedure can be bypassed in favor of a *full-stripe write*. A full-stripe write is faster than a read-modify-write because it does not require the read process to take place and it requires fewer XOR operations.

A full-stripe write procedure consists of the following steps:

1. All the new data stripe units are XORed together, generating a new parity value. The new data and new parity is written to a log.

2. The new parity is written to the parity stripe unit. The new data is written to the data stripe units. The entire stripe is written in a single write.

Figure 1-15 shows a full-stripe write.

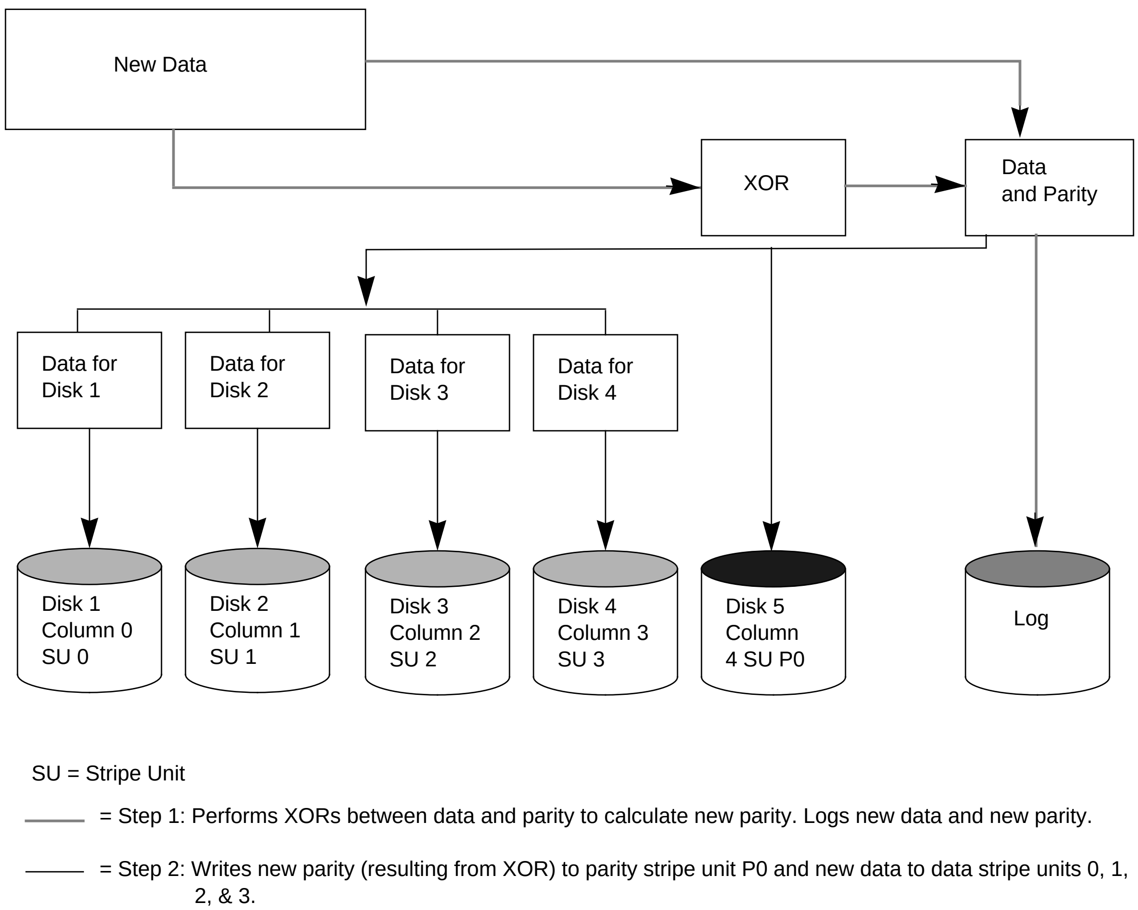

Figure 1-15 Full-Stripe Write

Reconstruct-Writes

When half or more of the data drives are being written to in a single I/O, a *reconstruct-write* can be used. A reconstruct-write saves I/O time because it does not require a read of the parity region and only requires a read of the unaffected data in the stripe.

A reconstruct-write procedure consists of the following steps:

1. Unaffected data is read from the unchanged data stripe unit(s).
2. The new data is XORed with the old, unaffected data to generate a new parity stripe unit. The new data and resulting parity is logged.
3. The new parity is written to the parity stripe unit. The new data is written to the data stripe units. The entire stripe is written in a single write.

Figure 1-16 illustrates a reconstruct-write. A reconstruct-write is preferable to a read-modify-write in this situation because it reads only the necessary data disks, rather than reading the disks and the parity disk.

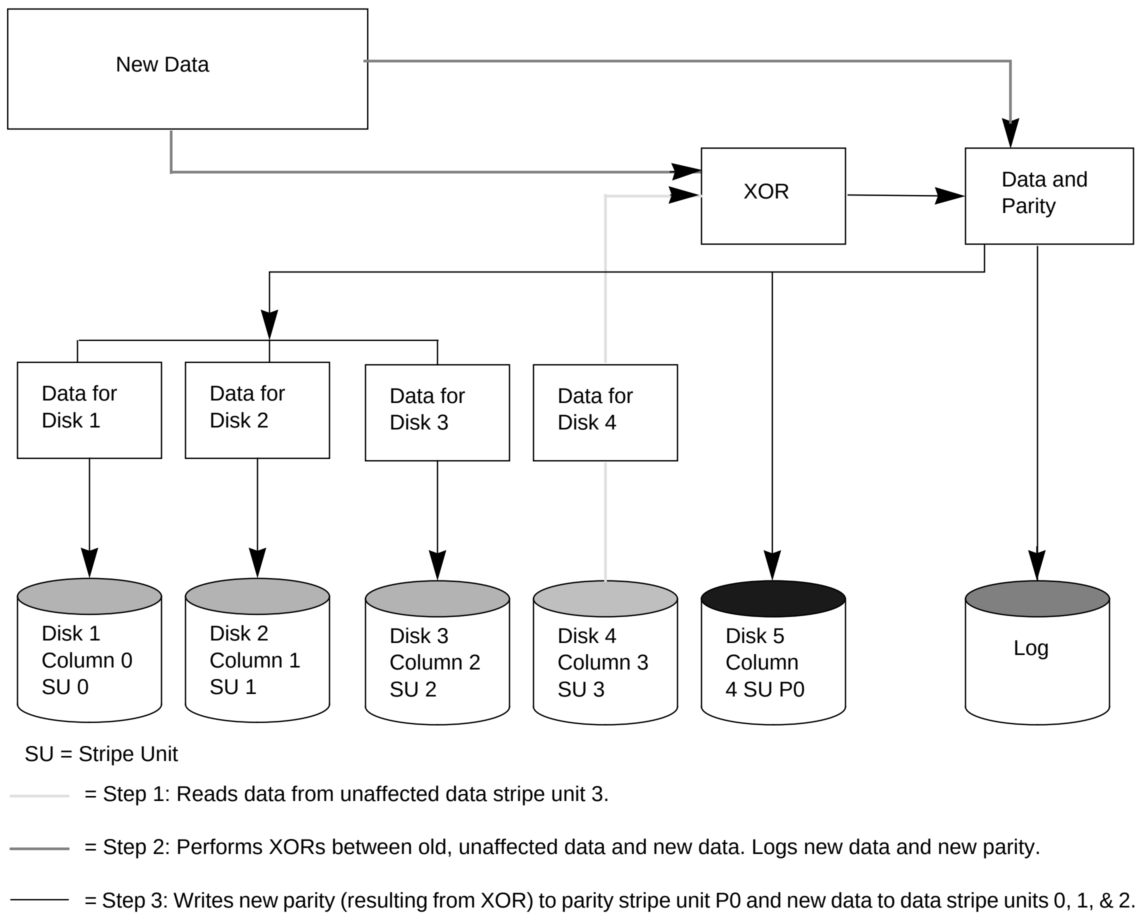

Figure 1-16 Reconstruct-Write

Follow these guidelines when using RAID-5:

- Calculate stripe sizes carefully. It is often best to set the stripe width to be the length of a disk track (you can obtain the track length using the `prtvtoc` command). If it is not feasible to set the stripe width to the track size, use 64 kilobytes for the stripe width.

- For random access I/O, avoid small stripe widths. Small stripe widths can result in poor system performance unless the stripe width times the number of stripe columns exactly matches the size of the I/O requests being done at the application layer.
- For sequential access I/O, you can achieve optimal performance when the I/O size exactly matches the stripe width times the number of columns.
- Never put more than one column of a RAID-5 plex on the same physical disk.
- Where possible, distribute the subdisks of a RAID-5 volume across drives connected to different controllers and buses.
- You cannot include a non-SPARCstorage Array disk within a RAID-5 configuration.
- You cannot add a mirror to a RAID-5 volume.
- Due to the parity calculation complexities given earlier, RAID-5 is much slower on writes than reads. Volumes with more than 20% - 30% writes should probably not be RAID-5 for this reason.
- RAID-5 sequential performance can be good for reads and writes if the stripe width times the number of columns minus 1 exactly matches the I/O size, which means that full stripe writes could be used. Note that even in this case, writes will be slower than reads.

Note – The procedure for setting up a RAID-5 configuration is given in the following sections:

- Using the GUI — Section 4.5 on page 4-35
- Using the CLI — Section 9.4 on page 9-9

1.2.2.5 Volumes

A *volume* can contain from one to eight plexes. An active volume can contain from one to eight plexes. Volume Manager uses *vol##* as the default naming convention for volumes, and *vol##-##* as the default naming convention for plexes in a volume. A volume with one plex contains one copy of the data and would look like the volume shown in Figure 1-17. Note that all the subdisks within a volume must belong to the same disk group.

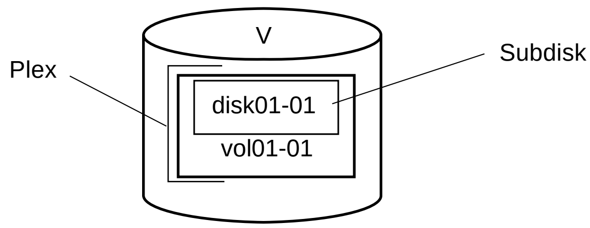

Figure 1-17 Volume with One Plex (Example)

Note that volume vol01 in Figure 1-17 has the following characteristics:

- The volume contains one plex named vol01-01.
- The plex contains one subdisk named disk01-01.

Figure 1-18 shows how a volume would look if it were set up for the simple striped configuration given earlier.

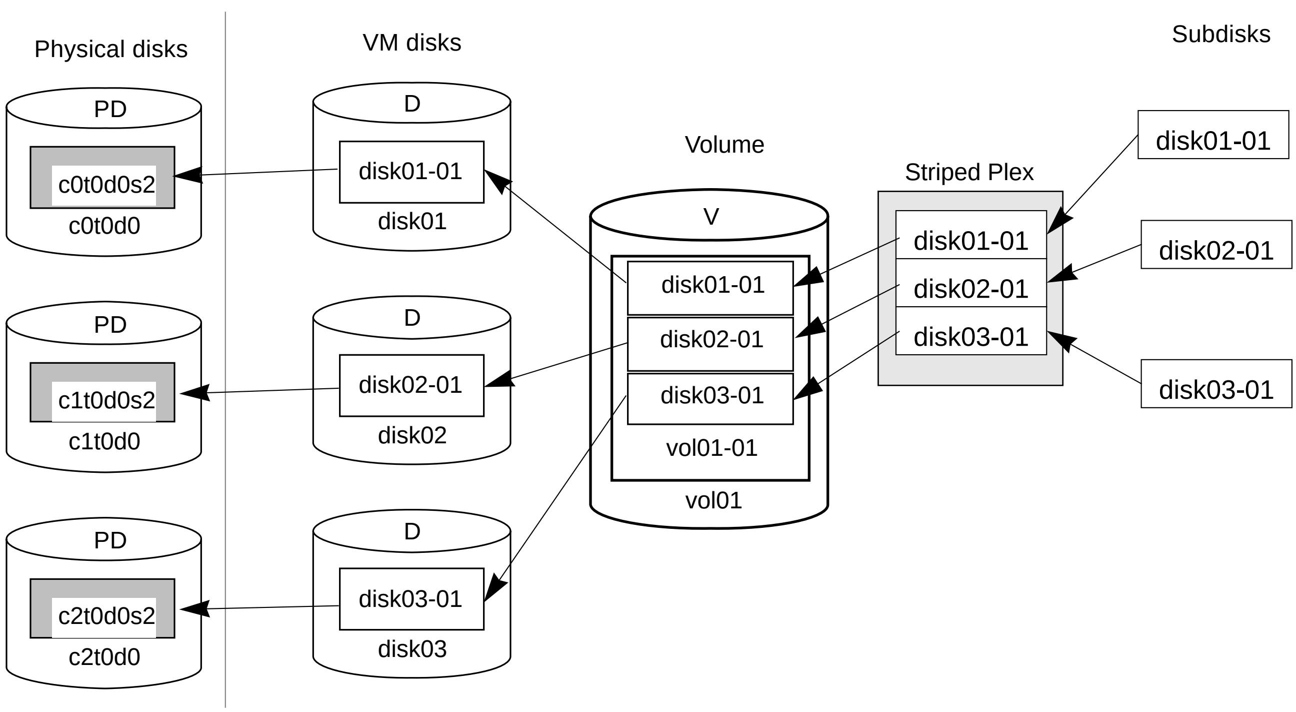

Figure 1-18 Volume in a Striped Configuration (Example)

Volumes can be thought of as virtual disk partitions. If you set up your system without using the disk management features available through the Volume Manager, file systems and database management systems would interact directly with the partitions on the physical disks. If you set up your system using the Volume Manager, however, the file systems and database management systems would actually interact with the volumes, which would then pass the information on to the disk partitions. For each volume, the Volume Manager creates devices in the `/dev` directory. Both raw and block devices are created for each volume.

Volumes must be built from components from a single disk group. Different pieces of a volume cannot span disk groups. This means all components of a striped, RAID-5 or mirrored volume must be contained in the same disk group. This also means that hot spares for a volume must be part of the same disk group as the volume. Take all of these things into consideration when creating a disk group.

For example, a file system is normally mounted on a disk partition device, such as `/dev/dsk/c0t0d0s2`. With the Volume Manager, the file system on a volume named *myvolume* in disk group *mygroup* is mounted on `/dev/vx/<dsk or rdsk>/mygroup/myvolume`. Note that a volume appears to higher level software (such as a Solaris file system or a database management system) as a disk partition.

Note – If you have any scripts that access disks using the path `/dev/dsk/c#t#d#s#` (for example, a script that backs up the data on disks), keep in mind that when a disk is placed under Volume Manager control, the volume that is associated with that disk is listed in the `/dev/vx/<dsk or rdsk>/`*diskgroup*`/`*volume* directory. You would therefore need to change all occurrences of `/dev/dsk/c#t#d#s#` to `/dev/vx/<dsk or rdsk>/`*diskgroup*`/`*volume* in the script so that it will access the volumes assigned to the disks rather than accessing the disks themselves.

Mirroring

Mirroring copies the same data onto two or more separate physical disk drives. This is useful if you want to make sure data is available even if one physical disk fails; the data can be retrieved from the other physical disk that has the copy of the original data. When striping or spanning across a large number of disks, failure of any one of those disks will generally make the entire mirror unusable. The chance of one out of several disks failing is sufficient to make it worthwhile to consider mirroring in order to improve the reliability of a striped or spanned volume. When one disk in a mirrored volume fails, the Volume Manager continues to provide read and write access to the volume, so mirroring is a way to provide continuous data access, even in the event of a disk failure.

If you wanted to mirror data across several physical disks, you would add a second plex to a volume to create a mirror. A volume with two or more plexes contains mirror images of the data and would look like the volume shown in Figure 1-19.

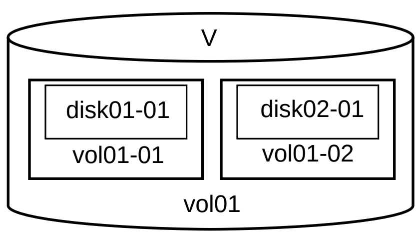

Figure 1-19 Volume with Two Plexes (Example)

Note that volume vol01 in Figure 1-19 has the following characteristics:

- The volume contains two plexes named vol01-01 and vol01-02.
- Each plex contains one subdisk.

Mirroring may also improve system performance. Unlike striping, however, performance gained through the use of mirroring depends on the read / write ratio of the disk accesses. If the system workload is primarily *read-intensive*, then mirroring can improve system performance, because the read can come from either plex in the mirrored volume. If the system workload is primarily *write-intensive* (for example, greater than 30 percent writes), then mirroring can result in somewhat reduced performance, because the write must go to both plexes in the mirrored volume.

Since mirroring will most often be used to protect against loss of data due to drive failures, it may sometimes be necessary to use mirroring for write-intensive workloads. In these instances, mirroring can be combined with striping to deliver both high availability and performance.

To provide optimal performance for different types of mirrored volumes, Volume Manager supports two read policies:

- The round robin read policy, in which read requests to the volume are satisfied in a round-robin manner from all mirrors in the volume
- The preferred read policy, in which read requests are satisfied from one specific mirror (presumably the mirror with the highest performance), unless that mirror has failed, in which case another mirror is accessed.

For example, in the configuration shown in Figure 1-20, the read policy of the volume labeled "Hot Vol" should be set to `preferred read` to the striped mirror labeled "Plex1." In this way, reads going to Plex1 distribute the load across a number of otherwise lightly used disk drives, as opposed to a single disk drive.

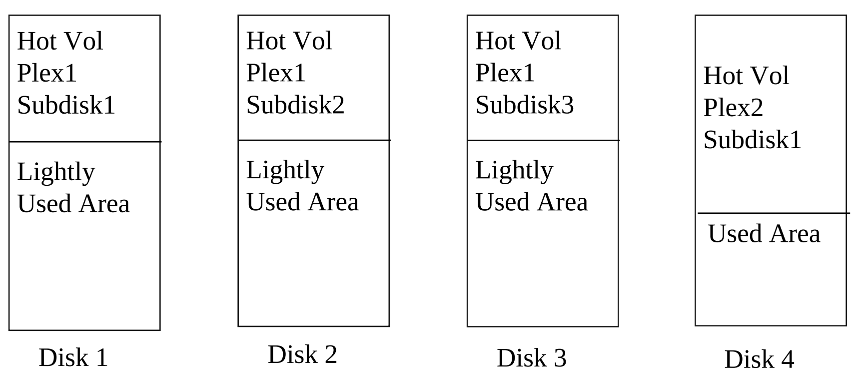

Figure 1-20 Improving System Performance using Mirroring and Striping

To improve performance for read-intensive workloads, up to eight mirrors can be attached to the same volume, although this scenario results in a decrease of effective disk space utilization. Performance can also be improved by striping across half of the available disks to form one mirror and across the other half to form another mirror.

Follow these guidelines when using mirroring:

- Never place subdisks from different plexes of a mirrored volume on the same physical disk. This action compromises the availability benefits of mirroring and significantly degrades performance.
- To provide optimum performance improvements through the use of mirroring, at least 70 percent of the physical I/O operations should be reads. A higher percentage of read operations results in a higher benefit of performance. Mirroring may provide no performance increase or result in a decrease of performance in a write-intensive workload environment.

Note – The Solaris file system implements a file system cache. Since read requests frequently can be satisfied from this cache, the read/write ratio for physical I/O's through the file system can be more biased toward writing than the read/write ratio application level.

- Where feasible, use disks attached to different controllers or busses when mirroring or striping.
- If one mirror exhibits superior performance—due to either being striped or concatenated across multiple disks, or because it is located on a much faster device—then the read policy can be set to prefer the "faster" mirror. By default, a volume with one striped plex should be configured with preferred reading of the striped plex.
- If only one mirror of a mirrored volume is striped, be sure to set the policy of the volume to `preferred read` for the striped mirror. (The default read policy, `select`, does this automatically.)
- If more than one mirror of a mirrored volume is striped, make sure the stripe width is the same for each striped mirror.
- You cannot add a mirror to a RAID 5 volume.

Note – The procedure for setting up a mirrored configuration is given in the following sections:

- Using the GUI — Section 4.3 on page 4-26
- Using the CLI — Section 9.3 on page 9-7

1.2.3 Performance Guidelines

Volume Manager provides flexibility in configuring storage to improve system performance. Two basic strategies are available for optimizing performance:

- Assigning data to physical drives in order to evenly balance the I/O load among the available disk drives
- Identifying the most frequently accessed data and increasing access bandwidth to that data through the use of mirroring, striping and RAID-5

When deciding where to allocate storage, a system administrator typically attempts to balance I/O load among available drives. This approach may not be very effective, however, since a system administrator may not be able to anticipate future usage patterns and cannot split file systems or raw access partitions across drives. For example, if a single disk volume receives most of the disk accesses, placing that volume on another drive will only move the bottleneck to another drive.

Since Volume Manager provides a way for volumes to be split across multiple drives, a finer level of granularity in data placement can be achieved. After measuring actual access patterns, the system administrator can adjust storage placement decisions. Volumes can be reconfigured online after performance patterns have been established or have changed, without adversely impacting volume availability.

1.2.4 Hot-Sparing

Hot-sparing can be used to automatically rebuild mirrored or RAID-5 data when a disk fails. When a disk with mirrored or RAID-5 data fails, Volume Manager looks for a spare disk and places the data from the failed disk on the spare disk. Hot spares are optional. You do not have to configure hot spares, but they provide an extra measure of protection for mirrored or RAID-5 data and can be used to defer physical replacement of a failed drive.

Hot sparing works on the following principles:

- You must manually designate a VM disk as a hot spare.
- The hot spare provides data recovery only for other VM disks in the same disk group.

- The hot spare works only for data on a mirrored or RAID-5 volume; otherwise, the data is lost when a disk fails and only the volume's configuration is maintained on the disk.

When a disk failure is detected, Volume Manager first attempts to correct the object error that brought the failure to its attention. If Volume Manager is unable to correct the error and cannot access the disk, it considers the disk to have failed and sends you a mail message about the failure (for more information, refer to the `vxnotify(1M)` man page).

At the same time, Volume Manager searches for a suitable replacement for the failed disk. The replacement selection is based upon what disks are available and what Volume Manager objects reside on the failed disk. For example, if the disk that failed was part of a striped volume, the replacement disk should preferably be on a different controller than the other disks in that volume.

For example, looking at the disk group in Figure 1-21, the following observations can be made:

- VM disks craig01 and craig02 are part of a mirrored volume (vol01), while VM disk craig03 is part of a simple volume (vol02).
- VM disk craig10 has been designated as a hot spare for the other VM disks in this disk group (shown by the HS at the top of the icon).

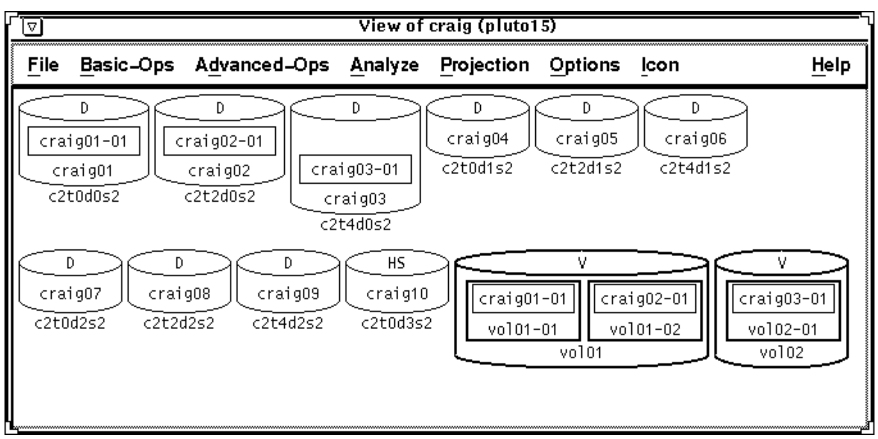

Figure 1-21 Disk Group for Hot Sparing (Example)

Because the VM disks craig01 and craig02 are part of a mirrored volume, if either of the physical disks associated with these VM disks goes bad, then the data from the remaining (good) physical disk is automatically placed on the hot spare VM disk (craig10) and the disk previously known as craig10 will take on the name craig03. However, since the VM disk craig03 is part of a simple volume, the hot spare VM disk craig10 will not recover the data from the volume associated with it (vol02).

Note that disk failures are detected when a read or write failure occurs. Therefore, if a disk or controller fails but not I/O is attempted to the disk or controller, then the Volume Manager will not detect the failure and hot sparing will not go into effect.

Note – If no spare disks large enough to contain all the data from the failed disk are available, Volume Manager notifies you about the failure, but no other automatic action is taken.

Note – The procedure for adding a hot spare is given in the following sections:

- Using the GUI — Section 5.1.3 on page 5-6
- Using the CLI — Section 10.1.5 on page 10-11

1.2.5 Dirty Region Logging

Dirty Region Logging (DRL) keeps track of the regions that have changed due to I/O writes to a mirrored volume. If DRL is not used and a system failure occurs, all mirrors of the volumes must be restored to a consistent state by copying the full contents of an ACTIVE mirror to other mirrors. This process can be lengthy and I/O intensive and it will unnecessarily recover the areas of volumes that are already consistent.

DRL considers a volume to be a set of consecutive regions. It keeps track of volume regions which are being written to. A dirty region log is maintained on a subdisk which contains a status bit for each region of the volume. The log subdisks can be chosen from an even integer between 2 and 10 sectors.

Before writing the data in a write operation to the volume, the regions being written are marked dirty in the log.

If the volume uses DRL, even if the system crashes before writing the data to all plexes of the volume, on system restart the Volume Manager will recover only those regions of the volume which are marked as dirty in the dirty region log.

The dirty bit for a region is not cleared immediately after writing the data to the region. Instead it is left untouched till the corresponding volume region becomes the least recently used.

DRL is an optional property of a mirrored or RAID-5 volume, used to provide a speedy recovery after a system failure. Note that DRL is effective only after a system failure; DRL is of no value if you manually detach and reattach a plex.

Note – DRL adds a small I/O overhead for most access patterns – when data is written to a region that is not already marked as dirty, the DRL subdisk must be updated.

Follow these guidelines when using DRL:

- Make sure that the subdisk that will be used as the log subdisk does not contain any data.
- Logging subdisks must be an even number of blocks from two to ten. If a volume is small, a logging subdisk of 2 blocks may be sufficient. The larger the logging subdisk size, the finer the granularity of a dirty region. For extremely large volumes, it is recommended that you create a large (eight or ten block) logging subdisk.
- In a write-intensive environment, log subdisks can become very busy and are potential performance bottlenecks. In read-intensive applications, DRL adds very little overhead.
- The log subdisk should not be placed on a heavily-used disk, if possible.
- Ideally, the log subdisk should not be placed on a physical disk that contains other parts of the mirrored or RAID-5 volume.

Note – The procedure for creating a Dirty Region Log is given in the following sections:

- Using the GUI — Section 6.11 on page 6-27
- Using the CLI — Section 11.10 on page 11-28

1.3 Volume Manager Rootability

The Volume Manager provides the capability of placing the `root` file system, `swap` device, and `usr` file system under Volume Manager control. Once under Volume Manager control, the `root` and `swap` devices appear as volumes and provide the same characteristics as other Volume Manager volumes. A volume that is configured for use as a swap area becomes the *swapvol* volume, and a volume that contains the root file system becomes the *rootvol* volume.

1.3.1 Boot Device Rules

Following is a list of rules when using Volume Manager on the boot device. These rules are explained in greater detail later on.

You can:

- Place these under Volume Manager control:
  - root filesystem (becomes `rootvol`)
  - swap device (becomes `swapvol`)
  - /usr filesystem
- Mirror these:
  - `rootvol`
  - `swapvol`
  - other parts of root disk (such as `/usr`)
- Stripe or span any mirrors of `rootvol` or `swapvol` if you are only concerned with performance and not with system recovery

You cannot:

- Stripe or span the primary plex on the `rootvol` or `swapvol` device
- Stripe or span any mirrors of `rootvol` or `swapvol` that may be needed for system recovery if the primary plex fails
- Create more than one `rootvol`, `swapvol`, or `usr` volume per host
- Have a primary plex with multiple non-contiguous subdisks on the `rootvol` or `swapvol` volumes

You must:

- Have the rootvol volume in the rootdg disk group
- Have the same size and layout for both disks if you are mirroring the information on the boot device
- Have these on the same disk if you are mirroring the information on the boot device:
  - rootvol
  - swapvol
  - /var partition
  - /usr partition

The primary plex for rootvol and swapvol must be simple; it cannot be spanned or striped. You can, however, add a striped or spanned mirror to the primary plex for rootvol and swapvol, as long as it's for performance reasons. If you are concerned with system recovery, you cannot add a striped or spanned mirror to the primary plex for rootvol and swapvol.

The rootvol and swapvol volumes, as well as other parts of the root disk (for example, /usr) required for a successful boot of the system, can be mirrored. This provides complete redundancy and recovery capability in the event of disk failure. Without the Volume Manager rootability, the loss of the root, swap or usr partition would result in the system being unable to be booted from surviving disks.

By mirroring drives critical to booting, you ensure that no single disk failure will leave your system unusable. Therefore, a suggested configuration would be to use the vxdiskadm command menu to mirror the critical disk onto another available disk. Keep in mind that you must have the same size and layout for both disks if you are mirroring the information on the boot device; you can check this information using the prtvtoc command.

In case of a failure of the disk containing the root partition, reboot the system from the disk containing the root mirror. For more information on mirroring the boot disk and system recovery procedures, see Appendix E, "Recovery."

Note – The procedure for mirroring the root volume is given in the following sections:

- Using the GUI — Section 5.1.8 on page 5-15
- Using the CLI — Section 10.1.9 on page 10-15

1.3.2 Booting with root Volumes

Ordinarily, when the system is booted, the `root` file system and `swap` area need to be available for use very early in the boot procedure, which is long before user processes can be run to load the Volume Manager configuration and start volumes. The `root` and `swap` device configurations must be completed prior to starting the Volume Manager. Starting `vxconfigd` as part of the `init` process is too late to configure volumes for use as a `root` or `swap` device.

To circumvent this restriction, the mirrors of the `rootvol` and `swapvol` volumes can be accessed by the system during startup. During startup, the system sees the `rootvol` and `swapvol` volumes as regular partitions and accesses them using standard partition numbering. Therefore, `rootvol` and the `swapvol` volumes must be created from contiguous disk space that is also mapped by a single partition for each. Due to this restriction, it is not possible to stripe or span `rootvol` and `swapvol` volumes.

1.3.3 Boot-time Volume Restrictions

The `rootvol`, `swapvol`, and `usr` volumes differ from other volumes in that they have very specific restrictions on the configuration of the volumes.

- Only one `rootvol` and `usr` volume can exist per host.
- The `rootvol` volume must reside in the `rootdg` disk group. Although you can create other volumes named `rootvol` in other disk groups, only the volume `rootvol` in `rootdg` can be used to boot the system.
- The `rootvol` volume has a specific minor device number, 0. The `usr` volume does not have a specific minor device number.
- Restricted mirrors of `rootvol`, `var`, and `usr` devices will have “overlay” partitions created for them. An “overlay” partition is one that exactly encompasses the disk space occupied by the restricted mirror. During boot (before the `rootvol`, `var`, and `usr` volumes are fully configured), the default volume configuration uses the overlay partition to access the data on the disk.
- Although it is possible to add a striped mirror to a `rootvol` or `swapvol` device for performance reasons, you cannot stripe the primary plex or any mirrors of `rootvol` that may be needed for system recovery if the primary plex fails.

- `rootvol` and `swapvol` cannot be spanned or contain a primary mirror with multiple non-contiguous subdisks.
- When you mirror parts of the boot disk, the disk being mirrored to must be large enough to hold the data from the original plex or mirroring may not work.
- `rootvol` and `usr` cannot be dirty region logging volumes.

1.4 Using the Graphical User Interface

The following sections give general information on using the Volume Manager's graphical user interface.

1.4.1 File Systems and the GUI

The Volume Manager GUI performs file system operations by executing file system commands directly. With the GUI, file system operations can only be performed by placing the file system on a volume.

Due to the nature of Solaris file systems, the GUI cannot detect whether an unmounted file system already exists on a given volume. Only mounted file systems can be detected by the GUI. Therefore, you must be aware of whether an unmounted file system already exists before performing certain file system operations using the GUI.

With the graphical user interface, there are no icons for file systems. However, the mount point name appears below the volume icon if a mounted file system exists on that volume. If the mount point is very long, only part of it appears under the volume icon; when the pointer moves over the volume or the mount point, the entire mount point becomes visible.

The GUI updates the `/etc/vfstab` file (file system table) when appropriate to maintain consistency.

1.4.2 Using the Mouse with the Graphical User Interface

The graphical user interface enables you to use the mouse (either alone or together with certain keyboard keys) to select, manipulate, or display the properties of icons and their underlying objects.

The following terms are associated with the use of the mouse.

Table 1-1 Terms Associated with Use of Mouse

Term	Definition
Click	Press and release the mouse button in a single, smooth motion.
Double-click	Click the mouse button twice in rapid succession.
Press and Hold	Press and continue to hold down the mouse button.
Point	Move the tip of the pointer onto an item on the screen.
Drag and Drop	Press and continue to hold down the button while moving the mouse and pointer (to drag an object). Release the button when the pointer reaches its destination (to drop the object).
Select	Click the mouse button while the pointer is directly over the item to be selected (such as an icon).

Throughout this manual, the three buttons on the mouse are referred to as the LEFT, MIDDLE, and RIGHT mouse buttons. Table 1-2 lists the mouse and keyboard combinations recognized by the Volume Manager.

Note – This guide assumes that you are using a right-handed mouse. It is possible to remap mouse buttons (by using the `xmodmap` command, for instance). Refer to the man pages for `xmodmap` for more information.

Table 1-2 Mouse and Keyboard Combinations

Action	Result
LEFT	Selects a single icon. Clicking LEFT on an icon causes that icon to be the currently selected one; any previously selected icons are de-selected.
SHIFT-LEFT	Toggles between minimizing or maximizing an icon. A maximized icon displays all of its components in detail. A minimized icon is compressed and its components are concealed.
MIDDLE	Selects either one icon or multiple icons simultaneously. When selecting with the MIDDLE button, any previously selected icons remain selected along with the newly selected one. Clicking MIDDLE on a selected icon de-selects that icon.

Table 1-2 Mouse and Keyboard Combinations (Continued)

Action	Result
SHIFT-MIDDLE	Toggles between starting or stopping projection on the selected icon.
RIGHT	On an icon that is *not* undergoing analysis, displays the properties form for that object. On an icon that is undergoing analysis, displays the analysis statistics form for that object.
SHIFT-RIGHT	Always displays the properties form for the object, regardless of whether analysis is in effect.

1.4.3 Views

Views are special GUI windows that display icons representing all or a subset of the objects currently known to Volume Manager. With views, the user can examine and manipulate different parts of the physical and logical storage systems. Icons can only be added to or removed from views via the GUI.

Each view window title includes the name of the machine on which the GUI session is running. The *root window* provides a view button area containing a button for every view on the system (see Figure 1-22). Views are accessed by clicking the LEFT mouse button on the appropriate view button in the views subwindow.

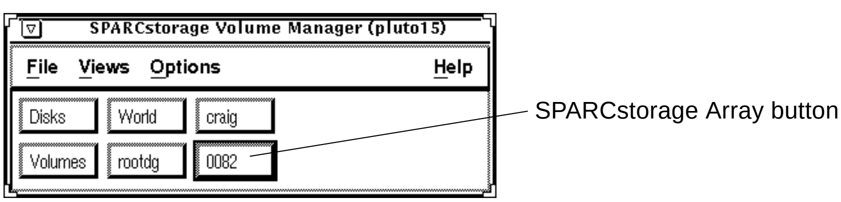

Figure 1-22 Root Window

There are two types of views in the GUI: default and user-created views. Both view types function identically, but certain restrictions are placed on default views. Default views cannot be removed or renamed by the user, as user-created views can.

1.4.3.1 Default Views

The default views provided by the GUI include:

- View of disk group — Displays all objects in a particular disk group (including VM disks, volumes, and associated objects)
- View of SPARCstorage Array — Displays all physical disks within a particular SPARCstorage Array
- View of Disks — Displays all physical disks on the system except disks within a SPARCstorage Array
- View of Volumes — Displays all volumes (as well as plexes and associated subdisks) on the system
- View of World — Displays all objects on the system (including disks, volumes, and associated objects)

Disk groups are represented as view windows rather than icons. The VM disks, volumes, and other objects belonging to a particular disk group are all displayed within its view.

There is always a disk group called `rootdg`, so there is always a `rootdg` disk group view (for more information on `rootdg`, see Section 1.2.2.2, “Disk Groups,” on page 1-8). If additional disk groups are created, the GUI creates views for them.

Note – You should perform operations in the disk group view (rootdg or another disk group view) whenever possible.

Figure 1-23 illustrates a View of rootdg window that does not yet contain any volumes.

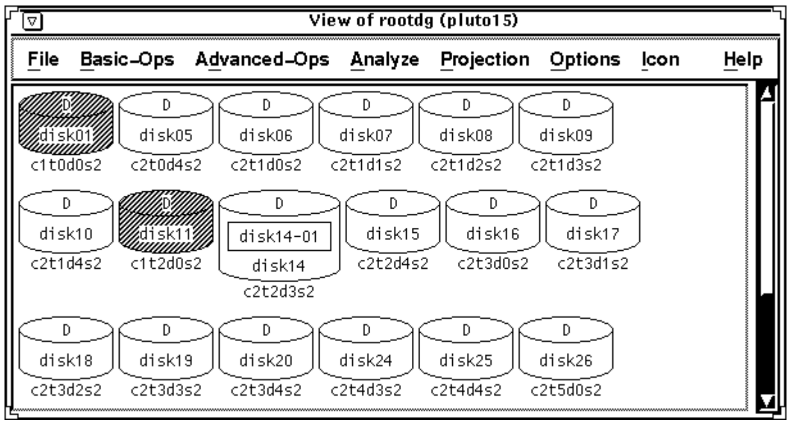

Figure 1-23 View of rootdg

The View of SPARCstorage Array displays all physical disks present in a SPARCstorage Array. There is one View of SPARCstorage Array for every SPARCstorage Array that is connected to the system. This view also displays other hardware components within the SPARCstorage Array, such as the fans, battery, controller and drive trays. Figure 1-24 shows a View of SPARCstorage Array Model 100; note that there is a different view for a SPARCstorage Array Model 200.

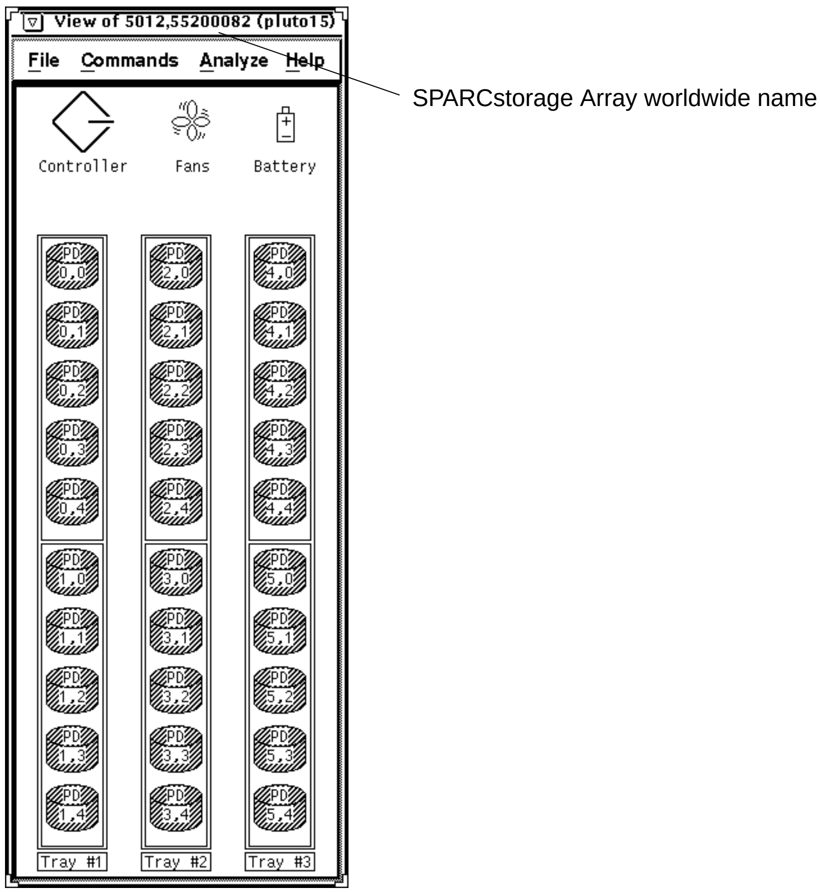

Figure 1-24 View of SPARCstorage Array

In the root window, there is a button for every SPARCstorage Array connected to the system. The four characters on the buttons correspond to the last four characters of the worldwide name of the SPARCstorage Array (see Figure 1-22). Note that the same four characters are displayed on the LCD display on the front of the array. The full worldwide name is displayed in the banner on the View of SPARCstorage Array window (see Figure 1-24).

The View of Disks displays icons for every non-SPARCstorage Array physical disk that has been initialized by Volume Manager. This view is used only for operations on non-SPARCstorage Array disks.

The View of Volumes displays all volumes, regardless of disk group. Plexes and any associated subdisks are also displayed here. This is the iconic representation of virtual disk partitions created by the Volume Manager. Although this view can be used for various operations, it is generally preferable to use a disk group view for most operations.

The View of World essentially combines both the Disks and Volumes views. It also contains any unassociated plexes, which do not appear in most other views. Although this view can be used for various operations, it is generally preferable to use a disk group view or SPARCstorage Array view for most operations.

1.4.3.2 User-created Views

A user-created view is a view window that focuses on a particular part of a physical and/or logical mass storage system, as defined by the system administrator. The system administrator can create views consisting of a selected collection of icons. For example, a user might create a special view to correspond to a physical or logical grouping (such as a view for the accounting department). User-created views enable the user to isolate part of the mass storage subsystem in order to observe or monitor that part of the configuration.

User-created views differ from default views in that they contain *copies* of icons from default views. Operations performed on these icon copies are reflected in the default views that display the affected icon(s). However, icons that appear in user-created views are not always updated whenever those icons are altered in the corresponding default view.

User-created views can be created using the Views pull-down menu from the root window. Once created, icons can be added to a new view window by copying them over from existing views via the Icon menu.

1.4.4 Menus

The GUI provides hierarchical menus called *pull-down menus*, which provide access to various features, operations, forms, and user preference settings. Some of these menus are *cascading menus*, which contain submenus of their own. These menus allow you to select operations or access forms. Help is available on every menu to assist you in understanding the current operation and navigating through the GUI. Menu access varies according to the type and location of the menu.

Each GUI window contains a title bar that identifies the window. Below the title bar is a menu bar, which provides access to the menus available with that window. The menu bar contains the names of each available pull-down menu. One of the characters of each menu name is underlined; this is the mnemonic associated with that particular menu.

Figure 1-25 illustrates both the title bar and menu bar of a typical view window.

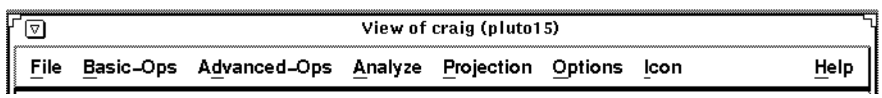

Figure 1-25 Title and Menu Bars

Pull-down menus on the menu bar can be accessed via either the mouse or the keyboard:

- Mouse — Place the pointer on the appropriate menu name in the menu bar and then click the LEFT mouse button.
- Keyboard — Press `Meta` - *`key`*, where *`key`* is the mnemonic (underlined character) associated with the menu name.

Once you access a pull-down menu, any available option in that menu can be selected. Unavailable options are displayed within the menu in greyed-out text. Some menu selections lead to cascading submenus; some lead to forms that can be completed and then applied; some simply execute a specific operation immediately.

Menu selections can be activated either via the mouse or keyboard:

- Mouse — Place the pointer on the appropriate menu option and then click the LEFT mouse button.
- Keyboard — Press the key corresponding to the mnemonic (underlined character) for that particular menu item.

The menus available in the menu bar vary according to the type of window that is being displayed. The following menus are, however, common to most GUI windows:

- File — Typically gives the user the option of closing the window and/or exiting the GUI completely.
- Options — Provides various options that allow the user to specify preferences for how the Volume Manager should operate.
- Help — Provides information relating to the current window, its menus, and related topics.

1.4.5 Forms

The GUI uses forms to present textual information for the user to examine and/or change. These forms provide useful information about existing objects and configurations. Forms generally contain one or more fields that may accept or require user input. Forms contain buttons that can be used to carry out certain actions relating to the form. Error messages appear directly in forms when there is a problem with the form or its input.

There are two basic types of forms:

- General forms tend to appear during operations or setup requests and accept or require user input. General forms are accessed via certain menu selections.
- Properties forms display detailed information about a specific object's current characteristics, some of which can be modified directly through the properties form. Properties forms are accessed by placing the pointer on the chosen icon and then clicking the RIGHT button (unless the icon is undergoing analysis, in which case `Shift`-RIGHT must be used instead).

1.4.5.1 Fields

Some form fields require some input in order to proceed with an operation; if a required field is either blank or incorrect, an error message will result. Other form fields already contain information (such as default values), which you may either alter or accept. Yet other form fields are read-only and therefore cannot be changed; these fields beep when a user attempts to change them.

Form fields can be altered or completed in various ways, depending on the type of field. Some require text to be typed in, while others make use of assorted buttons that can be toggled or selected.

1.4.5.2 Screen Buttons

All forms have screen buttons on the bottom that perform these standard functions:

- Apply — Accept the information on this form, check for errors, and continue with an operation. Pressing the Return key is the equivalent of selecting Apply with a form. For a properties form, the GUI issues the appropriate commands to make the changes.
- Reset — Fill in the fields of the form with its default values. If the form is a properties form, the GUI uses the values that were present when the form was first popped up.
- Cancel — Ignore all changes made on the form and close it. If this was brought up as part of an operation, that operation is canceled.
- Help — Pop up the Help window with information about this form.

Note – Some forms are read-only; only the `Help` and `Cancel` buttons are provided on these forms. On some displays, certain forms may be too long to fit entirely on the screen and the buttons and fields at the bottom of the form may not be visible. If this is the case, move the form to view the desired buttons or fields.

1.4.5.3 Error Messages

Error messages are presented to you when `Apply` is selected with one or more fields incorrect on the form. If this happens, a message is printed at the bottom of the form and you can correct the values for those field(s). If the error cannot be corrected or the operation is no longer desired, select `Cancel`.

1.4.6 Warning Boxes

The GUI uses dialog boxes to present error or warning messages. When a message is displayed in this manner, you must acknowledge it by selecting one of the buttons displayed in the error dialog box before proceeding. Some warning boxes announce that a prerequisite has not been met, so you must acknowledge this by clicking the displayed `Continue` button before reattempting the operation (see Figure 1-26).

Figure 1-26 Warning Box

1.4.7 Help

The GUI provides an extensive Help facility to assist you while navigating through the GUI. Help windows are scattered throughout the various GUI windows, menus, and forms to provide relevant information. Help information varies according to the location and circumstances.

1.4.7.1 Using Help

Depending on the location, you can access Help windows by clicking the LEFT mouse button on the:

- Help menu in a window's menu bar
- Help option in a menu
- Help button at the bottom of a form

Once Help is invoked, a Help window containing information relevant to the current window, menu, form, or operation is displayed. The Help window is equipped with both horizontal and vertical scrollbars, which you can use to scroll through the available Help text.

At the bottom of each Help window is a SEE ALSO area. This area lists similar or related Help topics and operates hypertext-style. To access any of the listed Help topics, click the LEFT button on the appropriate word(s) in the SEE ALSO list; a new Help screen containing information on the selected topic replaces the previous help text.

The Help facility keeps track of the order in which Help topics are visited. You can return to re-read a previous topic. Selecting Previous from the menu bar displays the previous topic. Selecting Next displays the next topic. Figure 1-27 shows a typical Help window.

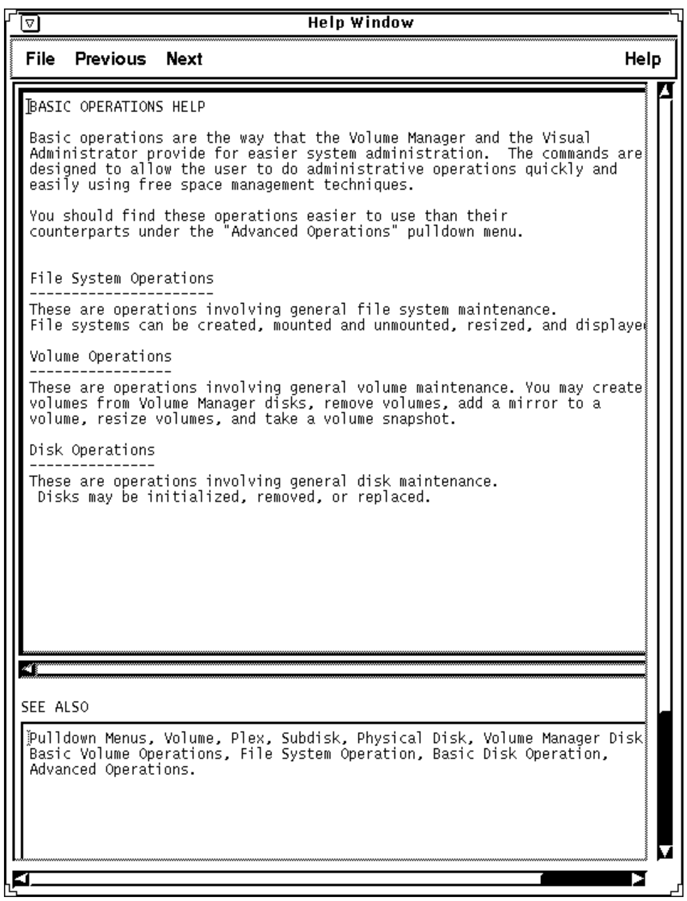

Figure 1-27 Help Window

The Help menu in the menu bar of the Help window provides access to:

- General Help — Accesses general information on the Volume Manager Help facility and how it is used.
- Help Index — Accesses a complete listing of the available Help topics, arranged in logical groupings. Once a topic is identified from this list, you can directly access that topic from the SEE ALSO section of this Help window, which lists all topics alphabetically. Clicking LEFT directly over the topic name in the SEE ALSO section causes that topic to appear in the Help window.

The Help window can be closed by selecting the Close option from the File menu. The record of help topics visited is retained.

1.4.8 Color and Bitmap Patterns Used in the Graphical User Interface

Depending on the type of monitor you use, the GUI employs color or bitmap patterns to indicate:

- State of an icon
- Activity level of an icon
- Relationships between icons
- Failure of an operation

By default, standard X Window System bitmaps are used to create these patterns. Table 1-3 gives the values for the default colors and bitmap patterns associated with the GUI icons under different conditions.

Table 1-3 Default Colors and Patterns

Situation	Color	Bitmap Pattern
Selected icon	green	gray3
Disabled icon	grey	stripe4
Alarmed icon	red	gray1
Free subdisk icon	yellow	root_weave
Projection	yellow	root_weave
Analysis: low	green	cross_weave
Analysis: medium	yellow	root_weave
Analysis: high	red	wide_weave

Note – This guide assumes that the default colors or bitmaps are being used. However, colors or bitmaps (as well as other interface preferences) can be redefined in your `.Xdefaults` file or on the command line interface. Refer to Appendix B, “GUI-Related X Resources,” for information on resetting these defaults.

It is possible for a single icon to be in multiple states represented by different colors or patterns at once. For example, a given icon may be both selected and under projection at the same time. In such cases, the GUI reflects the color or pattern that represents the highest priority. The following is the priority list for possible icon states, starting with the highest priority:

1. Busy
2. Error
3. Selected
4. Projected
5. Analyzed
6. Enabled

An icon that is in the busy state (highest priority) cannot allow any mouse or keyboard input. The text within a busy icon is greyed out to indicate that it is inaccessible.

Initializing the Software 2

This chapter gives information on initializing the software on the SPARCstorage Array.

Once you have installed the software, you will have the device drivers, libraries, and the command line interface installed on your system. The device drivers, libraries, and command line interface are required components for the SPARCstorage Array subsystem; you will not be able to operate a SPARCstorage Array without them.

The SPARCstorage Volume Manager is not a required component for the SPARCstorage Array subsystem. However, the Volume Manager gives you many useful disk management and user interface features, including:

- Logical volume management
- Support for disk concatenation, striping, and mirroring
- Online storage reorganization and load balancing
- Visual performance displays
- Graphical user interface (GUI)
- Command line interface (CLI) to the Volume Manager

Because of the disk management capabilities provided by Volume Manager, you may want to use the Volume Manager as a part of the overall SPARCstorage Array subsystem. However, if you do not want to use the Volume Manager, you can use different disk management software (such as Online: DiskSuite™), or use the SPARCstorage Array without any kind of disk management software and treat it as a collection of independent SCSI disks.

Note – You can use either Volume Manager or some other disk management software (such as Online: DiskSuite) on disks in your SPARCstorage Array; however, you cannot run *both* types of disk management software on the same disks at the same time.

If you decide that you want to use the Volume Manager, follow the instructions in the remaining sections to initialize the Volume Manager. If you decide that you do *not* want to use the Volume Manager, you can still perform the following procedures, since they are part of the standard SPARCstorage Array software:

- Reserving and Releasing Drives — page 8-2
- Setting Fast Writes — page 8-5

Note that you can perform these procedures only from the CLI if you do not plan to use the Volume Manager software; the GUI is part of the Volume Manager software, so if you want to perform these or any other procedures using the GUI, you must use the Volume Manager software.

2.1 Preparing to Initialize the Volume Manager

Note – The initialization procedures for the Volume Manager may be altered slightly. Read the *SPARCstorage Array Product Note* (801-7838-xx) before proceeding with this section for more information.

Most of the commands used in the installation are in the `/sbin` or `/usr/sbin` directories. You should add both of these directories to your `PATH` environment variable. For example, using the Bourne Shell (`sh` or `ksh`), enter:

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

If you are using a C Shell (`csh` or `tcsh`), enter:

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

Some of the installation processes require repeated reboots. You are asked for permission to reboot when the reboots are needed. If you choose not to reboot, the install will not complete. After the first reboot, the installation process takes control of the system until the installation is complete.

Before installing the Volume Manager, make sure you know what each of your disks contains. During the installation process, you will have to decide the following:

- Do you wish to place all or just some of the disks on the system and/or controller under Volume Manager control?
- Do you wish to place the system's root disk under Volume Management control?
- When you place data under Volume Manager control, do you wish to preserve data in any or all existing file systems and partitions by encapsulation?
- For each disk, do you want to choose whether to encapsulate, initialize or leave the disk alone (Custom Installation)? Alternatively, do you want to either encapsulate or initialize all disks on a controller (Quick Installation)?

Any disk other than the disk containing `root` and `swap` that is to be managed by Volume Manager must have two free partitions and a small amount of free space (1024 sectors). This free space should be at the beginning of the disk and should not belong to a partition. This space is used for storing disk group configurations and a disk label that ensures that the Volume Manager can identify the disk, even if it is moved to a different address or controller.

The Volume Manager allocates approximately 1024 sectors (512K) from each disk for the disk group configurations and the label. This space is sufficient to handle normal disk group configurations for up to 100 disks.

Note – Although it is not recommended, Volume Manager does support the use of "unlabeled" disks after installation.

The boot drive (the root disk) is a special case; if no other space is available, the Volume Manager will attempt to allocate space usually reserved for the swap partition. This process is known as “Swap Relocation” and will happen automatically during installation if necessary.

2.2 Initializing the Volume Manager

Follow the instructions in this section *only* if you are bringing disks under Volume Manager control for the first time. If you upgraded your system from Version 1.3 to 2.0 of the Volume Manager software, then you do not have to go through the following procedure; you can immediately begin to use the Volume Manager CLI or GUI as you did when you were running Version 1.3.

Note – If you halt or reboot your machine while you are still running through the `vxinstall` script, the system will go back to `vxinstall` during the reboot process. When you are finished with the `vxinstall` script, the system will complete the boot-up process.

---

1. **Determine if you want to exclude any disks from Volume Manager control.**
   If you want to exclude one or more disks from Volume Manager control, follow this procedure before entering the `vxinstall` command. (Note that at least one disk must be placed under Volume Manager control using either Quick or Custom Installation, so do not exclude *all* the disks connected to your system.)

Note – You *must* complete the entire `vxinstall` process for the installation procedure to work; if you quit out of `vxinstall` before you have installed all the disks listed, then you will have to run through the entire `vxinstall` process again from the beginning or the software will not function properly. If you do not want to go through the `vxinstall` process for certain disks on your system, you can exclude those disks using the following procedure and bring them under Volume Manager control at a later time.

---

   a. **Using a text editor, create a file called** `disks.exclude` **in the** `/etc/vx` **directory.**

b. **Enter the disks that you want to exclude into the `/etc/vx/disks.exclude` file in the `c`*n*`t`*n*`d`*n* format, with each entry on a separate line.**
Remember that you cannot enter all the disks into the `disks.exclude` file or the Volume Manager installation script will fail.

Following is a sample `disks.exclude` file:

```
c0t0d1
c0t0d2
c1t0d1
```

c. **Save and quit the `disks.exclude` file.**
When you enter the `vxinstall` command, you should see a series of messages telling you that the disks listed in the `/etc/vx/disks.exclude` file are being excluded. If you want to bring some or all of those disks under Volume Manager control after you have completed the installation procedure, follow the instructions in Section 3.3 on page 3-13 to use the GUI or in Section 8.2 on page 8-6 to use the CLI to bring the disks under Volume Manager control.

2. **Become superuser.**

3. **Enter the following command at the prompt**:

```
# vxinstall
```

Note – If a SPARCstorage Array exists on a system where the Volume Manager software has been successfully installed and the array is disconnected from the system, Volume Manager will continue to work. However, if no SPARCstorage Array is connected to the system for 14 days, the Volume Manager will stop working.

You can use concatenation, mirroring, striping and RAID-5 on data on all the physical disks that reside in the SPARCstorage Array. You can also concatenate and mirror data on physical disks that reside outside the SPARCstorage Array, but you cannot use striping or RAID-5 on data on disks that reside outside the SPARCstorage Array.

The `vxinstall` program lists all the controllers that it finds on the system:

```
Generating list of attached controllers....

Volume Manager Installation
Menu: VolumeManager/Install

  The Volume Manager names disks on your system using the
  controller and disk number of the disk, substituting them into
  the following pattern:

    c<controller>t<disk>d<disk>

  Some examples would be:

    c0t0d0 - first controller, first target, first disk
    c1t0d0 - second controller, first target, first disk
    c1t1d0 - second controller, second target, first disk

  The Volume Manager has detected the following controllers on
  your system:

    c0: io-unit@f,e0200000/sbi@0,0/dma@0,81000/esp@0,80000
    c1:unit@f,e0200000/sbi@0,0/SUNW,soc@3,0/SUNW,pln@b0005012,55200063

Hit RETURN to continue.
```

Press Return to continue the installation process.

`vxinstall` next displays a brief introduction to the installation process:

```
Volume Manager Installation
Menu: VolumeManager/Install

  You will now be asked if you wish to use Quick Installation or
  Custom Installation. Custom Installation allows you to select
  how the volume manager will handle the installation of each disk
  attached to your system.

  Quick Installation examines each disk attached to your system
  and attempts to create volumes to cover all disk partitions that
  might be used for file systems or for other similar purposes.

  If you do not wish to use some disks with the Volume Manager,
  or if you wish to reinitialize some disks, use the Custom
  Installation option. Otherwise, we suggest that you use the
  Quick Installation option.

Hit RETURN to continue.
```

Press Return to continue the installation process. `vxinstall` then displays a menu of installation options:

```
Volume Manager Installation Options
Menu: VolumeManager/Install

 1  Quick Installation
 2  Custom Installation

 ?  Display help about menu
 ?? Display help about the menuing system
 q  Exit from menus

Select an operation to perform:
```

4. **Select the menu option for the operation you want performed.**
   Table 2-1 gives the different software installation options. Keep in mind that the option you choose will be performed on the disks on *all* the controllers listed earlier; you cannot choose Quick Installation for the disks on one controller and Custom Installation for the disks on a different controller.

Table 2-1 Software Installation Options

Option	**Description**
Quick Installation	Uses default options for installation. If you want to go through the Quick Installation procedure, go to "Quick Installation" on this page.
Custom Installation	Allows you to control which physical disks will be brought under Volume Manager control. If you want to go through the Custom Installation procedure, go to "Custom Installation" on page 2-13.
?	Displays a help file describing the current operation or menu choices.
??	Displays general information about using `vxinstall`.
q	Exits from the current operation or from the `vxinstall` program.

2.2.1 *Quick Installation*

The Quick Installation option is easier to use. This option examines all disks connected to the system (except those placed in the `disks.exclude` file), encapsulates existing partitions (adds these partitions to the Volume Manager's control, leaving them intact and maintaining the integrity of the data), and initializes disks that do not have existing partitions.

The `/etc/vfstab` file is updated to ensure that file systems previously mounted on disk partitions will be mounted on volumes instead.

Note – Quick Installation will bring every disk attached to your system under Volume Manager control unless you have added those disks to the `disks.exclude` file as described earlier in this procedure. If you have several disks on your system that you do not want placed under Volume Manager control, either add them to the `disks.exclude` file or choose the Custom Installation option, which allows you to leave certain disks alone.

1. **Enter `1` (Quick Installation) at the `vxinstall` menu.**

   `vxinstall` will first find the boot disk for your system and ask you if you want to encapsulate it:

```
Volume Manager Quick Installation
Menu: Volume Manager/Install/Quick Install

  The c2t0d0 disk is your Boot Disk. You can not add it as a new
  disk. If you encapsulate it, you will make your root filesystem
  and other system areas on the Boot Disk into volumes. This is
  required if you wish to mirror your root filesystem or system
  swap area.

Encapsulate Boot Disk [y,n,q,?](default: n)
```

   - If you enter `n` or press Return to accept the default, your root and other file systems will be left alone.
   - If you enter `y`, `vxinstall` encapsulates your root file system as a volume, along with your swap device and all other disk partitions found on your boot disk. `/usr`, `/opt`, `/var`, and any other file systems on your boot disk are also encapsulated.

   `vxinstall` will then list all the disks on the first controller:

```
Volume Manager Quick Installation
Menu: VolumeManager/Install/QuickInstall/c1
Generating list of attached disks on c1....

  The Volume Manager has detected the following disks on
  controller c1:

    c1t0d0
    c1t0d1
    c1t0d2
    c1t0d3

Hit RETURN to continue.
```

   If any disks are listed in the `/etc/vx/disks.exclude` file, they will be listed here separately as excluded disks.

Press Return to continue the installation process.

2. Determine if you want to initialize all the disks on the controller:

```
Volume Manager Quick Installation For Controller c1
Menu: VolumeManager/Install/QuickInstall/c1

Initialize all disks on this controller? (destroys data on these
disks) [y,n,q,?] (default: n)
```

- If you enter `n` or press Return to accept the default, `vxinstall` assumes that you want to encapsulate all the disks on the controller and displays the following:

```
  Volume Manager will now try to encapsulate all the disk on this
  controller. Disks not having valid partitions will be
  initialized.
Hit RETURN to continue.
```

  Press Return to continue the installation process. Go to Step 3.

- If you enter `y`, you will be asked to verify your choice:

```
Are you sure (destroys data on these disks)
[y,n,q,?] (default: n)
```

Caution – Be careful when using this option as it may result in a loss of data.

Enter `n` if you don't want to initialize the disks or enter `y` to verify your choice. If you enter `y`, you will then see the following message:

```
  Volume Manager will now initialize all the disk on this
  controller including disks having valid partitions.
Hit RETURN to continue.
```

Press Return to continue the installation process.

3. **Determine if you want to use the default names for the disks on the controller:**

```
Volume Manager Quick Installation
Menu: VolumeManager/Install/QuickInstall/c1/Encap

Use default disk names for these disks ? [y,n,q,?] (default: y)
```

- If you enter `y` or press Return to accept the default, `vxinstall` will assign the default disk names for each disk:

```
  The c1t0d0 disk will be given disk name disk01
  The c1t0d1 disk will be given disk name disk02
  The c1t0d2 disk will be given disk name disk03
  The c1t0d3 disk will be given disk name disk04

Hit RETURN to continue.
```

  If encapsulation was selected, this listing also includes the following message for disks that can be encapsulated:

```
The c1t0d0 disk has been configured for encapsulation.
```

  and the following message for empty disks that will be initialized instead of encapsulated:

```
  The c3t0d2 disk appears to be empty. Adding as a new disk.
```

  Press Return to continue the installation process. Go to Step 4.
- If you enter `n`, you will be asked to enter the disk name of your choice:

```
Enter disk name for c1t0d0 [<name>,q,?](default: disk01)

Hit RETURN to continue.
```

Enter the disk name for each disk and press Return after each entry. When you have named each of the disks on the controller, press Return again to continue the installation process.

4. Repeat Step 2 and Step 3 for each controller on your system.
When you have completed the `vxinstall` procedure for all the controllers on your system, `vxinstall` displays a summary of the disks you have designated for initialization (`New Disk`) or encapsulation (`Encapsulate`) on each controller:

```
Volume Manager Quick Installation
Menu: Volume Manager/Install/Quick Install

  The following is a summary of your choices.

    c1t0d0  New Disk
    c1t0d1  New Disk
    c1t0d2  New Disk
    c1t0d3  New Disk
    c1t0d4  New Disk
    c3t0d0  Encapsulate
    c3t0d1  Encapsulate

Is this correct [y,n,q,?] (default: y)
```

- If you enter `y` or press Return to accept the default, `vxinstall` will encapsulate and initialize the disks listed.
- If you enter `n`, you will be asked to enter the disks that you want to exclude:

```
Enter disk to be removed from your choices. Hit return when done.
[<name>,q,?]
```

Enter the disk name(s) and press Return after each. When you have entered all the disks that you want to remove, press Return again to continue the installation process. You will be asked if the modified list of disks is correct; enter `y` if the list is correct or `n` if the list is incorrect.

5. **Reboot your system to allow the installation to take effect.**
   If one of the disks that you added using `vxinstall` was the root disk, you will be told that you need to shutdown the system after you have installed all the disk drives:

```
  The system now must be shut down and rebooted in order to
  continue the reconfiguration.

Shutdown and reboot now [y,n,q,?] (default: n)
```

   Answer `y` to this question to allow the installation to take effect. If you do not reboot after the installation, you may encounter problems with the Volume Manager software.

Caution – Do not make any changes to your disk and file system configurations before shutting down and rebooting your system.

2.2.2 Custom Installation

Custom Installation allows you to control which disks are added to Volume Manager control and how they are added. This is important if you do not want all your disks under Volume Manager control or if you want to place some of the physical disks in a disk group other than the default rootdg disk group.

Note – Disks that are not added by `vxinstall` for use with the Volume Manager can be added to the system later using either the GUI or the CLI. However, you must add at least one disk with `vxinstall` before you can add disks using other utilities or interfaces.

1. **Enter 2 (Custom Installation) at the `vxinstall` menu.**
   `vxinstall` will first find the boot disk for your system and ask you if you want to encapsulate it:

```
Volume Manager Custom Installation
Menu: VolumeManager/Install/Custom

  The c2t0d0 disk is your Boot Disk.  You can not add it as a new
  disk. If you encapsulate it, you will make your root filesystem
  and other system areas on the Boot Disk into volumes. This is
  required if you wish to mirror your root filesystem or system
  swap area.

Encapsulate Boot Disk [y,n,q,?](default: n)
```

   - If you enter `n`, your root and other file systems will be left alone.
   - If you enter `y`, `vxinstall` encapsulates your root file system as a volume, along with your swap device and all other disk partitions found on your boot disk. `/usr`, `/opt`, `/var`, and any other file systems on your boot disk are also encapsulated.

   `vxinstall` will then list all the disks on the first controller:

```
Volume Manager Custom Installation
Menu: VolumeManager/Install/Custom/c1
Generating listof attached disks on c1....


  The Volume Manager has detected the following disks on
  controller c1:

    c1t0d0
    c1t0d1
    c1t0d2
    c1t0d3

Hit RETURN to continue.
```

   If any disks are listed in the `/etc/vx/disks.exclude` file, they will be listed here separately as excluded disks.

Press Return to continue the installation process. `vxinstall` next gives you a list of Custom Installation options:

```
Installation options for controller c1
Menu: VolumeManager/Install/Custom/c1

 1  Install all disks as pre-existing disks. (encapsulate)
 2  Install all disks as new disks.(discards data on disks!)
 3  Install one disk at a time.
 4  Leave these disks alone.

 ?  Display help about menu
 ?? Display help about the menuing system
 q  Exit from menus

Select an operation to perform:
```

2. Enter an installation option at the prompt.
Table 2-2 gives an explanation for each Custom Installation option. Note that you can change options for each controller, so you could choose Option 1 for the disks on the first controller and Option 2 for the disks on the second controller.

Note – You must allow Volume Manager to install at least one disk during the installation process or Volume Manager will not be able to complete the installation.

Table 2-2 Custom Installation Options

Option	Description
Option 1— Install all disks as pre-existing disks.	Volumes will be created to encapsulate the partitions on all the disks on the controller. The `/etc/vfstab` file will be updated to ensure that file systems previously mounted on the disk partitions will be mounted as volumes instead.
Option 2 — Install all disks as new disks.	All the disks on this controller will be re-initialized. This destroys all data on the disks and makes the disks available as free space for allocating new volumes, or mirrors of existing volumes. Be careful when choosing this option as it may result in a loss of data. If you are running `vxinstall` on new disks that have never been placed under Volume Manager control before, choose this option so that the disks will be repartitioned according to the Volume Manager's specifications.
Option 3 — Install one disk at a time.	This will allow you to go disk-by-disk and either install the disk as a pre-existing disk, install it as a new disk, or leave it alone.
Option 4 — Leave these disks alone	No changes will be made to the disks. If there are applications that use these disks that you do not wish to upgrade to use the Volume Manager, you can use this option to ensure that your applications can continue to use these disks without modification.

- If you choose Option 1, you will be asked if you want to use the default names for the disks on the controller. Enter `y` or press Return to accept the default if you want to accept the default names, or press `n` if you want to enter the disk name of your choice for every disk on the controller.

  Note that if you choose Option 1 for disks that have never been brought under Volume Manager control before, they may not be formatted based on the Volume Manager specifications. If that is the case, you will see an error message similar to the following:

```
It is not possible to encapsulate c0t2d0, for the following reason:
    <vxvm:vxslicer: ERROR: Unsupported disk layout.>
```

If you see this message, you can either add the disks on as new disks (Option 2), which will destroy all the data on the disks, or you can tell Volume Manager to leave those disks alone (Option 4), which would mean that these disks will not be brought under Volume Manager control.

- If you choose Option 2, you will be asked if you want to use the default names for the disks on the controller. Enter `y` or press Return to accept the default if you want to accept the default names, or press `n` if you want to enter the disk name of your choice for every disk on the controller. Keep in mind that this option will destroy all data on the disks.
- If you choose Option 3, you will see a screen similar to the following for every disk on the controller:

```
Installation options for disk c3t0d0
Menu: VolumeManager/Install/Custom/c3/c3t0d0

 1  Install as a pre-existing disk. (encapsulate)
 2  Install as a new disk. (discards data on disk!)
 3  Leave this disk alone.

 ?  Display help about menu
 ?? Display help about the menuing system
 q  Exit from menus

Select an operation to perform:
```

These options are similar to those given in Table 2-2. Choose the installation option that you want performed for each disk on the controller and answer the necessary questions.

- If you choose Option 4, `vxinstall` will leave all the disks on the controller alone. No changes will be made to the disks and they will not be placed under Volume Manager control. If there are applications that use these disks that you do not wish to upgrade to use the Volume Manager, you can use this option to ensure that your applications can continue to use the disk without modification.

Note – If you choose to leave all disks on all controllers alone, `vxinstall` does not complete the installation of the Volume Manager or ask you for a reboot.

---

3. Repeat Step 2 for each controller on your system.

When you have completed the `vxinstall` procedure for all the controllers on your system, you will see a message asking you to verify your choices:

```
Volume Manager Custom Installation
Menu: VolumeManager/Install/Custom

  The following is a summary of your choices.

    c1t0d0  New Disk
    c1t0d1  New Disk
    c1t0d2  New Disk
    c1t0d3  New Disk
    c1t0d4  New Disk
    c3t0d0  Encapsulate
    c3t0d1  Encapsulate

Is this correct [y,n,q,?] (default: y)
```

- If you enter `y` or press Return to accept the default, `vxinstall` will encapsulate and initialize the disks listed.
- If you enter `n`, you will be asked to enter the disks that you want to exclude:

```
Enter disk to be removed from your choices. Hit return when done.
[<name>,q,?]
```

Enter the disk name(s) and press Return after each. When you have entered all the disks that you want to remove, press Return again to continue the installation process. You will be asked if the modified list of disks is correct; enter `y` if the list is correct or `n` if the list is incorrect.

4. **Reboot your system to allow the installation to take effect.**
If one of the disks that you added using `vxinstall` was the root disk, you will be told that you need to shutdown the system after you have installed all the disk drives:

```
 The system now must be shut down and rebooted in order to
 continue the reconfiguration.

Shutdown and reboot now [y,n,q,?] (default: n)
```

Answer `y` to this question to allow the installation to take effect. If you do not reboot after the installation, you may encounter problems with the Volume Manager software.

Caution – Do not make any changes to your disk and file system configurations before shutting down and rebooting your system.

2.2.3 Using `vxdisk` for Special Encapsulations

In some cases, you may want to encapsulate a disk that does not have any space that can be used for the Volume Manager private region partition. The normal disk encapsulation procedure using `vxencap` [see `vxencap(1M)`] requires that some space be available at the beginning or end of the disk for storing Volume Manager identification and configuration information.

The `vxdisk` utility can be used to encapsulate disks that do not have available space. This is done using special types of disk devices, called `nopriv` devices, that do not have private regions. To use this, create a partition on the disk device that maps all parts of the disk that you want to be able to access, then add the partition device for that partition with the command:

```
vxdisk define partition-device type=nopriv
```

Here, *partition-device* is the basename of the device in the `/dev/dsk` directory. For example, to use partition 3 of disk device `c0t4d0`, use the command:

```
vxdisk define c0t4d0s3 type=nopriv
```

To create volumes for other partitions on the disk drive, add the device to a disk group, figure out where those partitions reside within the encapsulation partition, then use `vxassist` to create a volume with that offset and length.

A major drawback with using these special encapsulation partition devices is that the Volume Manager cannot track changes in the address or controller of the disk. Normally, the Volume Manager uses identifying information stored in the private region on the physical disk to track changes in the location of a physical disk. Since `nopriv` devices do not have private regions and thus have no identifying information stored on the physical disk, this cannot occur.

The best use of special encapsulation partition devices is to encapsulate a disk so that the Volume Manager can be used to move space off of the disk. When space is made available at the beginning or end of the disk, the special partition device can be removed and the disk can then be encapsulated as a standard disk device.

A disk group cannot be formed entirely from `nopriv` devices. This is because `nopriv` devices do not provide space for storing disk group configuration information. Configuration information must be stored on at least one disk in the disk group.

2.3 Volume Manager Daemons

Two daemons must be running in order for the Volume Manager to work properly:

- vxconfigd
- vxiod

2.3.1 Volume Daemon

The volume daemon (`vxconfigd`) is responsible for maintaining a system's configuration in the kernel and on disk. `vxconfigd` must be running before normal operations can be performed.

2.3.1.1 *Starting the Volume Daemon*

`vxconfigd` is started by startup scripts during the boot procedure.

1. To determine whether the volume daemon is enabled, enter the following:

```
vxdctl mode
```

If `vxconfigd` is both running and enabled, the following message appears:

```
mode: enabled
```

If `vxconfigd` is running, but not enabled, the following message appears:

```
mode: disabled
```

2. To enable the volume daemon, enter the following:

```
vxdctl enable
```

If `vxconfigd` is not running, the following message appears:

```
mode: not-running
```

If the latter message appears, start `vxconfigd` as follows:

```
vxconfigd
```

Once started, `vxconfigd` automatically becomes a background process. For more information on the `vxconfigd` daemon, refer to the `vxconfigd`(1M) manual pages.

2.3.2 Volume Extended I/O Daemon

The `vxiod` daemon serves two purposes:

- It allows for some extended I/O operations without blocking calling processes
- It allows the virtual disk driver to schedule writes to volumes that have dirty region logging enabled

If there are volumes with dirty region logging enabled, then there will be multiple `vxiod` processes running on the system. Volume log I/O vxiod daemons are started by `vxconfigd` and are killed by the kernel when they are no longer needed.

For more detailed information about `vxiod`, refer to the `vxiod` (1M) manual page.

2.3.2.1 Starting the Volume Error Daemon

`vxiod` daemons are started at system boot time. There are typically two `vxiod` daemons running at all times. Rebooting after your initial installation should start `vxiod`.

Verify that `vxiod` is running by typing:

```
vxiod
```

Since `vxiod` is a kernel thread and is not visible to users via `ps`, this is the only way to see if any `vxiod`s are running.

If any `vxiod` daemons are running, the following should be displayed:

```
2 volume I/O daemons running
```

where `2` is the number of `vxiod` daemons currently running.

If no vxiod daemons are currently running, start some by entering the command:

```
vxiod set 2
```

where 2 may be substituted by the desired number of vxiod daemons.

2.4 Removing the Volume Manager

The following procedures can be used to remove the Volume Manager.

If root, swap, usr, or var is a volume under Volume Manager control, perform the following steps:

1. **Ensure that rootvol, swapvol, usr, and var volumes have only one associated plex each.**
   The plex must be contiguous, non-striped, non-spanned, and non-sparse. This information can be determined by issuing the following command:

```
# vxprint -ht rootvol swapvol usr var
```

   If any of these volumes have more than one plex associated, remove the unnecessary ones. Unnecessary plexes can be removed using the command:

```
# vxplex -o rm dis plex-name
```

   The plexes remaining for rootvol, swapvol, usr and var should have been created using vxrootmir since this guarantees that the underlying subdisks will start on cylinder boundaries and that partitions will be created for them.

2. **Run** `vxunroot`:

```
# /etc/vx/bin/vxunroot
```

This will change the volume entries in `/etc/vfstab` to the underlying disk partitions for `rootvol`, `swapvol`, `usr`, and `var`. It will also modify `/etc/system` and ask for a reboot so that disk partitions are mounted instead of volumes for `root`, `swap`, `usr`, and `var`.

Once the `root`, `swap`, `usr`, and `var` volumes have been taken care of, all remaining volumes need to be moved to disk partitions. This can be done in one of three ways:

- Back up the system fully onto tape and then recover from it.
- Back up each file system individually and then recover them all after creating new file systems on disk partitions.
- Incrementally move volumes onto disk partitions using the following steps:

1. **Evacuate as many disks as possible using `vxdiskadm`, the GUI, or the `vxevac` script.**
   Evacuation moves subdisks from the specified disks to target disks. The evacuated disks will provide the initial free disk space for volumes to be moved to disk partitions.

2. **Remove the evacuated disks from Volume Manager control using:**

```
# vxdg rmdisk diskname
# vxdisk rm devname
```

3. **Decide which volume to move first. If the volume to be moved is mounted, unmount it.**

4. **If the volume is being used as a raw partition for database applications, make sure that the application is not updating the volume and that data on the volume is synced.**

5. **Create a partition on free disk space of the same size as the volume.**
   If there is not enough free space for the partition, a new disk must be added to the system at least for the first volume which is removed. Subsequent volumes can use the free space generated by the removal of this volume.

6. `dd` **the data on the volume onto the newly-created disk partition using a command like the following:**

```
# dd if=/dev/vx/dsk/lhome of=/dev/dsk/c2t2d2s7
```

where `c2t2d2` is the disk outside of Volume Manager and `s7` is the newly-created partition.

7. **Replace the entry for that volume (if present) in** `/etc/vfstab` **with an entry for the newly-created partition.**

8. **Mount the disk partition if the corresponding volume was previously mounted.**

9. **Remove the volume from the Volume Manager using the command:**

```
# vxedit -rf rm volume_name
```

10. **Remove any disks that have become free (have no subdisks defined on them) by removing volumes from Volume Manager control.**
To check if there are still some subdisks remaining on a particular disk, use the command:

```
# vxprint -F "%sd num" diskname
```

If the output is not 0, there are still some subdisks on this disk which will get removed subsequently. If the output is 0, remove the disk from Volume Manager control as follows:

```
# vxdg rmdisk diskname
# vxdisk rm devname
```

The free space now created can be used for `dd`ing the data in the next volume to be removed.

11. **After all volumes have been converted into disk partitions successfully, reboot the system.**
After reboot, none of the volumes should be found open. This can be verified by using the command:

```
# vxprint -Aht -e v_open
```

If any volumes remain open, repeat the steps listed above.

12. **Shut down the Volume Manager by entering:**

```
# vxdctl stop
# vxiod -f set 0
```

13. **Remove the necessary packages from your system.**

Note – The package SUNWvmman is listed in the following commands, but this package was available for the Version 2.0 software and not for the Version 1.3 software. If you are removing Version 1.3 packages, do not include SUNWvmman in the pkgrm command.

- If you want to remove *only* the Volume Manager disk management software and you want to leave all remaining SPARCstorage Array packages on your system (such as the SPARCstorage Array diagnostics, drivers, and so forth), enter:

```
# pkgrm SUNWvmman SUNWvxvm SUNWvxva
```

- If you want to remove all the packages that came on the SPARCstorage Volume Manager CD from your system, enter:

```
# pkgrm SUNWassa SUNWdiagp SUNWssadv SUNWssahd SUNWssamn SUNWssaop SUNWvmman SUNWvxvm
  SUNWvxva
```

14. When you finish removing the packages, reboot your system:

```
# reboot
```

2.5 What’s Next

Now that you have installed and initialized the Volume Manager software on your SPARCstorage Array, you can perform any of the disk management tasks using the graphical user interface (GUI) or the command line interface (CLI).

Because the GUI is easier to operate than the CLI, you should use the GUI to perform the disk management tasks for your system. For the most part, the CLI is useful only for users who want to write scripts or who have a character-based terminal.

- If you want to operate the SPARCstorage Array by using the GUI, refer to Part 2, “Using the Graphical User Interface.”
- If you want to operate the SPARCstorage Array by using the CLI, refer to Part 3, “Using the Command Line Interface.”

Part 2— Using the Graphical User Interface

Getting Started 3

This chapter gives instructions on bringing up the graphical user interface (GUI) for the Volume Manager. It also gives the basic information to get your SPARCstorage Array up and running.

Note – The Volume Manager GUI has a tutorial mode which allows you to practice setting up your system without making any permanent changes. Before using the GUI for the first time, you may want to use the tutorial mode to practice; see Chapter 16, "Tutorial for the Graphical User Interface."

3.1 Initializing the GUI

In order to run the GUI from the command line, the user's `PATH` environment variable needs to contain `/opt/vxva/bin`. If necessary, update `PATH` to include `/opt/vxva/bin` before attempting to run the GUI.

- For a Bourne Shell (`sh` or `ksh`), update `PATH` to reflect `/opt/vxva/bin`:

```
PATH=/opt/vxva/bin:$PATH export PATH
```

- For a C Shell (`csh` or `tcsh`), update `PATH` to reflect `/opt/vxva/bin`:

```
setenv PATH /opt/vxva/bin:${PATH}
```

To start the GUI from the command line:

1. **Determine if you want to run the GUI from the tutorial or real mode.**
   - When you run the GUI from the tutorial mode, you can go through the process of setting up your system for certain disk management tasks, but none of these changes will actually be applied to your system. The tutorial mode is useful if you want to get used to the GUI and understand the components of the Volume Manager without making any changes to your system's setup. (A more detailed set of procedures using tutorial mode is given in Chapter 16, "Tutorial for the Graphical User Interface.") To run the GUI from the tutorial mode, enter:

```
vxva -t &
```

If this is the first time you've brought up the tutorial, you may see this error message:

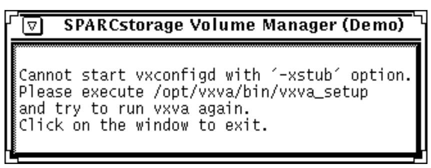

Figure 3-1 Tutorial Error Message

If you see this error message, enter the following command at the prompt:

```
/opt/vxva/bin/vxva_setup
```

Then enter `vxva -t &` once more to run the GUI from the tutorial mode.

   - When you run the GUI from the real mode, any changes you make by using the GUI will be applied directly to your system. To run the GUI from the real mode, first become superuser, then enter:

```
# vxva &
```

Note – You can configure the GUI according to your preferences and system requirements by editing the `.Xdefaults` file. For more information, refer to Appendix B, "GUI-Related X Resources."

A temporary window will appear, showing you that the GUI is starting up, then the root window will appear (see Figure 3-2).

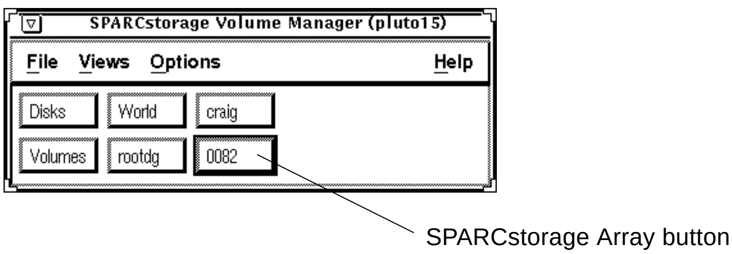

Figure 3-2 Root Window

2. **Determine what the worldwide name is for the SPARCstorage Array that you want to access.**
   If your SPARCstorage Array is up and running properly, you should see four digits in the front panel LCD display at the front of the array (see Figure 3-3). These are the last four digits in the worldwide name for that SPARCstorage Array.

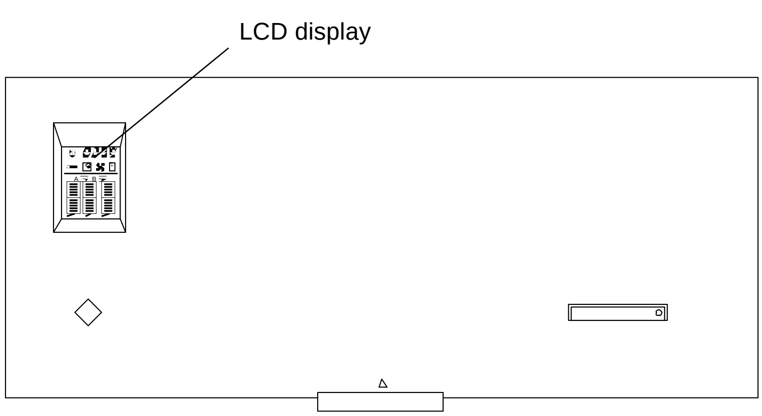

Figure 3-3 Front Panel LCD Display

3. **Locate the screen button in the root window with the same four digits and click LEFT on that screen button (see Figure 3-2).**
   This will bring up the View of SPARCstorage Array for that array.

   - If you have a SPARCstorage Array Model *100* attached to your system, you should see the view shown in Figure 3-4.
   - If you have a SPARCstorage Array Model *200* attached to your system, you should see the view shown in Figure 3-5.

Note – The only difference between the two views is the number and layout of the disks in the SPARCstorage Array; the menu options are the same for both views. The view of the SPARCstorage Array Model 100 will be used throughout this manual when showing menu options, but the same menu options can be found in the view of the SPARCstorage Array Model 200.

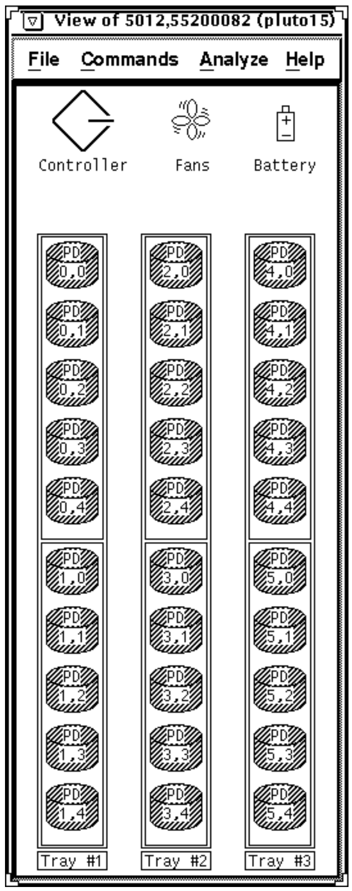

Figure 3-4 View of SPARCstorage Array Model 200

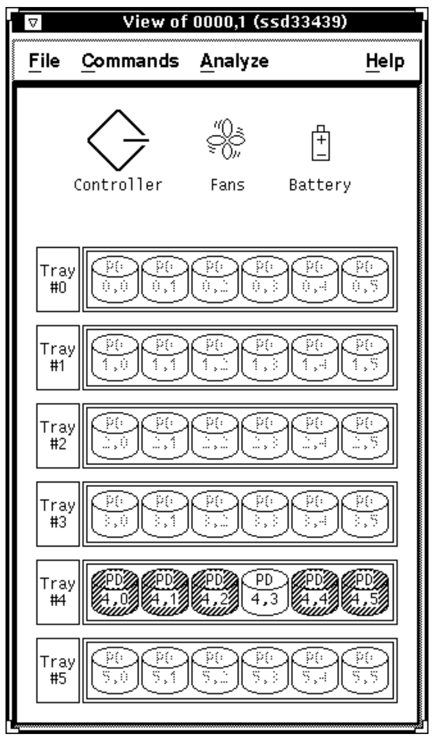

Figure 3-5 View of SPARCstorage Array Model 200

The GUI for your SPARCstorage Array is now up and running. At this point, you can treat the physical disks in your SPARCstorage Array as you would any other set of physical disks. If you want to take full advantage of the disk management tools offered through the SPARCstorage Array, however, there are several options available:

- If you do not want to use the Volume Manager software but you want to perform standard SPARCstorage Array software tasks, go to Section 3.2, "Performing Standard SPARCstorage Array Tasks," on page 3-7. Note that you can also use the options given in this section in conjunction with the Volume Manager software.
- If you want to take advantage of the disk management features offered through Volume Manager (such as striping and mirroring) on one or more of your SPARCstorage Array disks, you must bring these disks under Volume Manager control. (A complete explanation of these disk

management features is given in Section 1.2, “Disk Management With the SPARCstorage Volume Manager,” on page 1-4.) Read the instructions given in Section 3.3, “Bringing Physical Disks under Volume Manager Control,” on page 3-13.

3.1.1 Exiting the GUI

Most GUI windows contain a File menu, which contains one or both of the following options:

- Close — Close the current window only.
- Exit — Exit both the current window and the Volume Manager GUI completely.

To end a GUI session, click LEFT on the File option in the menu bar area of the root window (or any other window) and select Exit. A dialog box appears to confirm that the GUI session is to be closed completely.

Note – Quitting out of a GUI session does not stop the Volume Manager. Volume operations continue to be active unless you manually stop them with the `vxdctl stop` command.

3.2 Performing Standard SPARCstorage Array Tasks

The standard SPARCstorage Array tasks are:

- Reserving and releasing drives
- Setting fast writes

These software features come standard with the SPARCstorage Array software, so you can perform them whether you want to bring the SPARCstorage Array disks under Volume Manager control or not.

- If you plan on having multiple hosts on your system and you want to set up your SPARCstorage Array so that all or some of the drives in the SPARCstorage Array report to only one host, refer to Section 3.2.1, "Reserving and Releasing Drives," on page 3-8.
- If you want to set up your system for fast writes, follow the instructions given in Section 3.2.2, "Setting Fast Writes," on page 3-11 (for an explanation of what fast write does, refer to Section 1.1.1, "Description of Fast Write," on page 1-2).

3.2.1 Reserving and Releasing Drives

The reserve command allows a host system to reserve individual drives or all the drives in a SPARCstorage Array so that no other host systems can use those drives. This is useful if you have more than one host system connected to a SPARCstorage Array but you want only one system to be able to access certain drives. The release command releases these drives from their reserved state. These commands implement the reserve and release functions as defined by the industry standard SCSI-2 specification.

3.2.1.1 Reserving and Releasing All Drives in a SPARCstorage Array

Reserving All the Drives in a SPARCstorage Array

This option reserves all the drives in a SPARCstorage Array for exclusive use by the issuing host.

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 3-2).**
   This will bring up the View of SPARCstorage Array window.

2. **From the Commands menu, select Controller, then Reserve All Drives (see Figure 3-6).**
   If any of the drives had previously been reserved by another system, this command will fail.

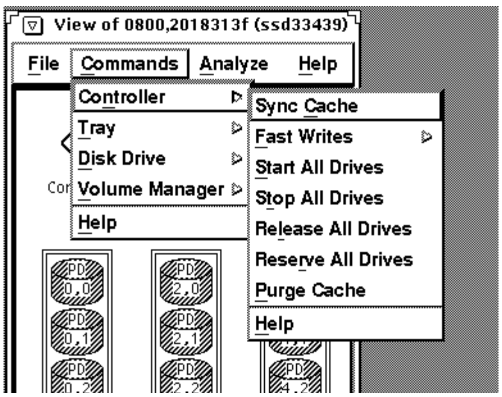

Figure 3-6 Accessing the Controller Menu

Releasing All the Drives in a SPARCstorage Array

This option releases all the drives in a SPARCstorage Array from exclusive use by the issuing host. Use this option if you previously reserved all drives for exclusive use with the instructions in the section "Reserving All the Drives in a SPARCstorage Array."

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 3-2).**
   This will bring up the View of SPARCstorage Array window.

2. **From the Commands menu, select Controller, then Release All Drives (see Figure 3-6).**
   This command will have no affect on drives that were not reserved earlier.

3.2.1.2 Reserving and Releasing Single Disks

Reserving a Physical Disk

This option reserves a physical disk for exclusive use by the issuing host.

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 3-2).**
   This will bring up the View of SPARCstorage Array window.

2. **Click LEFT over the physical disk icon that you want to reserve.**
   The physical disk icon will change its pattern or color, indicating that it has been selected.

3. **From the Commands menu, select Disk Drive, then Reserve (see Figure 3-7).**
   If the drive had previously been reserved by another system, this command will fail.

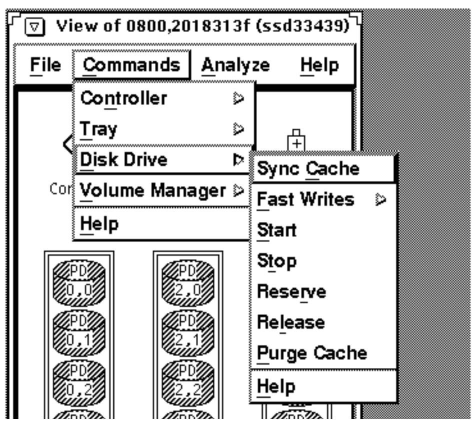

Figure 3-7 Accessing the Disk Drive Menu

Releasing a Physical Disk

This option releases a physical disk from exclusive use by the issuing host. Use this option if you previously reserved the physical disk for exclusive use with the instructions in the section "Reserving a Physical Disk."

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 3-2).**
   This will bring up the View of SPARCstorage Array window.

2. **Click LEFT over the physical disk icon that you want to release.**
   The physical disk icon will change its pattern or color, indicating that it has been selected.

3. **From the Commands menu, select Disk Drive, then Release (see Figure 3-7).**
   This command will have no affect on a drive that was not reserved earlier.

3.2.2 Setting Fast Writes

You can set fast writes from either the controller level, which would set fast writes for all the drives in the SPARCstorage Array, or from the drive level, which would set fast writes only for one specific drive.

- If you want to set fast writes from the controller level, refer to Section 3.2.2.1, "Setting Fast Writes at the Controller Level," on page 3-11.
- If you want to set fast writes from the drive level, refer to Section 3.2.2.2, "Setting Fast Writes at the Drive Level," on page 3-12.

3.2.2.1 Setting Fast Writes at the Controller Level

To set fast writes at the controller level:

1. **Click SELECT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 3-2).** This will bring up the View of SPARCstorage Arrays window.

2. **From the Commands menu, select Controller, then Fast Writes (see Figure 3-8).** From this menu, select either `Enable` to enable fast writes or `Synchronous Only` to enable fast writes for synchronous writes only.

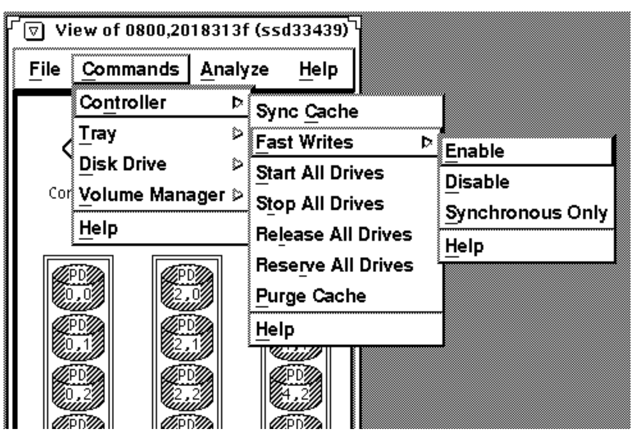

Figure 3-8 Accessing the Fast Write for Controllers Option

Your SPARCstorage Array is now set up so that all the drives within the array take advantage of the fast write option.

If at any point you want to stop fast writes for all the drives in the SPARCstorage Array, follow these instructions:

1. **Click SELECT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 3-2).**
   This will bring up the View of SPARCstorage Arrays window.

2. **From the Commands menu, select Controller, then Fast Writes, then Disable (see Figure 3-8).**

3.2.2.2 *Setting Fast Writes at the Drive Level*

To set fast writes at the drive level:

1. **Click SELECT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 3-2).**
   This will bring up the View of SPARCstorage Arrays window.

2. **Click SELECT over the physical disk icon in the SPARCstorage Array that you want to have set up for fast writes.**

3. **From the Commands menu, select Disk Drive, then Fast Writes (see Figure 3-9).**
   From this menu, select either `Enable` to enable fast writes or `Synchronous Only` to enable fast writes for synchronous writes only.

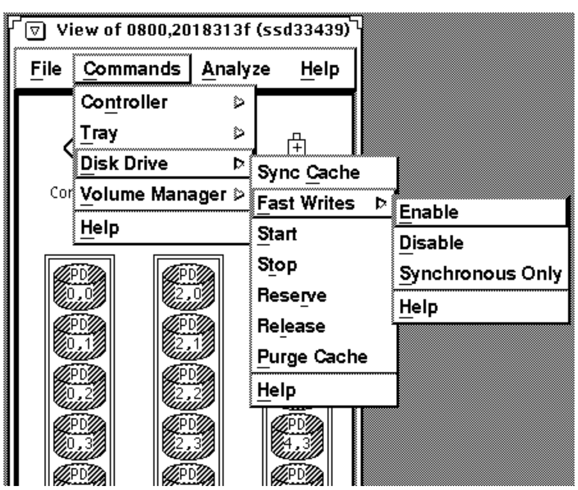

Figure 3-9 Accessing the Fast Write for Drives Option

The disk drive you selected is now set up so that it takes advantage of the fast write option. You can repeat this procedure for other drives, if you want set more than one drive for fast writes.

If at any point you want to stop fast writes for a single drive, follow these instructions:

1. **Click SELECT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 3-2).** This will bring up the View of SPARCstorage Arrays window.

2. **Click SELECT over the physical disk icon in the SPARCstorage Array that you want to stop fast writes for (see Figure 3-9).**

3. **From the Commands menu, select Disk Drive, then Fast Writes, then Disable (see Figure 3-9).**

3.3 Bringing Physical Disks under Volume Manager Control

If you decide to have one or more of your physical disks use the disk management features offered through Volume Manager, you must first bring those disks under Volume Manager control. When you bring a disk under Volume Manager control, a small portion of space on that disk must be free so that the SPARCstorage Volume Manager software can create a private region, which holds information necessary for Volume Manager to be run on that disk. If a disk has a swap partition (such as a boot disk), then Volume Manager uses part of the swap as the private region, so it doesn't need any free space on the boot disk in that case for the private region.

Note – If you have root and swap on separate disks and the boot device has no free space for the private region, you will not be able to encapsulate the boot disk.

- Refer to Section 3.3.1, "Bringing SPARCstorage Array Disks under Volume Manager Control," on page 3-14.

or

- Refer to Section 3.3.2, "Bringing Non-SPARCstorage Array Disks under Volume Manager Control," on page 3-16.

3.3.1 Bringing SPARCstorage Array Disks under Volume Manager Control

Follow the procedures in this section only if:

- you did *not* choose `Quick Installation` when you went through the `vxinstall` procedure earlier (if you chose `Quick Installation`, then all the physical disks connected to your system are already under Volume Manager control), and
- you chose `Leave this disk alone` for one or more physical disks when you chose `Custom Installation` earlier, since these disks are not under Volume Manager control at this time.

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array you want to access (see Figure 3-2).**
   This will bring up the View of SPARCstorage Array (see Figure 3-10).

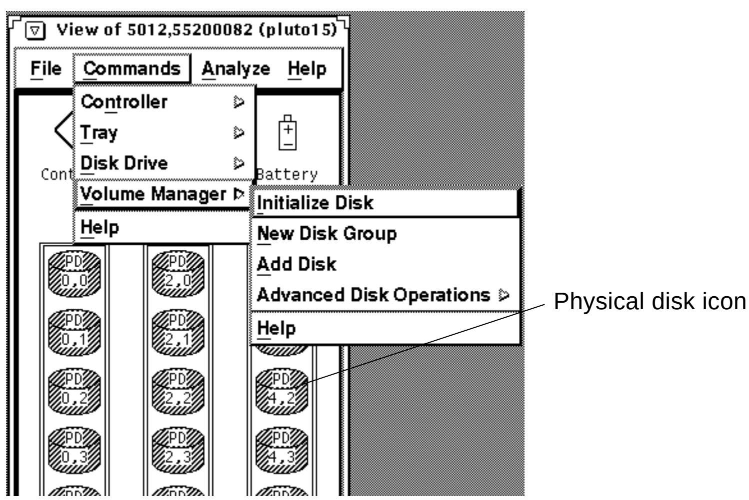

Figure 3-10 Accessing the Volume Manager Menu

2. **Click LEFT over the physical disk icon that you want to bring under Volume Manager control (see Figure 3-10).**

3. **Determine if the physical disk that you have selected was brought under Volume Manager control before.**
   - If the physical disk was brought under Volume Manager control before, go to Step 4.
   - If the physical disk was never brought under Volume Manager control before, then from the Commands menu, select Volume Manager, and Initialize Disk (see Figure 3-10).

4. **Determine if you want to create a new disk group for this physical disk or add it to an existing disk group.**
   - If you want to create a new disk group for this disk, go to Step 5.
   - If you want to add the disk to an existing disk group, go to Step 8.

5. **From the Commands menu, select Volume Manager, then New Disk Group (see Figure 3-10).**
   This will bring up the Disk Group Initialize Form (see Figure 3-11).

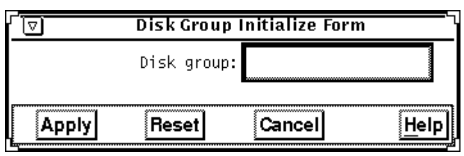

Figure 3-11 Disk Group Initialize Form

6. **Enter a name for the disk group that you want to create and click LEFT over the Apply screen button.**
   This will create the new disk group and add the physical disk to that disk group, thereby bringing it under Volume Manager control. If you look at the root window, you will see the new disk group displayed as a screen button.

7. **Determine if you want to bring more physical disks under Volume Manager control.**
   - If you do not want to bring any more physical disks under Volume Manager control, you do not have to proceed with the remaining steps in this section.
   - If you want to bring more physical disks under Volume Manager control, continue with Step 8.

8. **Go to the root window and click LEFT over the screen button for the disk group that you want to put the disk into (see Figure 3-2).**
   This will bring up the view of the disk group.

9. **Go to the View of SPARCstorage Array window, click LEFT on the physical disk icon that you want to bring under Volume Manager control and drag it into the view of your disk group. Drop the physical disk icon anywhere within that window.**
   A VM disk icon will appear in the view of your disk group. This VM disk is assigned to the physical disk you selected from the SPARCstorage Array window.

10. **Repeat Step 9 for every SPARCstorage Array disk that you want to bring under Volume Manager control.**

Once you have brought the desired SPARCstorage Array disks under Volume Manager control, you can perform any of the disk management tasks available through Volume Manager on those disks. These tasks are discussed in later chapters.

Note – When you bring disks under Volume Manager using these instructions, the VM disks are given default disk media names (disk##). You can change the names of the VM disks by following the instructions in Section 5.5.2, "Changing Names of Objects," on page 5-47.

3.3.2 Bringing Non-SPARCstorage Array Disks under Volume Manager Control

Follow the procedures in this section only if:

- you did *not* choose `Quick Installation` when you went through the `vxinstall` procedure earlier (if you chose `Quick Installation`, then all the physical disks connected to your system are already under Volume Manager control), and
- you chose `Leave this disk alone` for one or more physical disks when you chose `Custom Installation` earlier, since these disks are not under Volume Manager control at this time.

1. **From the root window, click LEFT over the Disks screen button (see Figure 3-12).**

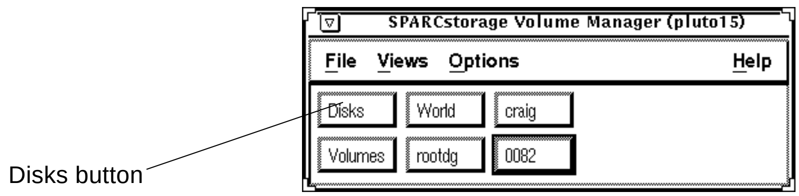

Figure 3-12 Accessing the View of Disks

2. **Determine if you want to create a new disk group for this physical disk or add it to an existing disk group.**
   - If you want to create a new disk group for this disk, go to Step 3.
   - If you want to add the disk to an existing disk group, go to Step 5.

3. **From the Advanced-Ops menu, select Disk Group, then Initialize (see Figure 3-13).**

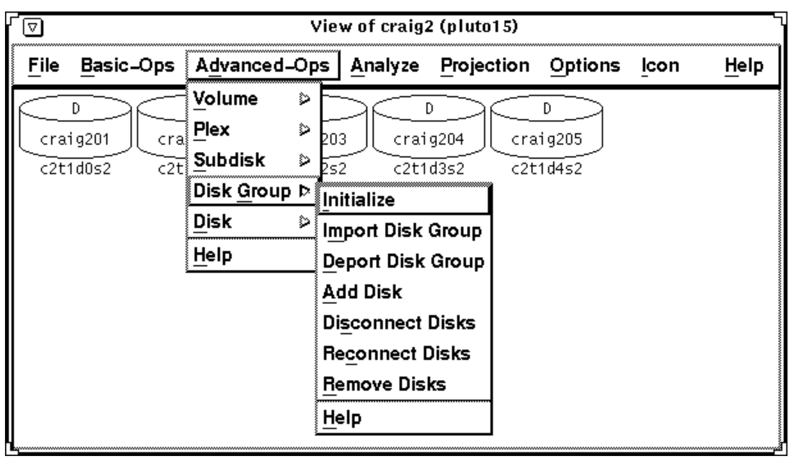

Figure 3-13 Accessing the Disk Group Menu

This will bring up the Disk Group Initialize Form (see Figure 3-14).

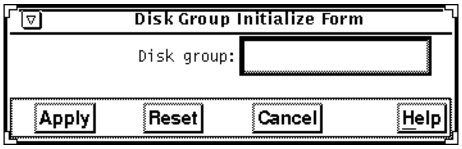

Figure 3-14 Disk Group Initialize Form

4. **Enter a name for the disk group that you want to create and click LEFT over the Apply screen button.**
   The new disk group will now appear in your root window.

5. **From the Advanced-Ops menu, select Disk, then Initialize (see Figure 3-15).**

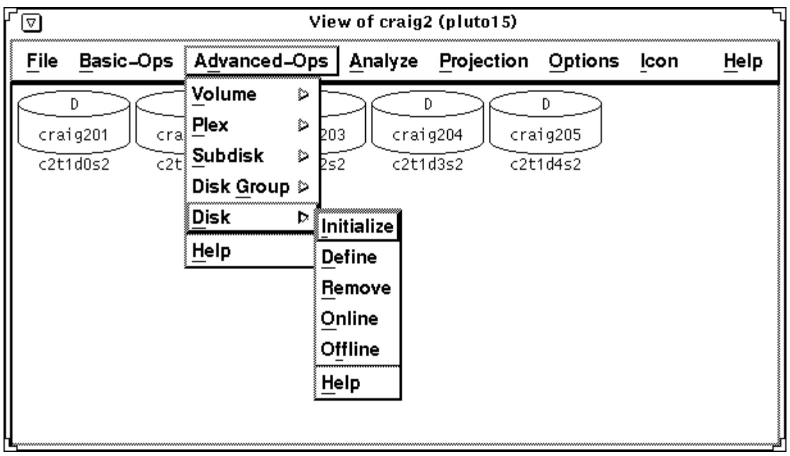

Figure 3-15 Accessing the Disk Menu

This will bring up the Disk Init Form.

6. **Complete the Disk Init form as follows:**

Table 3-1 Disk Init Form

Public Device:	The pathname of the device node that represents a partition available for use. This must be a valid name in `/dev/rdsk/`. A name of the form c#t#d# is appropriate.
Device Type:	Select the desired disk type. The `simple` type (default) assumes that the public and private regions are stored on the same disk partition, with the public region following the private region. The `sliced` type assumes that the public and private regions are stored on different disk partitions. The nopriv type has no private region, and log and configuration copies cannot be written to the disk.
Public length (0 for whole disk):	This is the length of the public section of the disk. If zero is provided as the length, the Volume Manager computes a default value from available partition table information.
Private Length:	This is the length of the private region of the disk. When one isn't specified, the Volume Manager chooses a default value.
Number of config copies:	This is the number of configuration copies to be stored in the private section of this disk. The default value is two copies.
Comment:	Optionally enter a comment for this Volume Manager disk. The maximum length of the comment is 40 characters.

7. **When the form is completed, select Apply to activate the disk initialization.**
   A new physical disk icon containing a partition icon appears.

8. **Access the View of Disks from the root window and click LEFT over the partition in the physical disk icon.**

9. **Drag the partition icon over to the view of the disk group that you want to add the disk to and drop it anywhere in that window.**
   The VM disk will now appear in the view of the disk group.

Once you have brought the appropriate number of disks under Volume Manager control, you can perform any of the disk management tasks available through Volume Manager on those disks. These tasks are given in the following chapters.

Setting Up the Software Configuration 4

Note – You can perform the tasks given in this chapter only on disks that have been placed under Volume Manager control. Refer to Section 3.3, "Bringing Physical Disks under Volume Manager Control," on page 3-13 for those instructions.

This chapter gives the procedures for setting up the configuration for your system. The different types of configurations available are:

- Simple
- Striped (RAID 0)
- Mirrored (RAID 1)
- Striped and mirrored (RAID 0 + 1)
- RAID-5

Table 4-1 gives a brief description of each type of configuration. For a more detailed description and the pros and cons of each type, refer to the *SPARCstorage Array Configuration Guide.*

Table 4-1 Software Configurations Offered Through Volume Manager

Configuration	**Description**
Simple	Data is arranged both sequentially and contiguously over one or more physical disks.
Striped	Data is spread across relatively small, equally-sized fragments that are allocated alternately and evenly across multiple physical disks. Throughput increases with the number of physical disks across which data is striped. Striping helps to balance I/O load in cases where high traffic areas exist on certain physical disks.
Mirrored	Data is duplicated, or mirrored, over two or more physical disks. If one physical disk fails, you can use the mirrored disk as you would the original.
Striped and Mirrored	Data is first striped across multiple physical disks, and is then mirrored over two or more separate physical disks. This is useful if you want to take advantage of both the speed of striping and the data redundancy of mirroring.
RAID-5	Data is spread across relatively small, equally-sized fragments that are allocated alternately and evenly across multiple physical disks. Parity fragments are also created on the same disks, so that each stripe run contains the striped data and a single parity fragment. Compared to the performance of a striped configuration, throughput of RAID-5 configurations decreases, since parity information needs to be updated each time data is accessed. However, compared to mirroring, the amount of space used in RAID-5 is significantly smaller.

Table 4-2 gives a brief explanation of the components of the Volume Manager that are used to set up the configurations. For a more detailed description of Volume Manager and its components, refer to Chapter 1, "Disk Management for the SPARCstorage Array."

Table 4-2 Components of the Volume Manager

Components	Description
VM disk	When a partition from a physical disk is brought under Volume Manager control, Volume Manager creates a *VM disk* that corresponds to that partition.
Disk group	A collection of VM disks that have been grouped together because they have something in common. For example, you could put all the disks from a single SPARCstorage Array into one disk group.
Subdisk	A VM disk can be divided into one or more *subdisks*. Since a VM disk is assigned to a partition on a physical disk, all the subdisks within a VM disk are part of a single physical disk.
Plex	One or more subdisks can be grouped together to form a *plex*. The subdisks in a plex can be located on different VM disks.
Volume	One or more plexes can be grouped together to form a *volume*.

Note – In tutorial mode, the underlying volumes are artificial ones. Since file system commands require real entities in order to succeed, file system operations cannot be simulated by the Volume Manager in tutorial mode.

Figure 4-1 gives a graphical representation of the relationships between these Volume Manager components.

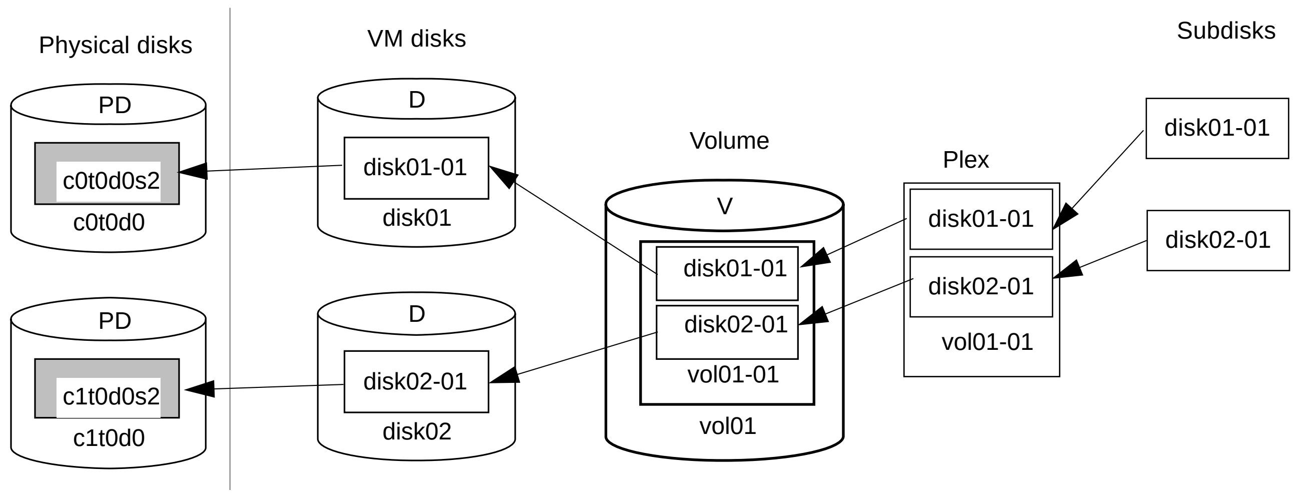

Figure 4-1 Graphical Representation of Volume Manager Components

Once you have decided upon the configuration that you want to use, you will use these components of the Volume Manager to set up that configuration. For example, if you want to set up your system to stripe data across two physical disks, you and / or Volume Manager would:

1. Determine which physical disks you want the data striped across.
2. Locate the VM disks that are assigned to the partition on each physical disk.
3. Create one subdisk on each VM disk to be used for striping. Since you have two VM disks in this example, you would have two subdisks total, one on each VM disk.
4. Group the two subdisks into a single plex.
5. Create a volume to house this plex.

In several sections in this chapter, you will be offered a choice between using the *basic* or *advanced* approach to set up the configuration. Following is an explanation of both approaches:

Table 4-3 Explanation of Basic and Advanced Approaches

Basic	Takes information about what you want to accomplish and then performs the necessary underlying tasks so that you don't have to perform any lower-level procedures. This approach requires only minimal input from you, but does permit more detailed specifications.
Advanced	Consists of a number of fairly complicated commands that typically require you to specify detailed input. These commands use a "building block" approach that requires you to have a detailed knowledge of the underlying structure and components in order to manually perform the sequences of commands necessary to accomplish a certain task.

Following are the different configurations you can create using the Volume Manager:

4.1 Setting Up a Simple Configuration

A simple configuration is one in which data is arranged both sequentially and contiguously over one or more physical disks. You will set up a simple configuration by creating a simple volume which can then be used for file systems, data bases, and so forth.

- If you want to create a simple volume for file systems, follow the instructions in Section 4.1.1, "Creating a Simple Volume for a File System," on page 4-6.
- If you want to create a simple volume for something other than file systems (such as a data base), follow the instructions in Section 4.1.2, "Creating a Simple Volume," on page 4-9.

4.1.1 Creating a Simple Volume for a File System

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select the VM disk(s) to be used for the new volume, if necessary.**
   - If you select one VM disk, the data will reside on the physical disk that is associated with that VM disk. Ensure that the VM disk has sufficient free space to accommodate the desired length of the new volume by clicking RIGHT on an unobscured portion of the VM disk icon to access its properties form, then checking the value in the Maximum free space: field.
   - If you select more than one VM disk, then the data will be concatenated across the physical disks that are associated with those VM disks. Ensure that the VM disks have sufficient free space to accommodate the desired length of the new volume by clicking RIGHT on an unobscured portion of each VM disk icon to access its properties form, then check the value in the Maximum free space: field.
   - If you do not select any VM disk, disks with sufficient free space will automatically be used.
3. **From the Basic-Ops menu, select File System Operations, then Create (see Figure 4-2).**
   A submenu listing basic volume types appears.

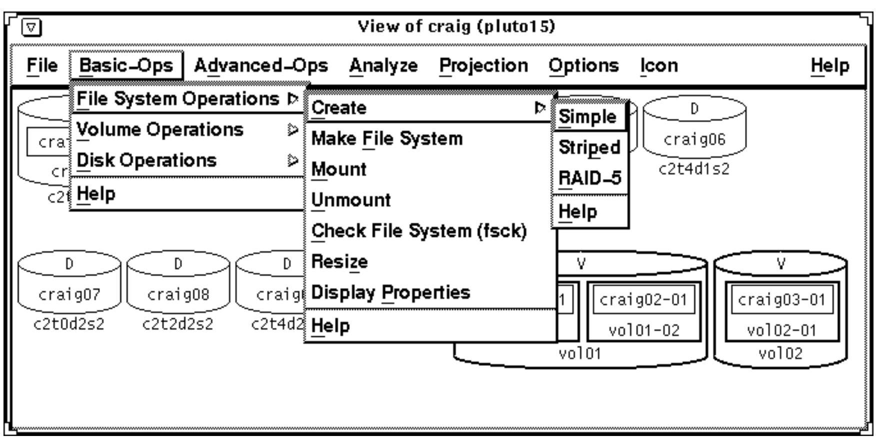

Figure 4-2 Accessing the File System Create Menu

4. Select Simple.

The Simple Volume / FS Create form appears (see Figure 4-3). This form creates a file system on a concatenated volume. The form is divided into two sections, one for volume creation and the other for file system creation. Most of the form fields are set to the defaults for the creation of a file system on a new volume.

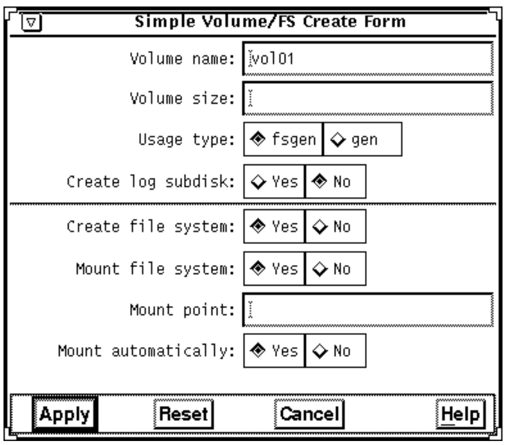

Figure 4-3 Simple Volume / FS Create Form

5. Complete the Simple Volume/FS Create Form as follows:

Table 4-4 Simple Volume / FS Create Form

Volume name:	Either leave the default name or enter a different one. The name must be unique within this disk group. The maximum length of this field is 31 characters.
Volume size:	Enter the desired volume size. The size should be entered as a number followed immediately by the letter `k`, `m`, g, or `s` to indicate kilobytes, megabytes, gigabytes or sectors, respectively. If no unit is specified, the default is sectors.
Usage type:	Select a usage type. The default, `fsgen`, is already selected. `fsgen` is the file system generic usage type, which assumes that the volume is being used by a file system. `gen` is the generic usage type, which makes no assumptions regarding the data content of the volume.

Table 4-4 Simple Volume / FS Create Form *(Continued)*

Create log subdisk:	Select Yes or No. A log subdisk is used to log recent disk activity.
Create file system:	Indicate whether a file system is to be created. Since the object of this operation is to create a file system, `Yes` is already selected.
Mount file system:	Indicate whether the file system should be mounted after creation. If the answer is `Yes` (the default), a mount point must also be specified in the next field. All fields below this field are only accessible when `Yes` is specified here.
Mount point:	Enter the desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it.
Mount automatically:	Indicate whether this file system should be mounted every time the system comes up (by placing an entry in `/etc/vfstab`). `Yes` is the default.

6. **When the form is completed, select Apply to create the volume and file system.**
   A new volume icon appears. Since this is a simple, concatenated volume, it contains a single plex. If the file system is mounted, it is represented by the mount point, which appears below the new volume.

Figure 4-4 shows a simple, concatenated volume with a mounted file system.

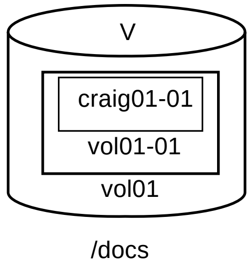

Figure 4-4 Simple Volume and File System

4.1.2 Creating a Simple Volume

You can create a simple volume using either a *basic* approach or an *advanced* approach:

- Basic approach — Section 4.1.2.1 on this page
- Advanced approach — Section 4.1.2.2 on page 4-12

4.1.2.1 Basic Approach for Creating a Simple Volume

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select the VM disk(s) to be used for the new volume, if necessary.**
   - If you select one VM disk, the data will reside on the physical disk that is associated with that VM disk. Ensure that the VM disk has sufficient free space to accommodate the desired length of the new volume by clicking RIGHT on an unobscured portion of the VM disk icon to access its properties form, then checking the value in the Maximum free space: field.
   - If you select more than one VM disk, then the data will be concatenated across the physical disks that are associated with those VM disks. Ensure that the VM disks have sufficient free space to accommodate the desired length of the new volume by clicking RIGHT on an unobscured portion of each VM disk icon to access its properties form, then check the value in the Maximum free space: field.
   - If you do not select any VM disk, disks with sufficient free space will automatically be used.
3. **From the Basic-Ops menu, select Volume Operations, then Create (see Figure 4-5).**
   A submenu listing basic volume types appears.

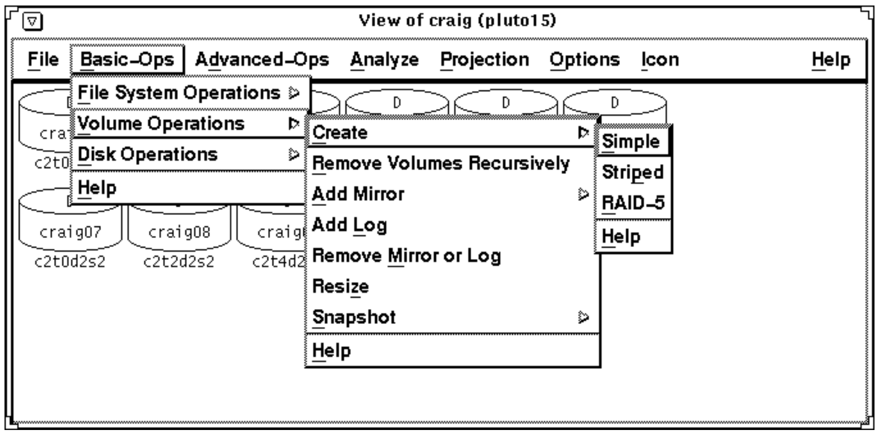

Figure 4-5 Accessing the Volume Operations Create Menu

4. Select Simple.

The Simple Volume/FS Create form appears (see Figure 4-6). This form creates a concatenated volume and optionally creates a file system on the new volume. The form is divided into two sections, one for volume creation and the other for file system creation. Most of the form fields are already set to the defaults for the creation of a new volume. The file system fields are grayed out because the default is not to add a file system to the volume.

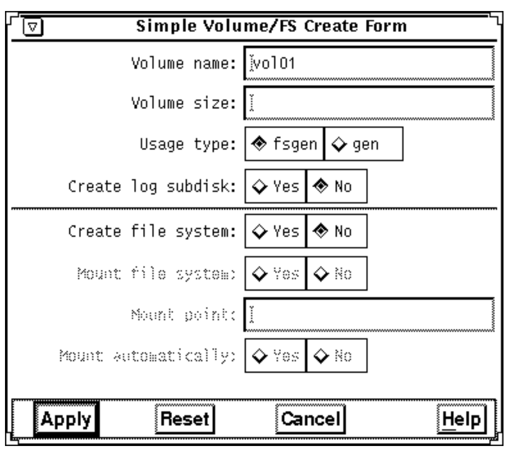

Figure 4-6 Simple Volume/FS Create Form

5. **Complete the Simple Volume/FS Create form as follows:**

Table 4-5 Top Half of Simple Volume/FS Create Form

Volume name:	Either leave the default name or enter a different one. The name must be unique within this disk group. The maximum length of this field is 31 characters.
Volume size:	Enter the desired volume size. The size should be entered as a number followed immediately by the letter `k`, `m`, g, or `s` to indicate kilobytes, megabytes, gigabytes or sectors, respectively. If no unit is specified, the default is sectors.
Usage type:	Select a usage type. The default, `fsgen`, is already selected. `fsgen` is the file system generic usage type, which assumes that the volume is being used by a file system. `gen` is the generic usage type, which makes no assumptions regarding the data content of the volume.
Create log subdisk:	Select Yes or No. A log subdisk is used to log recent disk activity.
Create file system:	Indicate whether a file system is to be created. Since the default is not to create a file system, `No` is already selected. Select `Yes` if you want to create a file system on this volume.

The following fields only apply if you set the Create file system: field to `Yes`. Otherwise, these fields are inaccessible.

Table 4-6 Bottom Half of Simple Volume/FS Create Form

Mount file system:	Indicate whether the file system should be mounted after creation. If the answer is `Yes` (the default), a mount point must also be specified in the next field. All fields below this field are only accessible when `Yes` is specified here.
Mount point:	Enter the desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it.
Mount automatically:	Indicate whether this file system should be mounted every time the system comes up (by placing an entry in `/etc/vfstab`). `Yes` is the default.

6. **When the form is properly completed, select Apply to activate the volume (and optional file system) creation.**
   A new volume icon appears. Since this is a simple, concatenated volume, it contains a single plex. If a file system exists and is mounted, it is represented by the mount point, which appears below the new volume.

Figure 4-7 illustrates a simple, concatenated volume.

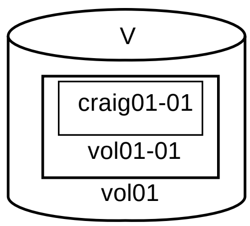

Figure 4-7 Simple Volume

4.1.2.2 Advanced Approach for Creating a Simple Volume

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select a VM disk on which to create the subdisk.**
   Ensure that the VM disk has sufficient free space to accommodate the desired length of the new volume by clicking RIGHT on an unobscured portion of the VM disk icon to access its properties form, then check the value in the Maximum free space: field. If there is sufficient free space, select the VM disk by clicking LEFT on its icon.
3. **From the Advanced-Ops menu, select Subdisk, then Create.**
   The Subdisk Create form appears (see Figure 4-8). This form establishes the parameters for the subdisk. Some of the fields already contain default values and others are optional; only the subdisk length is required.

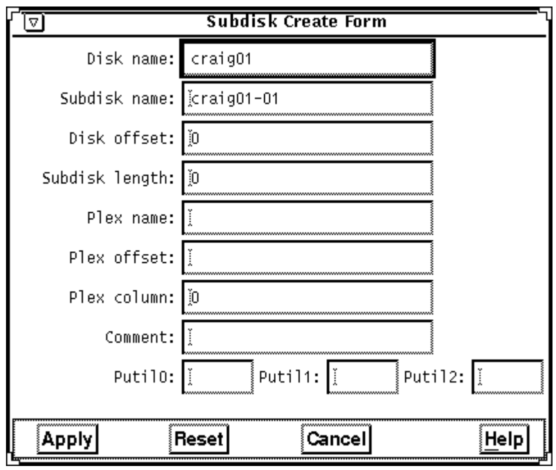

Figure 4-8 Subdisk Create Form

4. Complete the Subdisk Create form, entering the subdisk length and optionally altering other fields.

Table 4-7 Subdisk Create Form

Subdisk length:	Enter the length of the subdisk to be created. If no units are specified, the number is assumed to be in sectors. This offset should not place this subdisk within the bounds of another subdisk on the disk or past the end of the disk. The maximum length of this field is 14 characters.

Refer to this form's Help window for more information on any of its fields.

5. When the form is properly completed, select Apply to activate the subdisk creation.

A new subdisk icon appears in the selected disk icon.

6. Drag the new subdisk icon (by pressing and continuing to hold the LEFT button) beyond the borders of its disk and into an open area of the view window. Drop the subdisk by releasing the LEFT button.

The Plex Create form appears (see Figure 4-9).

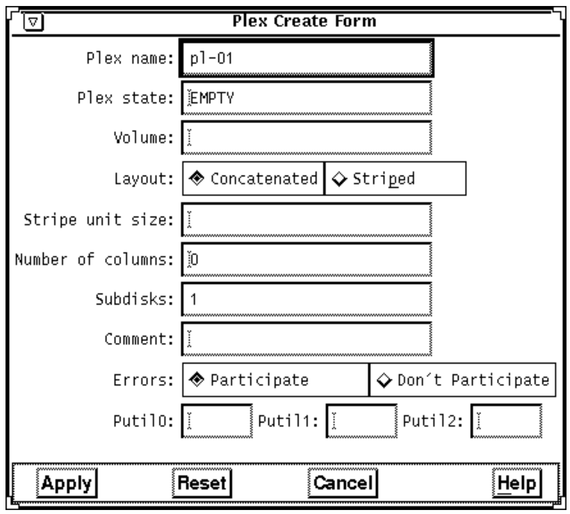

Figure 4-9 Plex Create Form

When this form is invoked in this way, it will automatically create a new plex and associate the dragged subdisk with it. The Plex Create form establishes the parameters for the plex. Some of the fields already contain default values and others are optional. Since this is to be a concatenated (simple) volume, the layout must be set to Concatenated.

7. **Complete the Plex Create form, indicating a concatenated layout and optionally altering any fields.**
   Refer to this form's Help window for more information on any of its fields.

8. **When the form is properly completed, select Apply to activate both the plex creation and the association of subdisk and plex.**
   A plex icon containing a subdisk icon appears. The fact that the subdisk icon is within the plex icon confirms their association. The length of the plex should be the same as that of the subdisk.

9. **Select the new plex by clicking the LEFT button on its icon.**
   When the plex is selected immediately prior to the creation of the volume in this way, the plex will automatically be associated with the new volume.

10. **From the Advanced-Ops menu, select Volume, then Create.**
    The Volume Create form appears (see Figure 4-10).

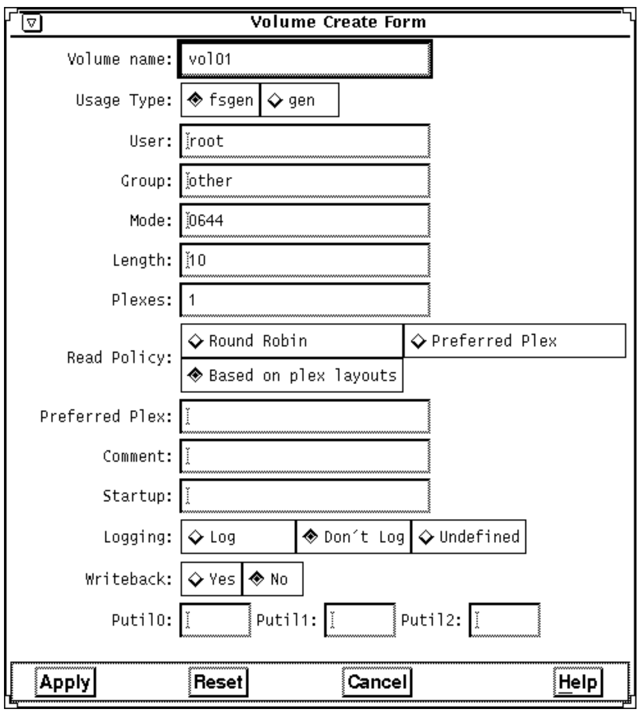

Figure 4-10 Volume Create Form

This form establishes the parameters for the volume. Some of the fields already contain default values and others are optional. Notice that the length is the same as that of the subdisk in the selected plex.

11. **Complete the Volume Create form, optionally altering any fields.**

12. **When the form is properly completed, select Apply to activate both the volume creation and the association of plex and volume.**
    A volume icon appears around the plex and its associated subdisk. The fact that the plex icon is within the volume icon confirms their association.

13. **Select the volume that you just created by clicking the LEFT button on its icon.**

14. From the Advanced-Ops menu, select Volume, then Initialize Volumes.
A submenu appears and offers the following choices for the volume initialization:

Table 4-8 Volume Initialization Form

Active	Enable the selected volume and its associated plexes, and set the state of all associated plexes to ACTIVE.
Enable	Enable the selected volume and its associated plexes, but leave the plex states as EMPTY.
Clean	Set the state for all associated plexes of the selected volume to CLEAN. This can be applied only under limited circumstances.
Zero	Enable the selected volume and its associated plexes, then write zeroes over the entire volume. After the operation completes, all associated plexes are set to ACTIVE, assuming that there are no I/O errors.

15. Select Active (this is suggested for most circumstances).

4.2 Setting Up a Striped Configuration

A striped configuration is one in which data is spread across relatively small, equally-sized fragments that are allocated alternately and evenly across multiple physical disks. You can set up a striped configuration by creating a striped volume which can then be used for file systems, database management system, and so forth.

Caution – There are several striped configurations that are not allowed or not recommended for the SPARCstorage Array. Refer to page 1-15 for general striping guidelines and to the section entitled “Boot Device Rules” on page 1-33 for boot device guidelines before proceeding with this section.

- If you want to create a striped volume for file systems, follow the instructions in Section 4.2.1, “Creating a Striped Volume for a File System,” on page 4-17.
- If you want to create a striped volume for something other than file systems (such as a database management system), follow the instructions in Section 4.2.2, “Creating a Striped Volume,” on page 4-20.

4.2.1 Creating a Striped Volume for a File System

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select at least two VM disk(s) to be used for the new volume, if necessary.**
   - If you select the VM disks, ensure that they have sufficient free space to accommodate the desired length of the new volume by clicking RIGHT on an unobscured portion of each VM disk icon to access its properties form, then check the value in the Maximum free space: field. The VM disks belong to the same disk group.
   - If you do not select any VM disk, disks with sufficient free space will automatically be used.

3. **From the Basic-Ops menu, select File System Operations, then Create (see Figure 4-2).**
   A submenu listing basic volume types appears.

4. **Select Striped.**
   The Striped Volume/FS Create form appears (see Figure 4-11). This form creates a file system on a striped volume. The form is divided into two sections, one for volume creation and the other for file system creation. Most of the form fields are already set to the defaults for the creation of a file system on a new volume.

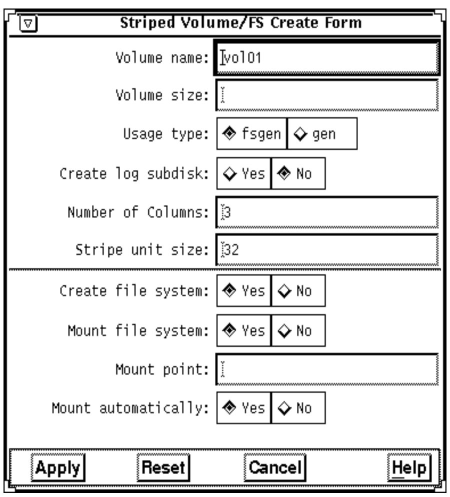

Figure 4-11 Striped Volume/FS Create Form

5. Complete the Striped Volume/FS Create form as follows:

Table 4-9 Striped Volume / FS Create Form

Volume name:	Either leave the default name or enter a different one. The name must be unique within this disk group. The maximum length of this field is 31 characters.
Volume size:	Enter the desired volume size. The size should be entered as a number followed immediately by the letter `k, m,` g, or `s` to indicate kilobytes, megabytes, gigabytes or sectors, respectively. If no unit is specified, the default is sectors.
Usage type:	Select a usage type. The default, `fsgen`, is already selected. `fsgen` is the file system generic usage type, which assumes that the volume is being used by a file system. `gen` is the generic usage type, which makes no assumptions regarding the data content of the volume.
Create log subdisk:	Select Yes or No. A log subdisk is used to log recent disk activity.
Number of Columns:	Enter the number of stripes that the volume's plex is to have. This is effectively the number of disks on which the volume is to be created. If some number of disks have already been selected, that number of stripes appears in this field. This number corresponds to the number of disks across which data will be striped.
Stripe unit size:	This is the width of the stripes on the volume's plex. The value specified may be optimized for your particular drive configuration. The size should be entered as a number followed immediately by the letter `k, m,` g, or `s` to indicate kilobytes, megabytes, gigabytes or sectors, respectively. The default value for this field is 32 sectors. The stripe unit size should be calculated as follows: *stripe_unit_size* = *number_of_columns* * *n* / *vol_size*, where *n* is an integer greater than zero.
Create file system:	Indicate whether a file system is to be created. Since the object of this operation is to create a file system, `Yes` is already selected.

Table 4-9 Striped Volume / FS Create Form *(Continued)*

Mount file system:	Indicate whether the file system should be mounted after creation. If the answer is `Yes` (the default), a mount point must also be specified in the next field. All fields below this field are only accessible when `Yes` is specified here.
Mount point:	Enter the desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it.
Mount automatically:	Indicate whether this file system should be mounted every time the system comes up (by placing an entry in `/etc/vfstab`). `Yes` is the default.

6. **When the form is properly completed, select Apply to create the volume and file system.**

   A new volume icon appears. Since this is a striped volume, it contains a single plex and multiple subdisks. If the file system is mounted, it is represented by the mount point, which appears below the new volume.

   Figure 4-12 shows a striped volume with a mounted file system.

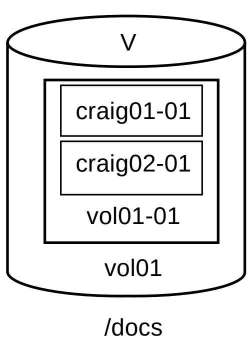

Figure 4-12 Striped Volume and File System

4.2.2 Creating a Striped Volume

You can create a striped volume using either a *basic* approach or an *advanced* approach.

- Basic approach — Section 4.2.2.1 on this page
- Advanced approach — Section 4.2.2.2 on page 4-23

4.2.2.1 Basic Approach for Creating a Striped Volume

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select at least two VM disks to be used for the new volume.**
   These VM disks must belong to the same disk group. Ensure that the VM disks have sufficient free space to accommodate their portion of the desired length of the new volume by clicking RIGHT on an unobscured portion of each VM disk icon to access its properties form, then check the value in the Maximum free space: field.

   If there is sufficient free space, select the VM disks by clicking MIDDLE on their icons. If no VM disks are selected, disks with sufficient free space are automatically used.

3. **From the Basic-Ops menu, select Volume Operations, then Create (see Figure 4-5).**
   A submenu listing basic volume types appears.

4. **Select Striped.**
   The Striped Volume/FS Create form appears (see Figure 4-13). This form creates a concatenated volume and optionally creates a file system on the new volume. The form is divided into two sections, one for volume creation and the other for file system creation. Most of the form fields are already set to the defaults for the creation of a new volume. The file system fields are grayed out because the default is not to add a file system to the volume.

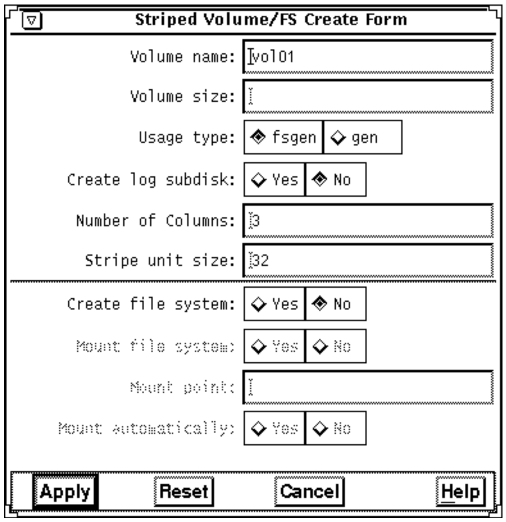

Figure 4-13 Striped Volume/FS Create Form

5. Complete the Striped Volume/FS Create form as follows:

Table 4-10 Top Half of Striped Volume/FS Create Form

Volume name:	Either leave the default name or enter a different one. The name must be unique within this disk group. The maximum length of this field is 14 characters.
Volume size:	Enter the desired volume size. The size should be entered as a number followed immediately by the letter `k`, `m`, or `s` to indicate kilobytes, megabytes, or sectors, respectively. If no unit is specified, the default is sectors.
Usage type:	Select a usage type. The default, `fsgen`, is already selected. `fsgen` is the file system generic usage type, which assumes that the volume is being used by a file system. `gen` is the generic usage type, which makes no assumptions regarding the data content of the volume.
Create log subdisk:	Select Yes or No. A log subdisk is used to log recent disk activity. Only one log subdisk can exist on a given plex.

Table 4-10 Top Half of Striped Volume / FS Create Form *(Continued)*

Number of Columns:	Enter the number of stripes that the volume's plex is to have. This is effectively the number of disks on which the volume is to be created. If some number of disks have already been selected, that number of stripes appears in this field. This number corresponds to the number of disks across which data will be striped.
Stripe unit size:	This is the width of the stripes on the volume's plex. The value specified may be optimized for your particular drive configuration. The size should be entered as a number followed immediately by the letter `k`, `m`, g, or `s` to indicate kilobytes, megabytes, gigabytes or sectors, respectively. The default value for this field is 32 sectors. The stripe unit size should be calculated as follows: *stripe_unit_size* = *number_of_columns* * *n* / *vol_size*, where *n* is an integer greater than zero.
Create file system:	Indicate whether a file system is to be created. Since the default is not to create a file system, `No` is already selected. Select `Yes` if you want to create a file system on this volume.

The following fields apply only if you set the Create file system: field to `Yes`. Otherwise, these fields are inaccessible.

Table 4-11 Bottom Half of Striped Volume / FS Create Form

Mount file system:	Indicate whether the file system should be mounted after creation. If the answer is `Yes` (the default), a mount point must also be specified in the next field. All fields below this field are only accessible when `Yes` is specified here.
Mount point:	Enter the desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it.
Mount automatically:	Indicate whether this file system should be mounted every time the system comes up (by placing an entry in `/etc/vfstab`). `Yes` is the default.

6. **When the form is properly completed, select Apply to activate the volume and file system creation.**
   A new volume icon appears. Since this is a striped volume, it contains a single plex and multiple subdisks. If a file system exists and is mounted, it is represented by the mount point, which appears below the new volume.

   Figure 4-14 shows a striped volume.

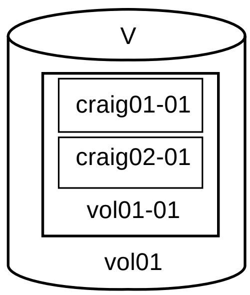

Figure 4-14 Striped Volume

4.2.2.2 *Advanced Approach for Creating a Striped Volume*

When creating a striped volume, at least two disks must be selected across which to stripe the data on the volume's plex.

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select the VM disk on which to create the first subdisk.**
   Ensure that the VM disk has sufficient free space to accommodate the desired length of the new volume by clicking RIGHT on an unobscured portion of that VM disk icon to access its properties form, then check the value in the Maximum free space: field. If there is sufficient free space, select the VM disk by clicking LEFT on its icon.

3. **From the Advanced-Ops menu, select Subdisk, then Create.**
   The Subdisk Create form appears (see Figure 4-8). This form establishes the parameters for the subdisk. Some of the fields already contain default values and others are optional; only the subdisk length is required.

4. **Complete the Subdisk Create form, entering the subdisk length and optionally altering other fields.**

Table 4-12 Subdisk Create Form

Subdisk length:	Enter the length of the subdisk to be created. If no units are specified, the number is assumed to be in sectors. This offset should not place this subdisk within the bounds of another subdisk on the disk or past the end of the disk. The maximum length of this field is 14 characters.

Refer to this form's Help window for more information on any of its fields.

5. **When the form is properly completed, select Apply to activate the subdisk creation.**
   A new subdisk icon appears in the selected disk icon.

6. **Repeat the previous steps to select another VM disk in this disk group and create a subdisk of the same length as the first one on that VM disk.**

7. **Select all of the new subdisks by clicking the MIDDLE button on their icons.**

8. **From the Advanced-Ops menu, select Plex, then Create.**
   The Plex Create form appears (see Figure 4-9). When this form is invoked while subdisks are selected, it will automatically create a new plex and associate the selected subdisks with it. The Plex Create form establishes the parameters for the plex. Some of the fields already contain default values and others are optional. The number of stripes is set to the number of selected subdisks (two). Since this is to be a striped volume, the layout must be set to Striped and the stripe width must be entered.

9. Complete the Plex Create form, indicating stripe information and optionally altering any other fields.

Table 4-13 Plex Create Form

Layout	Select Striped as the layout for this plex.
Stripe unit size:	Enter the width of the stripes on this plex. If no units are specified, the number entered here is assumed to be in sectors. Only positive numbers greater than zero are accepted. This field is required, as a striped plex layout was specified. The maximum length of this field is 14 characters. Best striping performance is achieved when the stripe width corresponds to the track width of the drive. The stripe unit size should be calculated as follows: *stripe_unit_size* = *number_of_columns* * *n* / *vol_size*, where *n* is an integer greater than zero.

Refer to this form's Help window for more information on any of its fields.

10. When the form is properly completed, select Apply to activate both the plex creation and the association of subdisks and plex.
A plex icon containing both subdisk icons appears. The fact that the subdisk icons are within the plex icon confirms their association.

11. Select the plex by clicking the LEFT button on its icon.
When the plex is selected immediately prior to the creation of the volume in this way, it will automatically be associated with the new volume.

12. From the Advanced-Ops menu, select Volume, then Create.
The Volume Create form appears (see Figure 4-10). This form establishes the parameters for the volume. Some of the fields already contain default values and others are optional. Notice that the length is the combined length of the subdisks in the selected plex.

13. Complete the Volume Create form, optionally altering any fields. Refer to this form's Help window for more information on any of its fields.

14. When the form is properly completed, select Apply to activate both the volume creation and the association of plexes and volume.
A volume icon appears around the plex and its associated subdisks. The fact that the plex icon is within the volume icon confirms their association.

15. **Select the volume that is to be initialized by clicking the LEFT button on its icon.**

16. **From the Advanced-Ops menu, select Volume, then Initialize Volumes. A submenu appears and offers the following choices for the volume initialization:**

Table 4-14 Volume Initialization Form

Active	Enable the selected volume and its associated plexes, and set the state of all associated plexes to ACTIVE.
Enable	Enable the selected volume and its associated plexes, but leave the plex states as EMPTY.
Clean	Set the state for all associated plexes of the selected volume to CLEAN. This can be applied only under limited circumstances.
Zero	Enable the selected volume and its associated plexes, then write zeroes over the entire volume. After the operation completes, all associated plexes are set to ACTIVE, assuming that there are no I/O errors.

17. **Select Active (this is suggested for most circumstances).**

4.3 Setting Up a Mirrored Configuration

A mirrored configuration is one in which data is mirrored over two or more physical disks, thus creating an automatic backup of all your data. You can set up a mirrored configuration by first creating a volume (either simple or striped) and then creating a mirror for that volume. The mirror layout can be simple or striped.

Caution – There are several mirrored configurations that are not allowed or not recommended for the SPARCstorage Array. Refer to page 1-28 for general mirroring guidelines.

Note – If you are creating more than one mirror for a volume, you must wait until each mirror is completely made before you can create the next one. If a mirror is grayed out on the screen, it is in the process of being created; you must wait until it is no longer gray before you can create another mirror for that volume.

You can create a mirrored volume using either a *basic* approach or an *advanced* approach.

- Basic approach — Section 4.3.1 on page 4-27
- Advanced approach — Section 4.3.2 on page 4-28

4.3.1 Basic Approach for Creating a Mirrored Volume

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Create a simple or striped volume.**
   - If you want to create a simple volume, see Section 4.1, "Setting Up a Simple Configuration," on page 4-5.
   - If you want to create a striped volume, see Section 4.2, "Setting Up a Striped Configuration," on page 4-16.

Note – You cannot add a mirror to a RAID-5 volume.

---

3. **Select the new volume to which a mirror is to be added by clicking LEFT on its icon.**
4. **Select the VM disk(s) to be used for the mirror, if necessary.**
   - If the mirror is to be striped, you must select more than one VM disk. Ensure that the VM disks have sufficient free space to accommodate the desired length of the new volume by clicking RIGHT on an unobscured portion of each VM disk icon to access its properties form, then check the value in the Maximum free space: field. These VM disks must belong to the same disk group.
   - If you do not select any VM disk, disks with sufficient free space will automatically be used.
5. **From the Basic-Ops menu, select Volume Operations, then Add Mirror.** A submenu listing mirror layout options appears.
6. **Select either Simple Mirror or Striped Mirror, depending on the desired mirror layout.** The volume's icon expands visibly and a new mirror appears within its borders. The new mirror layout depends on what was specified during the Add Mirror operation.

Figure 4-15 shows a striped volume with a striped mirror.

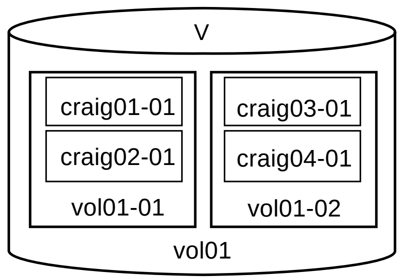

Figure 4-15 Striped and Mirrored Volume

4.3.2 Advanced Approach for Creating a Mirrored Volume

When creating a mirrored volume, you must create at least two plexes on at least two disks. Both plexes must then be associated with the same volume.

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select the VM disk on which to create the first subdisk.**
   Ensure that the VM disk has sufficient free space to accommodate the desired length of the new volume by clicking RIGHT on an unobscured portion of that VM disk icon to access its properties form, then check the value in the Maximum free space: field. If there is sufficient free space, select the VM disk by clicking LEFT on its icon.

3. **From the Advanced-Ops menu, select Subdisk, then Create.**
   The Subdisk Create form appears (see Figure 4-8). This form establishes the parameters for the subdisk. Some of the fields already contain default values and others are optional; only the subdisk length is required.

4. **Complete the Subdisk Create form, entering the subdisk length and optionally altering other fields.**

Table 4-15 Subdisk Create Form

Subdisk length:	Enter the length of the subdisk to be created. If no units are specified, the number is assumed to be in sectors. This offset should not place this subdisk within the bounds of another subdisk on the disk or past the end of the disk. The maximum length of this field is 14 characters.

Refer to this form's Help window for more information on any of its fields.

5. **When the form is properly completed, select Apply to activate the subdisk creation.**
   A new subdisk icon appears in the selected disk icon.

6. **Drag the new subdisk icon (by pressing and continuing to hold the LEFT button) beyond the borders of its disk and into an open area of the view window. Drop the subdisk by releasing the LEFT button.**
   The Plex Create form appears (see Figure 4-9). When this form is invoked in this way, it will automatically create a new plex and associate the dragged subdisk with it. This form establishes the parameters for the plex. Some of the fields already contain default values and others are optional. Since this volume is to contain two concatenated (simple) plexes for mirroring, the layout must be set to Concatenated.

7. **Complete the Plex Create form, indicating a concatenated layout and optionally altering any other fields.**
   Refer to this form's Help window for more information on any of its fields.

8. **When the form is properly completed, select Apply to activate both the plex creation and the association of subdisk and plex.**
   A plex icon containing a subdisk icon appears. The fact that the subdisk icon is within the plex icon confirms their association. The length of the plex should be the same as that of the subdisk.

9. **Repeat Step 2 through Step 8 to select another VM disk in this disk group, create a subdisk of the same length as the first one on that disk, create a plex of the same type as the first, and associate the second subdisk with the second plex.**

10. **Select both plexes by clicking the MIDDLE button on their icons.**
When the plexes are selected immediately prior to the creation of the volume in this way, they will automatically be associated with the new volume.

11. **From the Advanced-Ops menu, select Volume, then Create.**
The Volume Create form appears (see Figure 4-10). This form establishes the parameters for the volume. Some of the fields already contain default values and others are optional. Notice that the length is the same as that of the subdisks in the selected plexes.

12. **Complete the Volume Create form, optionally altering any fields.**
Refer to this form's Help window for more information on any of its fields.

13. **When the form is properly completed, select Apply to activate both the volume creation and the association of plexes and volume.**
A volume icon appears around both plexes and their associated subdisks. The plexes are side-by-side in the volume, to illustrate that they are mirrored. The fact that both plex icons are within the volume icon confirms their association.

14. **Select the volume that is to be initialized by clicking the LEFT button on its icon.**

15. **From the Advanced-Ops menu, select Volume, then Initialize. A submenu appears and offers the following choices for the volume initialization:**

Table 4-16 Volume Initialization Form

Active	Enable the selected volume and its associated plexes, and set the state of all associated plexes to ACTIVE.
Enable	Enable the selected volume and its associated plexes, but leave the plex states as EMPTY.
Clean	Set the state for all associated plexes of the selected volume to CLEAN. This can be applied only under limited circumstances.
Zero	Enable the selected volume and its associated plexes, then write zeroes over the entire volume. After the operation completes, all associated plexes are set to ACTIVE, assuming that there are no I/O errors.

16. **Select Active (this is suggested for most circumstances).**

4.4 Setting Up a Striped and Mirrored Configuration

A striped and mirrored configuration is one in which data is first striped across multiple physical disks, and is then mirrored over one or more physical disk(s). You can set up a striped and mirrored configuration by first creating a striped volume and then creating a mirror for that volume. The mirror layout can be simple or striped.

The number of available disks must be sufficient to accommodate the layout type of both the existing volume and the mirror to be added. At least three disks are required to create a striped and mirrored volume. The striped plex requires two disks (one per stripe); the plex that acts as the mirror must occupy a third disk if it is simple or a third and fourth disk if it is also striped. If the disks occupied by these plexes overlap, the striping and mirroring will not both be effective.

You can create a striped and mirrored volume using either a *basic* approach or an *advanced* approach.

- Basic approach — Section 4.4.1 on this page
- Advanced approach — Section 4.4.1 on page 4-31

4.4.1 Basic Approach for a Striped and Mirrored Volume

1. **Create a striped volume using the instructions given in Section 4.2, “Setting Up a Striped Configuration,” on page 4-16.**

2. **Select the new volume to which a mirror is to be added by clicking LEFT on its icon.**

3. **Select the VM disk(s) to be used for the mirror, if necessary.**
   - If the mirror is to be striped, you must select more than one VM disk. Ensure that the VM disks have sufficient free space to accommodate the desired length of the new volume by clicking RIGHT on an unobscured portion of each VM disk icon to access its properties form, then check the value in the Maximum free space: field. These VM disks must belong to the same disk group.
   - If you do not select any VM disk, disks with sufficient free space will automatically be used.

4. **From the Basic-Ops menu, select Volume Operations, then Add Mirror.**
   A submenu listing mirror layout options appears.

5. **Select either Simple Mirror or Striped Mirror, depending on the desired mirror layout.**
   The striped volume's icon expands visibly and a new mirror appears within its borders. The new mirror layout depends on what was specified during the Add Mirror operation.

Figure 4-16 illustrates a striped volume with a striped mirror.

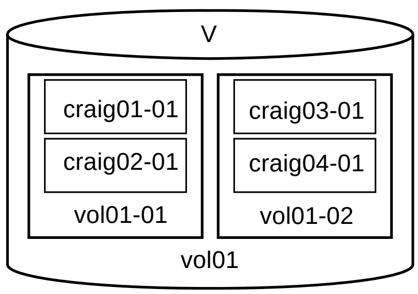

Figure 4-16 Striped and Mirrored Volume

4.4.2 *Advanced Approach for a Striped and Mirrored Volume*

You can create a volume that is both striped and mirrored. The simplest way to do this is to create a striped volume and then add a mirror to that volume. Since a striped volume is being mirrored, it is preferable for the mirror layout to be striped in order to maintain the advantages of striping.

At least four disks are required to create a striped volume with a striped mirror. The original striped plex requires two disks and the plex that acts as a striped mirror requires an additional two disks for mirroring to be effective. If only three disks are available, a simple mirror can be added instead (although this can cause some performance drawbacks).

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Create a striped volume using the instructions given in Section 4.2, "Setting Up a Striped Configuration," on page 4-16.**

3. **Select a third VM disk on which to create the first subdisk for the striped mirror.**
   You can use projection to ensure that the third VM disk is not currently being used by the striped volume's subdisks (by pressing Shift-RIGHT on the striped volume's icon and observing which VM disks now contain highlighted subdisks). The VM disk should have sufficient space to accommodate a subdisk as long as the subdisk length in the striped volume. If there is sufficient free space, select the VM disk by clicking LEFT on its icon.

4. **From the Advanced-Ops menu, select Subdisk, then Create.**
   The Subdisk Create form appears (see Figure 4-8). This form establishes the parameters for the subdisk. Some of the fields already contain default values and others are optional; only the subdisk length is required.

5. **Complete the Subdisk Create form, entering the subdisk length (that matches that of an existing subdisk in the striped volume) and optionally altering other fields.**

Table 4-17 Subdisk Create Form

Subdisk length:	Enter the length of the subdisk to be created. If no units are specified, the number is assumed to be in sectors. This offset should not place this subdisk within the bounds of another subdisk on the disk or past the end of the disk. The maximum length of this field is 14 characters.

Refer to this form's Help window for more information on any of its fields.

6. **When the form is properly completed, select Apply to activate the subdisk creation.**
   A new subdisk icon appears in the selected disk icon.

7. **To create an identically sized subdisk on a fourth disk, drag the newly created subdisk icon (by pressing and continuing to hold the LEFT button) onto a VM disk that contains sufficient space to accommodate a subdisk of the same length and is not already in use by the striped volume or the selected subdisk. Drop the subdisk onto the desired VM disk by releasing the LEFT button.**
   A dialog box appears to announce that the dragged subdisk was not associated and that Volume Manager has therefore created a new subdisk rather than moving one. Select the Continue button to proceed. A new subdisk icon appears on its targeted VM disk.

8. **Select both of the new subdisks by clicking MIDDLE on their icons.**

9. **From the Advanced-Ops menu, select Plex, then Create.**
   The Plex Create form appears (see Figure 4-9). This form establishes the parameters for the plex. Some of the fields already contain default values and others are optional. The number of stripes is set to the number of selected subdisks (two). Since a striped plex is to be used for mirroring, the layout must be set to Striped.

10. **Complete the Plex Create form, indicating stripe information and optionally altering any other fields.**

Table 4-18 Plex Create Form

Layout:	Select Striped as the layout for this plex, as a striped volume is to be created.
Stripe unit size:	Since a striped layout was selected, enter the width of the stripes on this plex. If no units are specified, the number entered here is assumed to be in sectors. Only positive numbers greater than zero are accepted. This field is required, as a Striped plex layout was specified. The maximum length of this field is 14 characters. Best striping performance is achieved when the stripe width corresponds to the track width of the drive. The stripe unit size should be calculated as follows: *stripe_unit_size* = *number_of_columns* * *n* / *vol_size*, where *n* is an integer greater than zero.

Refer to this form's Help window for more information on any of its fields.

11. **When the form is properly completed, select Apply to activate both the plex creation and the association of subdisk and plex.**
    A plex icon containing two subdisk icons appears. The fact that the subdisk icons are within the plex icon confirms their association. The length of the plex should be the same as that of the volume to be mirrored.

12. **The new plex icon can now be associated with the striped volume, putting mirroring into effect. Drag the new plex icon (by pressing and continuing to hold the LEFT button) onto the striped volume icon. Position the plex so that the pointer (in the image of a hand) is directly over an unobscured portion of the volume icon. Drop the plex by releasing the LEFT button.**
    The striped volume's icon expands visibly and the mirror appears within its borders.

13. **Select the volume that is to be initialized by clicking the LEFT button on its icon.**

14. **From the Advanced-Ops menu, select Volume, then Initialize. A submenu appears and offers the following choices for the volume initialization:**

Table 4-19 Volume Initialization Form

Active	Enable the selected volume and its associated plexes, and set the state of all associated plexes to ACTIVE.
Enable	Enable the selected volume and its associated plexes, but leave the plex states as EMPTY.
Clean	Set the state for all associated plexes of the selected volume to CLEAN. This can be applied only under limited circumstances.
Zero	Enable the selected volume and its associated plexes, then write zeroes over the entire volume. After the operation completes, all associated plexes are set to ACTIVE, assuming that there are no I/O errors.

15. **Select Active (this is suggested for most circumstances).**

4.5 Setting Up a RAID-5 Configuration

A RAID-5 configuration is one in which data is spread across relatively small, equally-sized fragments that are allocated alternately and evenly across multiple physical disks. Parity fragments are also created on the same disks, so that each stripe run contains the striped data and a single parity fragment.

- If you want to create a RAID-5 volume for file systems, follow the instructions in Section 4.5.1, “Creating a RAID-5 Volume for a File System,” on page 4-35.
- If you want to create a RAID-5 volume for something other than file systems (such as a data base), follow the instructions in Section 4.5.2, “Creating a RAID-5 Volume,” on page 4-39.

4.5.1 Creating a RAID-5 Volume for a File System

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select at least two VM disks to be used for the new volume.**
While you can use two VM disks for a RAID-5 configuration, keep in mind that RAID-5 configurations work best if you spread the data across four or more VM disks. Also keep in mind that a RAID-5 log subdisk is highly recommended, so if you are going to add a RAID-5 log subdisk to the configuration, a subdisk from one of the VM disks that you select will be used (in other words, if you choose four VM disks, three VM disks will be used for the RAID-5 volume and one VM disk will be used to hold the RAID-5 log subdisk). For more information on RAID-5 logs, see "RAID 5" on page 1-16.

These VM disks must belong to the same disk group. Ensure that the VM disks have sufficient free space to accommodate their portion of the desired length of the new volume by clicking RIGHT on an unobscured portion of each VM disk icon to access its properties form, then checking the value in the Maximum free space: field.

If there is sufficient free space, select the VM disks by clicking MIDDLE on their icons. If no VM disks are selected, disks with sufficient free space are automatically used.

3. **From the Basic-Ops menu, select File System Operations, then Create (see Figure 4-2).**
A submenu listing basic volume types appears.

4. **Select RAID-5.**
The RAID-5 Volume/FS Create form appears (see Figure 4-17). This form creates a RAID-5 volume and optionally creates a file system on the new volume. The form is divided into two sections, one for volume creation and the other for file system creation. Most of the form fields are already set to the defaults for the creation of a new volume. The file system fields are grayed out because the default is not to add a file system to the volume.

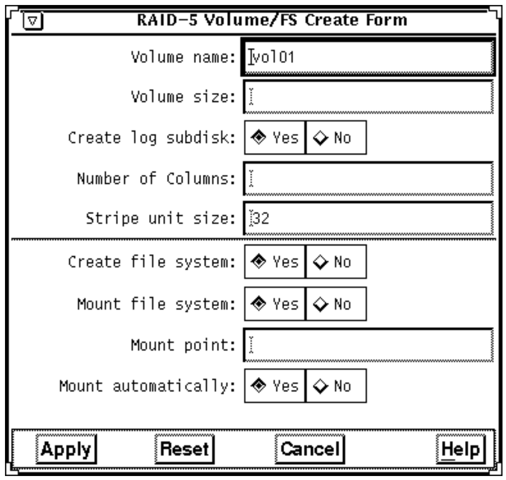

Figure 4-17 RAID-5 Volume / FS Create Form

5. Complete the RAID-5 Volume/FS Create form as follows:

Table 4-20 RAID-5 Volume / FS Create Form

Volume name:	Either leave the default name or enter a different one. The name must be unique within this disk group. The maximum length of this field is 14 characters.
Volume size:	Enter the desired volume size. The size should be entered as a number followed immediately by the letter `k`, `m`, `g`, or `s` to indicate kilobytes, megabytes, gigabytes or sectors, respectively. If no unit is specified, the default is sectors. If the size specified is not wholly divisible by the stripe width, it is rounded up before the volume is created. Since RAID-5 reserves separate space for parity information, the actual space contained in plexes is larger than the addressable sized of the volume. The size specified in this field represents the usable space.
Create log subdisk:	Indicate whether you want to create a log subdisk or not; the default is `Yes`. At least one log subdisk should be created for a RAID-5 volume.

Table 4-20 RAID-5 Volume / FS Create Form *(Continued)*

Number of Columns:	This field represents the number of columns that the volume's plex is to have. Typically, the number needed for RAID-5 should be four columns or more. If you chose `Yes` in the `Create log subdisk` field, this field will be initialized to one fewer than the number of VM disks selected to allow for the creation of a log subdisk.
Stripe unit size:	This is the width of the stripes on the volume's plex. The value specified may be optimized for the particular drive configuration. For best striping performance, the stripe width should correspond to the track width of the drive. The default value for this field is 32 sectors, since this is a good stripe width for most systems.
Create file system:	Indicate whether a file system is to be created. Since the default is not to create a file system, `No` is already selected. Select `Yes` if you want to create a file system on this volume.
Mount file system:	Indicate whether the file system should be mounted after creation. If the answer is `Yes` (the default), a mount point must also be specified in the next field. All fields below this field are only accessible when `Yes` is specified here.
Mount point:	Enter the desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it.
Mount automatically:	Indicate whether this file system should be mounted every time the system comes up (by placing an entry in `/etc/vfstab`). `Yes` is the default.

6. **When the form is properly completed, select Apply to create the volume.**
   A new volume icon appears. Since this is a RAID-5 volume, it contains a single plex and multiple subdisks. If a file system exists and is mounted, it is represented by the mount point, which appears below the new volume.

   Figure 4-18 shows a RAID-5 volume with a RAID-5 log subdisk.

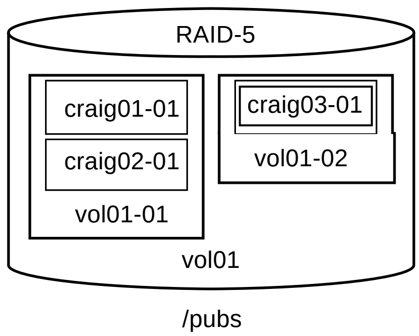

Figure 4-18 RAID-5 File System and RAID-5 Log

4.5.2 Creating a RAID-5 Volume

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select at least two VM disks to be used for the new volume.**
   While you can use two VM disks for a RAID-5 configuration, keep in mind that RAID-5 configurations work best if you spread the data across four or more VM disks. Also keep in mind that a RAID-5 log subdisk is highly recommended, so if you are going to add a RAID-5 log subdisk to the configuration, a subdisk from one of the VM disks that you select will be used (in other words, if you choose four VM disks, three VM disks will be used for the RAID-5 volume and one VM disk will be used to hold the RAID-5 log subdisk). For more information on RAID-5 logs, see "RAID 5" on page 1-16.

   These VM disks must belong to the same disk group. Ensure that the VM disks have sufficient free space to accommodate their portion of the desired length of the new volume by clicking RIGHT on an unobscured portion of each VM disk icon to access its properties form, then checking the value in the Maximum free space: field.

   If there is sufficient free space, select the VM disks by clicking MIDDLE on their icons. If no VM disks are selected, disks with sufficient free space are automatically used.

3. **From the Basic-Ops menu, select Volume Operations, then Create (see Figure 4-5).**
   A submenu listing basic volume types appears.

4. **Select RAID-5.**
   The RAID-5 Volume/FS Create form appears (see Figure 4-19). This form creates a RAID-5 volume and optionally creates a file system on the new volume. The form is divided into two sections, one for volume creation and the other for file system creation. Most of the form fields are already set to the defaults for the creation of a new volume. The file system fields are grayed out because the default is not to add a file system to the volume.

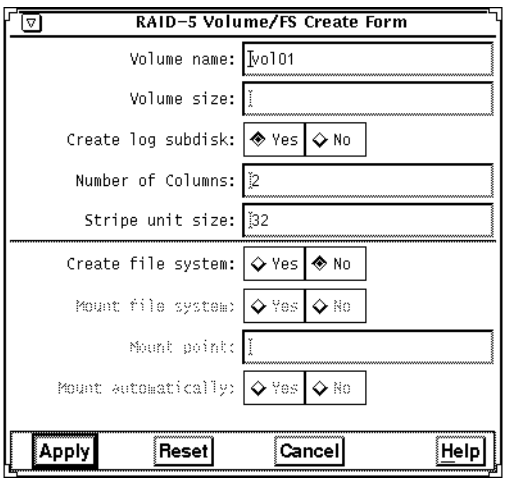

Figure 4-19 RAID-5 Volume/FS Create Form

5. Complete the RAID-5 Volume/FS Create form as follows:

Table 4-21 Top Half of RAID-5 Volume/FS Create Form

Volume name:	Either leave the default name or enter a different one. The name must be unique within this disk group. The maximum length of this field is 31 characters.
Volume size:	Enter the desired volume size. The size should be entered as a number followed immediately by the letter `k`, `m`, `g`, or `s` to indicate kilobytes, megabytes, gigabytes or sectors, respectively. If no unit is specified, the default is sectors. If the size specified is not wholly divisible by the stripe width, it is rounded up before the volume is created. Since RAID-5 reserves separate space for parity information, the actual space contained in plexes is larger than the addressable sized of the volume. The size specified in this field represents the usable space.
Create log subdisk:	Indicate whether you want to create a log subdisk or not; the default is `Yes`. At least one log subdisk should be created for a RAID-5 volume.
Number of Columns:	This field represents the number of columns that the volume's plex is to have. Typically, the number needed for RAID-5 should be four columns or more. If you chose `Yes` in the `Create log subdisk` field, this field will be initialized to one fewer than the number of VM disks selected to allow for the creation of a log subdisk.
Stripe unit size:	This is the width of the stripes on the volume's plex. The value specified may be optimized for the particular drive configuration. For best striping performance, the stripe width should correspond to the track width of the drive. The default value for this field is 32 sectors, since this is a good stripe width for most systems.
Create file system:	Indicate whether a file system is to be created. Since the default is not to create a file system, `No` is already selected. Select `Yes` if you want to create a file system on this volume.

The following fields apply only if you set the Create file system: field to Yes. Otherwise, these fields are inaccessible.

Table 4-22 Bottom Half of Striped Volume / FS Create Form

Mount file system:	Indicate whether the file system should be mounted after creation. If the answer is Yes (the default), a mount point must also be specified in the next field. All fields below this field are only accessible when Yes is specified here.
Mount point:	Enter the desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it.
Mount automatically:	Indicate whether this file system should be mounted every time the system comes up (by placing an entry in /etc/vfstab). Yes is the default.

6. **When the form is properly completed, select Apply to create the volume.**
A new volume icon appears. Since this is a RAID-5 volume, it contains a single plex and multiple subdisks. If a file system exists and is mounted, it is represented by the mount point, which appears below the new volume.

Figure 4-20 shows a RAID-5 volume with a RAID-5 log.

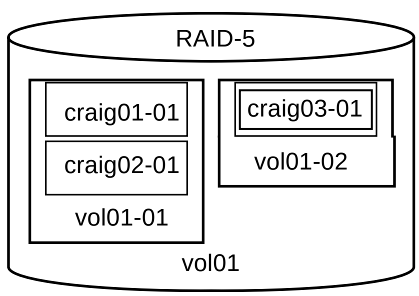

Figure 4-20 RAID-5 Volume and RAID-5 Log

Changing Components of the Software Configuration 5

This chapter gives the procedures for changing all or part of the software configuration. It is assumed that you understand the terminology presented in Chapter 4, “Setting Up the Software Configuration,” and have set up your configuration based on the instructions given in that chapter.

5.1 Adding Components to a Configuration

5.1.1 Creating a Disk Group

Note – The instructions in this section apply only to SPARCstorage Array disks. If you want to create a disk group for non-SPARCstorage Array disks, refer to Appendix A, "Using the Volume Manager Software on Non-SPARCstorage Array Disks."

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array you want to access (see Figure 5-1).**

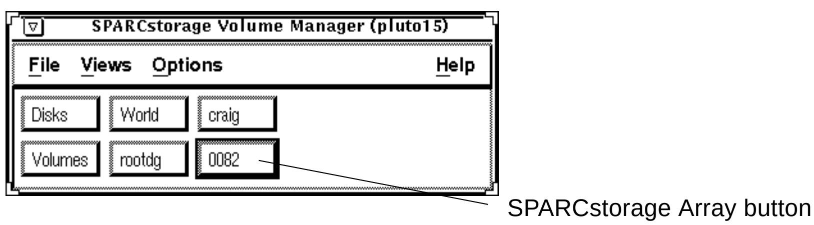

Figure 5-1 Accessing a SPARCstorage Array

This will bring up the View of SPARCstorage Array (see Figure 5-2).

2. **Click LEFT over the physical disk icon that you want to bring under Volume Manager control (see Figure 5-2).**

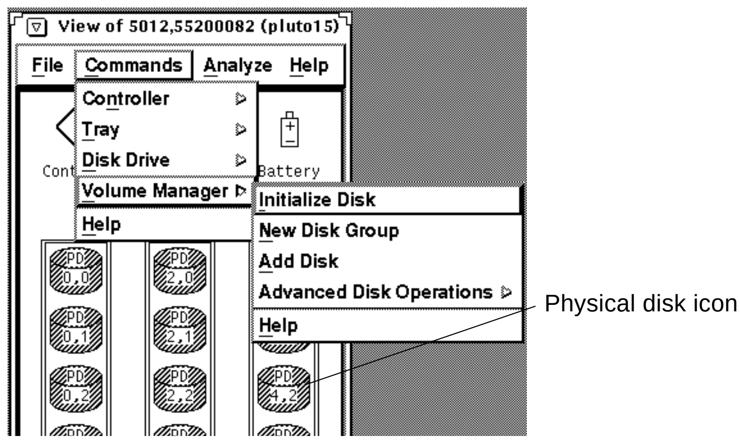

Figure 5-2 Accessing the Volume Manager Menu

3. From the Commands menu, select Volume Manager, then New Disk Group (see Figure 5-2).
This will bring up the Disk Group Initialize Form (see Figure 5-3).

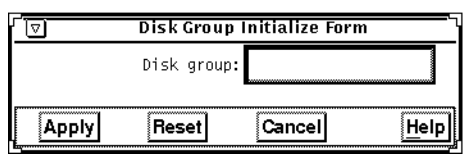

Figure 5-3 Disk Group Initialize Form

4. Enter a name for the disk group that you want to create and click LEFT over the Apply screen button.
This will create the new disk group and add the physical disk to that disk group, thereby bringing it under Volume Manager control. If you look at the root window, you will see the new disk group displayed as a screen button.

5.1.2 Adding a Physical Disk to a Disk Group

Note – The instructions in this section apply only to SPARCstorage Array disks. If you want to add a non-SPARCstorage Array disk to a disk group, refer to Appendix A, “Using the Volume Manager Software on Non-SPARCstorage Array Disks.”

1. **Go to the root window and click LEFT over the screen button for the disk group that you want to add the disk drive to (see Figure 5-1).**
   This will bring up the view of that disk group.

2. **Go to the View of SPARCstorage Array window, click LEFT on the physical disk icon that you want to add to the disk group and drag it into the view of the disk group. Drop the physical disk icon anywhere within that window.**
   A VM disk icon will now appear in the view of the disk group. This VM disk is assigned to the physical disk you selected from the SPARCstorage Array window.

3. **Repeat Step 2 for every disk drive that you want to add to the disk group.**

You can also add a disk to a disk group using menus:

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array you want to access (see Figure 5-1).**

2. **Click LEFT on the physical disk icon that you want to bring add to the disk group.**

3. **From the Commands menu, select Volume Manager, then Add Disk (see Figure 5-2).**
   This will bring up the Add Disk Form (see Figure 5-4).

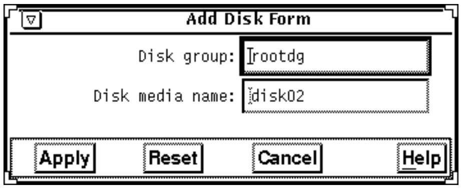

Figure 5-4 Add Disk Form

4. **Complete the Add Disk Form and click LEFT over the Apply screen button.**
   The VM disk will now appear in the view of the disk group.

5.1.3 Adding a VM Disk as a Hot Spare

You can add one or more VM disks to a disk group as hot spares. Hot spares are used to automatically recover data when a physical disk fails. The hot spare will only be effective for data that is on a mirrored volume; if one of the two disks from the mirrored volume fails and a hot spare is present in the same disk group, then the information from the failed disk is automatically placed on the hot spare. For more information on hot spares, refer to "Hot-Sparing" on page 1-29.

Note – Using the Volume Manager GUI, you can create a plex on a volume that is assigned to a hot spare disk. This is not a problem if the plex assigned to the hot spare disk is so small that the hot spare disk has enough space remaining to allow for data from a failed disk in that disk group. However, if the plex assigned to the hot spare disk is so large that the hot spare disk doesn't have enough room left to accommodate the data from a failed disk in the same disk group, then hot sparing for that hot spare disk will not go into effect.

To add a VM disk as a hot spare:

1. **Go to the disk group view where the hot spare is to be created.**

2. **Select the VM disk that is to be used as a hot spare.**
   If you want to make a new VM disk into a hot spare, follow the instructions given in Section 5.1.2, "Adding a Physical Disk to a Disk Group," on page 5-5 to create a new VM disk.

3. **Click RIGHT over the VM disk icon to bring up the properties window (see Figure 5-5).**

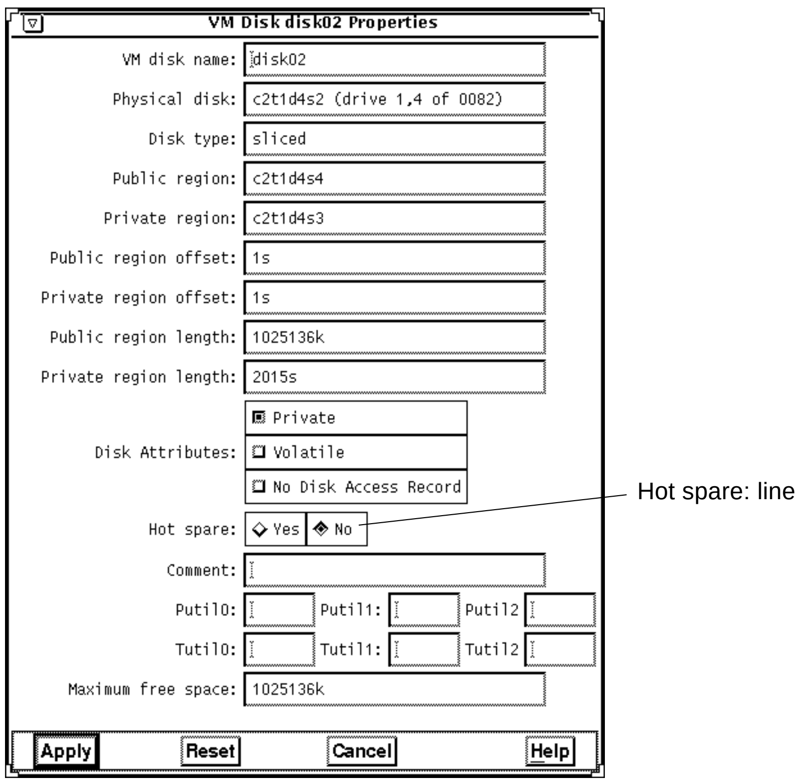

Figure 5-5 VM Disk Properties Window

4. **Locate the Hot spare: line in the properties window and click LEFT over Yes.**

5. **Click LEFT over the Apply screen button to make the VM disk a hot spare.**
   The `D` at the top of the VM disk icon will change to an `HS`, showing that it is now a hot spare.

5.1.4 Importing a Disk Group

Importing a disk group allows you to access a disk group that has been deported. To import a deported disk group, you must know its former name and leave at least one partition unused (containing a disk access record) that was formerly assigned to the deported disk group. If you have reused all disks associated with a deported disk group, you cannot import that disk group.

1. **Click LEFT over the screen button in the root window for the view you want to access (see Figure 5-1).**

2. **From the Advanced-Ops menu, select Disk Group, then Import Disk Group (see Figure 5-6).**

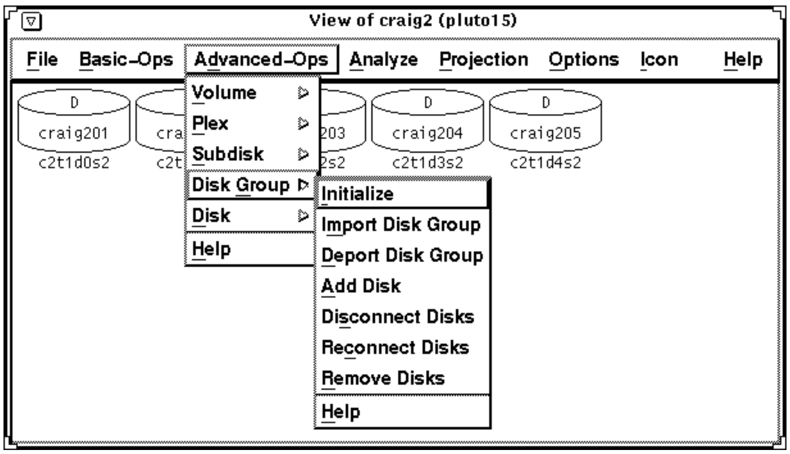

Figure 5-6 Accessing the Disk Group Menu

The Import Disk Group form appears (see Figure 5-7).

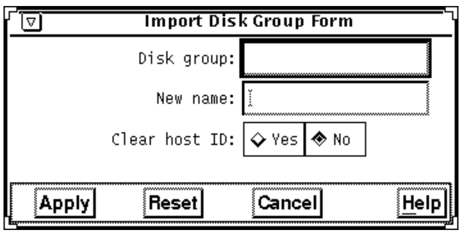

Figure 5-7 Import Disk Group Form

3. Complete the Import Disk Group form, entering the name of the disk group to be imported.

Refer to Table 5-1 for an explanation of the fields in the Import Disk Group form. The named disk group must have been deported at one time. At least one partition formerly assigned to the deported disk group must still exist (unused) in order for the disk group to be importable. Note that you would select `Yes` on the `Clear host ID` field only for one of the following cases:

- After you have physically moved disks between machines and you are in the process of importing the disk group onto a new host.
- If you are certain that only one machine is attempting to access a particular disk group at a given time (though two machines may be duel porting the same disk).

Table 5-1 Import Disk Group Form

Field	**Description**
Disk group:	The name of the disk group to be imported and made available to the system. This must be a valid and unique disk group name. This field is required. This is a read / write field.
New name:	This disk group will be renamed to the contents of this field when imported. This field does not contain a default value. If you do not fill in this field, the disk group will not be renamed.
Clear host ID:	Use this field with caution to clear the existing host ID that is recorded on all disks in the disk group. The default is `No`.

4. When the form is properly completed, select Apply to activate the disk import.

A view button corresponding to the disk group that has been imported reappears in the root window. Click LEFT on the view button to open the view of the disk group. The partition icons corresponding to the VM disk icons that formerly belonged to that disk group and have not been reused becomes shaded. When the view of the newly imported disk group is accessed, its VM disk icons are visible.

5.1.5 Adding a File System to an Existing Volume

File systems can be created and placed on existing volumes, one file system per volume. You must select one volume icon for this operation to succeed.

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select the volume on which to make the file system.**

3. **From the Basic-Ops menu, select File System Operations, then Make File System (see Figure 5-8).**

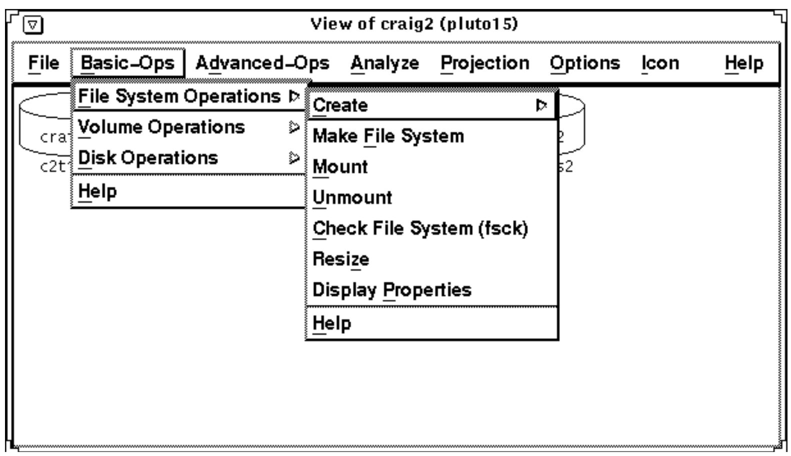

Figure 5-8 Accessing the File System Operations Menu

The Make File System form appears (see Figure 5-9). Most of the form fields are already set to the defaults for making a file system on an existing volume.

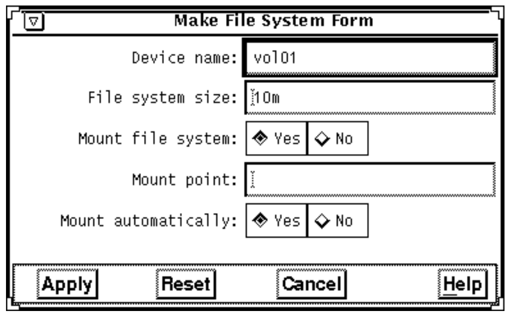

Figure 5-9 Make File System Form

4. Complete the Make File System form as follows:

Table 5-2 Make File System Form

Device name:	This field displays the block device on which to make the file system. This corresponds to the name of the selected volume, and cannot be changed.
File system size:	This field displays the length of the file system to be made. If no units are specified, sectors are assumed. This length should typically correspond to the length of the volume on which the file system is to be made, although it can be altered for special circumstances.
Mount file system:	Indicate whether the file system should be mounted after creation. If the answer is Yes (the default), a mount point must also be specified in the next field. All fields below this field are only accessible when Yes is specified here.
Mount point:	Enter the desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it.
Mount automatically:	Indicate whether this file system should be mounted every time the system comes up (by placing an entry in /etc/vfstab). Yes is the default.

5. When the form is properly completed, select Apply to create the file system.

If the file system is mounted, it is represented by the mount point, which appears below the volume.

5.1.6 Mounting a File System

You must mount a file system before you can access or create files. In situations where a file system exists on a volume but is not currently mounted, you can mount that file system at any time. You must select a volume icon containing a valid, unmounted file system for this operation to succeed.

Note – Volume Manager has no way of knowing whether a valid, unmounted file system already exists on a given volume. You are responsible for being aware of the existence of an unmounted file system on a volume, as well as that file system's characteristics.

---

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select the volume containing the file system to be mounted.**
3. **From the Basic-Ops menu, select File System Operations, then Mount (see Figure 5-8).**
   The Mount File System form appears (see Figure 5-10). Most of the form fields are already set to the defaults for mounting a file system on a volume.

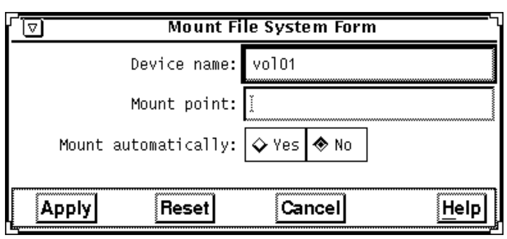

Figure 5-10 Mount File System Form

4. **Complete the Mount File System form as follows:**

Table 5-3 Mount File System Form

Device name:	This field displays the block device on which to make the file system. This corresponds to the name of the selected volume, and cannot be changed.
Mount point:	Enter the desired mount point for the file system. If the specified mount point does not already exist, Volume Manager will automatically create it. Volume Manager attempts to provide a default mount point, which it obtains by scanning /etc/vfstab.
Mount automatically:	Indicate whether this file system should be mounted every time the system comes up (by placing an entry in /etc/vfstab). No is the default.

5. **When the form is properly completed, select Apply to mount the file system.**
   When the file system is successfully mounted, it is represented by the mount point, which appears below the volume.

5.1.7 Adding a Mirror to a Volume

You mirror a volume to provide redundancy of data. Each plex on a mirrored volume duplicates the information stored on that volume.

You can add a mirror to an existing volume by associating another plex (of the correct length) to the volume. The layout of the mirror to be added can be either simple or striped. The number of available VM disks must be sufficient to accommodate the layout type of both the existing volume and the mirror to be added.

You must select a volume for this operation to succeed. You can select VM disks that are not already in use by the volume itself for this operation.

Note – If you are creating more than one mirror for a volume, you must wait until each mirror is completely made before you can create the next one. If a mirror is grayed out on the screen, it is in the process of being created; you must wait until it is no longer gray before you can create another mirror for that volume.

---

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select the volume to which a mirror is to be added by clicking LEFT on its icon.**

3. **Select the VM disk(s) to be used for the mirror, if necessary.**
   - If the mirror is to be striped, you must select more than one VM disk. Ensure that the VM disks have sufficient free space to accommodate the desired length of the new volume by clicking RIGHT on an unobscured portion of each VM disk icon to access its properties form, then checking the value in the Maximum free space: field. These VM disks must belong to the same disk group.
   - If you do not select any VM disk, disks with sufficient free space will automatically be used.

4. **From the Basic-Ops menu, select Volume Operations, then Add Mirror.**
   A submenu listing mirror layout options appears.

5. **Select either Simple Mirror or Striped Mirror, depending on the desired mirror layout.**
   The volume's icon expands visibly and a new mirror appears within its borders. The new mirror layout depends on what was specified during the Add Mirror operation.

Figure 5-11 illustrates a striped, mirrored volume.

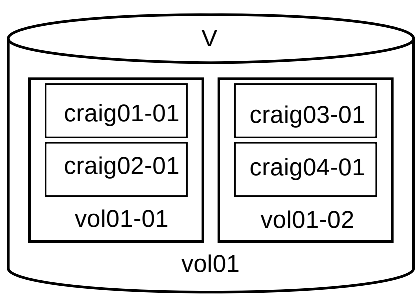

Figure 5-11 Mirrored Volume

5.1.8 Adding a Mirror to a Root Volume

When you select a volume for mirroring, Volume Manager automatically checks whether that volume is a root volume (containing bootable information). If it is, a dialog box appears and asks whether to make the new mirror bootable.

1. **Go to the View of rootdg.**
2. **Select the root volume to which a mirror is to be added by clicking LEFT on its icon.**
   In Volume Manager, a root volume appears as a normal volume icon (named `rootvol`, by default) with a mounted file system named `/`.
3. **Follow the steps for adding a mirror to a volume, as described in Section 5.1.7, "Adding a Mirror to a Volume."**
   When Volume Manager detects that the volume to be mirrored is a root volume, a dialog box appears and asks whether to make the new mirror bootable. Answer `Yes` to that question so that the mirror's disk can be used for booting if the root volume's original disk fails. If the root volume's mirror is made bootable, mirrors are also added to the volumes `swapvol` and `standvol` (but not to any other volumes that happen to have subdisks that reside on the root disk).

5.1.9 Attaching a Plex

A plex that has been detached (either automatically by Volume Manager or by you) can be reattached to its parent volume. This involves copying data from an active plex on the volume to the attaching plex. Depending on the amount of data to be copied, this operation may take some time. You can attach one or more plexes to a parent volume at a time. In order for a plex to be attached, it must already be detached and still associated with an enabled volume.

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select the (detached) plex to be attached by clicking the LEFT button on its icon.**
3. **From the Advanced-Ops menu, select Plex, then Attach Plexes.**
   When the attach process completes, the attached plex icon loses the color or shading that previously indicated that it was detached.

5.1.10 Creating a Plex

Plexes need to be created to connect subdisks to volumes. Without plexes, volumes are useless. The creation of a plex alone is useful only if one or more subdisks are associated with that plex. If one or more subdisks have been selected at the time of the plex creation, they will automatically be associated with the new plex.

To create a plex and associate it with a single subdisk:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Create a new subdisk, as described in Section 5.1.13, “Creating a Subdisk,” on page 5-20.**
3. **Drag the new subdisk icon (by pressing and continuing to hold the LEFT button) beyond the borders of its disk and into an open area of the view window. Drop the subdisk by releasing the LEFT button.**
   The Plex Create form appears (see Figure 5-12). When this form is invoked in this way, it will automatically create a new plex and associate the dragged subdisk with it. The Plex Create form establishes the parameters for the plex. Some of the fields already contain default values and others are optional.

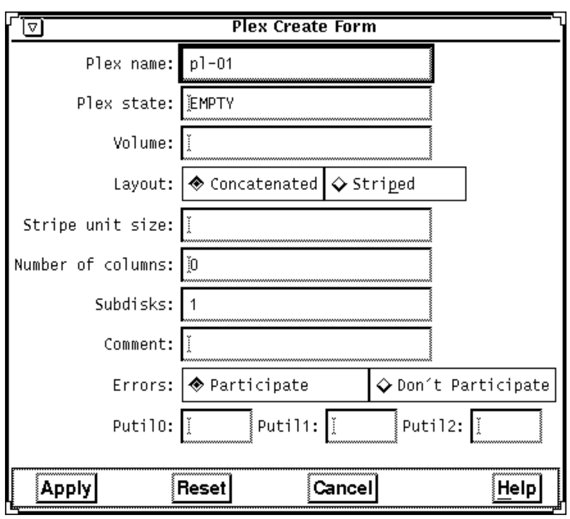

Figure 5-12 Plex Create Form

4. **Complete the Plex Create form according to the type of plex to be created.**
   Refer to this form's Help window for more information on any of its fields.

5. **When the form is properly completed, select Apply to activate both the plex creation and the association of subdisk and plex.**
   A plex icon containing a subdisk icon appears. The fact that the subdisk icon is within the plex icon confirms their association.

To create a plex and associate it with multiple subdisks:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Create the new subdisks, as described in Section 5.1.13, "Creating a Subdisk," on page 5-20.**

3. **Select the new subdisks by clicking the MIDDLE button on their icons.**

4. **From the Advanced-Ops menu, select Plex, then Create.**
   The Plex Create form appears (see Figure 5-12). When this form is invoked while subdisks are selected, it will automatically create a new plex and associate the selected subdisks with it. The Plex Create form establishes the parameters for the plex. Some of the fields already contain default values and others are optional.

5. **Complete the Plex Create form according to the type of plex to be created.**
   Refer to this form's Help window for more information on any of its fields.

6. **When the form is properly completed, select Apply to activate both the plex creation and the association of subdisks and plex.**
   A plex icon containing both subdisk icons appears. The fact that the subdisk icons are within the plex icon confirms their association.

5.1.11 Associating a Plex with a Volume

Once a plex has been created and has at least one associated subdisk, you can associate that plex with a volume. A volume is of little use until it has at least one associated plex. When multiple plexes are associated with a volume, mirroring is in effect. If the volume is already started and an additional plex is being associated, the Volume Manager will update the newer plex by copying all necessary data to that plex. This operation may take some time.

You can associate several plexes with a single volume at a time, but you can select only one volume per association operation.

There are a few methods of associating a plex with a volume:

- Selecting the volume and plex(es) and then using menus
- Dragging each plex icon onto the volume icon and then dropping the plex icon on that volume when the pointer is positioned over an unobscured portion of the volume icon
- Automatically associating one or more plexes with a volume by selecting the plex icon(s) immediately prior to the creation of a volume (see the section on volume operations for further details)

To associate a plex with an existing volume by using menus:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select both the plex and the volume to which it is to be associated by clicking the MIDDLE button on both of their icons.**
3. **From the Advanced-Ops menu, select Plex, then Associate Plexes.**
   The volume and plex icons disappear momentarily and then the plex icon reappears within the volume icon. The plex is now associated with the volume.

To associate multiple plexes with a single volume, select multiple plexes along with the volume and repeat the previous procedure.

5.1.12 Moving a Subdisk from One VM Disk to Another

The evacuation of subdisks involves moving all subdisks from one disk to another. This is useful when a disk appears to be failing and you want to protect any volumes that rely on that disk. This is also useful for performance tuning (by moving data off a busy disk to a less busy disk) and for general system administration housecleaning. This operation can only be performed between two disks in the same disk group.

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Drag the desired subdisk icon from one VM disk to a second disk and drop it there.**

A free subdisk identical to the original subdisk is created on the new disk and contents of the original subdisk are copied to the free subdisk. As the original subdisk is copied to its new disk, a temporary plex appears. When the transfer is complete, the original subdisk is removed.

You can also move a subdisk from one VM disk to another using the menus:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select the disk *from* which subdisks are to be evacuated by clicking LEFT on its icon.**
3. **From the Basic-Ops menu, select Disk Operations, then Evacuate Subdisks.**
   The Evacuate Subdisks form appears.

4. **Complete the Evacuate Subdisks form as follows:**

Table 5-4 Evacuate Subdisks Form

Disk group name:	Enter the name of the disk group to which both disks belong. Both disks must share the same disk group.
Evacuate From:	This field displays the name of the disk from which the subdisks are to be evacuated, which can be changed.
To:	Enter the name of the disk to which the subdisks are to be moved. This field is optional. However, the evacuated subdisks will be moved to one or more random disks if no target disk is specified.

5. **When the form is properly completed, select Apply to activate the subdisk evacuation.**
   The subdisks move from their original disk icon to the targeted disk icon.

5.1.13 Creating a Subdisk

Subdisks are the lowest-level building blocks in a volume and need to be created in order to build a volume. Subdisks are created on selected disks. Once created, subdisks can be associated with plexes to build volumes.

When building mirrored or striped volumes, take care to create subdisks on separate disks in order for striping or mirroring to work properly.

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Designate the disk on which to create the subdisk.**

3. **From the Advanced-Ops menu, select Subdisk, then Create.**
   The Subdisk Create form appears (see Figure 5-13). This form establishes the parameters for the subdisk. Some of the fields already contain default values and others are optional; only the subdisk length is required.

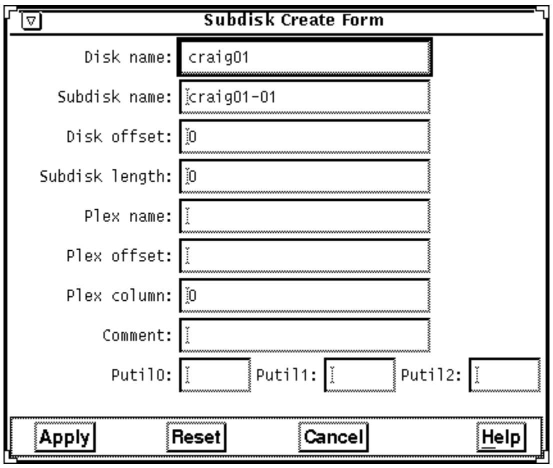

Figure 5-13 Subdisk Create Form

4. **Complete the Subdisk Create form, entering the subdisk length and optionally altering other fields.**

Table 5-5 Create Subdisk Form

Subdisk length:	Enter the length of the subdisk to be created. If no units are specified, the number is assumed to be in sectors. The maximum length of this field is 14 characters. The offset field entry also needs to be considered (and possibly adjusted) when selecting a length. The length and offset combination should not place this subdisk within the bounds of another subdisk on the disk or past the end of the disk.

Refer to this form's Help window for more information on any of its fields.

5. **When the form is properly completed, select Apply to activate the subdisk creation.**
   A new subdisk icon appears in the selected disk icon.

5.1.14 Creating a Log Subdisk

A subdisk can be associated with a plex as a log subdisk (with Dirty Region Logging in effect). A log subdisk is used to log recent disk activity. Only one log subdisk can exist on a given plex. For mirrored striped or concatenated volumes, this will add a Dirty Region Log; for RAID-5 volumes, this will add a RAID-5 log.

Note – Logs are only useful for mirrored volumes or RAID-5 volumes.

To create a log subdisk for a new volume:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Create a volume using the instructions given in Chapter 4, "Setting Up the Software Configuration."**
3. **In the Volume/FS Create form, locate the Create log subdisk line and click LEFT on Yes.**

To create a log subdisk for an existing volume:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select a mirrored or RAID-5 volume by clicking LEFT on the icon.**
3. **From the Basic-Ops menu, select Volume Operations, then Add Log.**
   A log associated with a subdisk will appear in the volume icon. A DRL log subdisk for a mirrored volume can be identified by its double border, unlike RAID-5 logs.

5.1.15 Associating a Subdisk

Once a subdisk has been created, it can be associated with a plex, which can then be used to build a volume. It is not possible to read or write to lone subdisks.

Several subdisks can be associated with a single plex at a time, but only one plex can be selected per association operation. When a subdisk is associated with a plex that is already associated with a volume, the plex length is automatically increased by the subdisk length, but the volume length is not updated.

There are a few methods of associating a subdisk with a plex:

- Selecting the plex and subdisk(s) and then using menus
- Dragging each subdisk icon onto the plex icon and then dropping the subdisk icon on that plex when the pointer (in the image of a hand) is positioned over an unobscured portion of the plex icon
- Dragging a subdisk icon onto a gap (or "hole" formed by the removal of a subdisk) between two subdisks on a plex and dropping the subdisk there. If there is not sufficient space in the hole for the new subdisk, an error will result.
- Automatically associating subdisks with a plex by selecting the subdisk icon(s) immediately prior to the creation of a plex (see the section on plex operations for further details)

To associate a subdisk with an existing plex by using menus:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select both the subdisk and the plex to which it is to be associated by clicking the MIDDLE button on both of their icons.**
3. **From the Advanced-Ops menu, select Subdisk, then Associate Subdisks.**
   The subdisk icon appears within the plex icon. The subdisk is now associated with the plex.

To associate multiple subdisks with a single plex, select multiple subdisks along with the plex and repeat the previous procedure.

5.2 Removing Components from a Configuration

5.2.1 Removing a VM Disk From a Disk Group

You can remove a VM disk from a disk group when it is no longer necessary for the underlying physical disk to be controlled by the Volume Manager. You must remove any subdisk residing on the VM disk that is assigned to that physical disk or move the subdisk to another VM disk before the VM disk can be removed from its disk group.

Note – If you want to remove the last VM disk in a disk group, the disk group itself must be deported using the procedures in Section 5.2.3, “Deporting a Disk Group.” This will remove the disk group *and* the last VM disk in that disk group.

---

1. **Click LEFT over the screen button in the root window for the view you want to access (see Figure 5-1).**
2. **Select the VM disk to be removed by clicking the LEFT button on its icon or select multiple disks by clicking RIGHT on their icons.**
   The VM disk cannot contain any subdisks.
3. **From the Advanced-Ops menu, select Disk Group, then Remove Disks (see Figure 5-6).**
   The disk(s) will disappear from the screen.

Note – This procedure removes a VM disk from a disk group only until the next reboot or re-initialization, at which time it will be brought back under Volume Manager control because the private region still exists on the physical disk. If you want to completely remove the disk from Volume Manager control, you must relabel the physical disk and remove the private region.

5.2.2 Unmounting a File System

File system administration often requires the unmounting of file systems. You can unmount a file system from a volume as long as the mount point is not currently in use. You must select a volume icon containing a mounted file system for this operation to succeed.

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select the volume whose file system is to be unmounted.**
3. **From the Basic-Ops menu, select File System Operations, then Unmount (see Figure 5-8).**
   The mount point should disappear from beneath the volume icon, indicating that the unmount operation has succeeded. If the mount point is in use, the unmount will fail.

5.2.3 Deporting a Disk Group

Note – The disk group `rootdg` cannot be deported.

Deporting a disk group does not actually remove a disk group; it disables access to that disk group. You can deport a disk group so that its last VM disk can be removed, thus allowing the physical disk assigned to that VM disk to be reused, reinitialized, or added to other disk group. Once deported, the disk group is inaccessible and any VM disk icons that belonged to that disk group disappear along with the disk group view. However, the partition retains its disk access record and knowledge of the deported disk group until it is reused (assigned to another disk group) or removed. A deported disk group can be re-imported later to the same host system or can be imported by a different host system running the Volume Manager.

1. **Click LEFT over the screen button in the root window for the view you want to access (see Figure 5-1).**
2. **From the Advanced-Ops menu, select Disk Group, then Deport Disk Group (see Figure 5-6).**
   The Deport Disk Group form appears (see Figure 5-14).

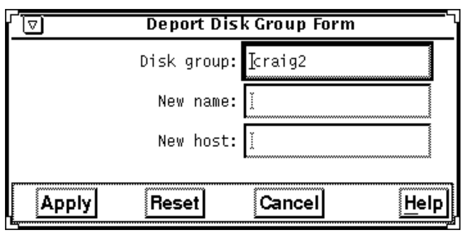

Figure 5-14 Deport Disk Group Form

3. **Complete the Deport Disk Group form.**
   Refer to Table 5-6 for an explanation of the fields in the Deport Disk Group form. The Deport Disk Group form should already display the current disk group name as the disk group to be deported. If this is correct, select Apply to deport the disk group.

Table 5-6 Deport Disk Group Form

Field	Description
Disk group:	The name of the disk group to be deported and made inaccessible to the system. This must be a valid disk group.
New name:	This will become the disk group's new name after it is deported. This field does not contain a default value. If you do not fill in this field, the existing name will remain the same.
New host:	This will set the disk group's host as part of the deport operation. This field does not contain a default value. If you do not fill in this field, the host name will remain unchanged. This field, along with the `New name` field, are useful for fixing a `rootdg` on another machine that shares disks with the current host.

The disk group view disappears, along with any VM disks that it contained. The view button corresponding to the deported disk group also disappears from the root window.

5.2.4 Removing a Volume

You can remove volumes when they are no longer needed or the disk space that they occupy needs to be reused.

The removal of a volume involves the removal of plex and subdisk components. With the basic approach to volume removal, the underlying plexes and subdisks are removed automatically. You indicate which volume is to be removed; the volume and its components are then removed and the disk space set aside for that volume is de-allocated. You must select a volume for this operation to succeed.

Note – This is a permanent operation and cannot be undone. If completed, it will be difficult or impossible for you to retrieve the data associated with that volume. For this reason, a confirmation window is presented if the selected volume is not ready for removal.

You can remove a volume using either a *basic* approach or an *advanced* approach:

Basic	Takes information about what you want to accomplish and then performs the necessary underlying tasks. This approach requires only minimal input from the user, but permits more detailed specifications.
Advanced	Consists of fairly complicated commands that typically require you to specify detailed input. These commands use a “building block” approach that requires you to have detailed knowledge of the underlying structure and components in order to manually perform the sequences of commands necessary to accomplish a certain task.

5.2.4.1 Basic Approach for Removing a Volume

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select the volume to be removed by clicking LEFT on its icon.**

3. **From the Basic-Ops menu, select Volume Operations, then Remove Recursively.**
   If the selected volume is enabled and therefore in danger of losing valuable data, a Volume Manager Warning window appears to announce this fact. The warning window requires input before Volume Manager will proceed with anything (see Figure 5-15).

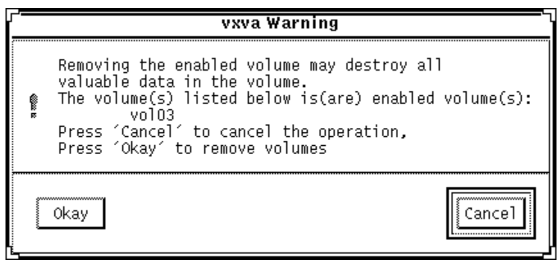

Figure 5-15 Volume Removal Warning

   If you are certain that the data on the selected volume does not need to be preserved, select `Okay` to proceed with the removal; otherwise, select `Cancel` to abandon the removal. The volume disappears when the removal is complete.

5.2.4.2 *Advanced Approach for Removing a Volume*

When you remove a volume using the Advanced-Ops menu, its plex and subdisk components are not automatically removed; you must also manually remove each of its components. It is generally not a good idea to leave the subdisks of a removed volume behind, as the space occupied by these subdisks cannot be used by the free space pool until they are removed.

If you want to remove a volume instantly and recursively, the easiest approach is to use the Basic-Ops menu. However, the following procedure describes how to manually remove a volume and some or all of its components.

To remove a volume with a single plex and subdisk:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Stop the volume as described in Section 5.4.2, "Starting and Stopping Volumes," on page 5-43.**

3. **Select the volume to be removed by clicking the LEFT button on its icon.**

4. **From the Advanced-Ops menu, select Volume, then Remove Volumes.**
   The volume icon momentarily shrinks and appears next to its plex icon, then disappears entirely. The volume is now removed, but its plex and subdisk remain.

To remove multiple volumes simultaneously, follow the previous steps, but use the MIDDLE button to select multiple volumes for removal at once.

To remove a removed volume's plex:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select the plex to be removed by clicking the LEFT button on its icon.**

3. **From the Advanced-Ops menu, select Plex, then Remove Plexes.**
   The plex icon momentarily shrinks and appears next to its previous position, then disappears entirely along with the subdisk icon. The plex is now removed, but its subdisk remains on its disk and is now categorized as a free subdisk.

To remove multiple plexes simultaneously, follow the previous steps, but use the MIDDLE button to select multiple plexes for removal at once.

To remove a removed volume and plex's subdisk:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Identify the subdisk icon that formerly belonged to the removed plex and volume.**
   If projection is set to show free subdisks (through the Projection menu), then this subdisk icon should be highlighted on its disk.

3. **Select the subdisk to be removed by clicking the LEFT button on its icon.**

4. **From the Advanced-Ops menu, select Subdisk, then Remove Subdisks.**
   The subdisk icon disappears.

To remove multiple subdisks simultaneously, follow the previous steps, but use the MIDDLE button to select multiple subdisks for removal at once.

5.2.5 Removing a Mirror or Log

A mirror or log can be removed recursively, which involves automatically removing any subdisks associated with that plex. Mirrors and logs that are associated with volumes, as well as those that are not, can be removed in this way. Space formerly allocated to the removed subdisks is returned to the free space pool.

Note – The last valid mirror in a started or enabled volume cannot be removed, unless the volume is stopped before the remove operation.

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select the mirror or log to be removed by clicking LEFT on its icon.**
3. **From the Basic-Ops menu, select Volume Operations, then Remove Mirror or Log.**

The mirror or log and any associated subdisk icons disappear, indicating that these objects have been removed. Though a RAID-5 log can be removed using the same procedure, you should keep at least one log associated with a RAID-5 volume to help cope with disk failures.

5.2.6 Dissociating a Plex

Associated plexes can be dissociated from their parent volumes. However, the last plex in a started volume cannot be dissociated until that volume is stopped. A plex can be dissociated if it is no longer required to reflect a copy of a volume. Plexes must be dissociated before they can be removed.

There are a couple of methods of dissociating a plex from a volume:

- Selecting the plex(es) and then using menus.
- Dragging each plex icon beyond the borders of the volume icon and then dropping the plex in an open area of the view window.

To dissociate a plex from its volume using menus:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select the plex to be dissociated from its volume by clicking the LEFT button on its icon.**

3. **From the Advanced-Ops menu, select Plex, then Dissociate Plexes.**
   The plex icon disappears momentarily and then reappears beyond the borders of the volume icon. The plex is now dissociated from its volume.

To dissociate multiple plexes from a volume, select multiple plexes (using the MIDDLE button) and repeat this procedure.

5.2.7 Detaching a Plex

A detached plex is inaccessible for reads and writes, but is still associated with a volume. One or more plexes can be detached from their parent volume at a time. The volume must be started before a plex can be detached.

Note – This operation is not permitted when the specified plex is the last valid plex on the volume.

---

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select the plex to be detached by clicking the LEFT button on its icon.**
   To detach multiple plexes, select multiple plexes with the MIDDLE button. The selected plex cannot be the last plex in the volume.

3. **From the Advanced-Ops menu, select Plex, then Detach Plexes.**
   The plex icon changes color or pattern to indicate that it is now detached. Figure 5-16 illustrates a detached plex (`pl-02`).

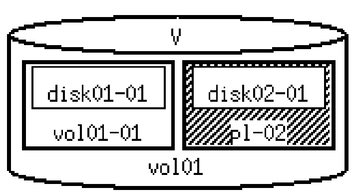

Figure 5-16 Detached Plex

This command temporarily detaches the plex, but maintains the association between the plex and its volume; however, the plex will not be used for I/O. A plex detached using the preceding instructions will be recovered on a system reboot.

5.2.8 Removing a Plex

You can remove a plex if it is no longer required for mirroring or if the space that it occupies needs to be reused. You can remove a plex recursively (in which case any associated subdisks are automatically removed) or manually (in which case its subdisks are left behind).

The Basic-Ops menu provides an option that dissociates a plex if it is still associated with a volume, then recursively removes that plex and its subdisks.

When you remove a plex using the Advanced-Ops menu, its subdisk components are not automatically removed; you must also manually remove these subdisks once the plex is gone. It is generally not a good idea to leave the subdisks of a removed plex behind, as the space occupied by these subdisks cannot be used by the Volume Manager's free space pool until they are removed.

To remove a plex:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select the plex to be removed by clicking the LEFT button on its icon.**
   The selected plex can be either associated or dissociated.
3. **From the Basic-Ops menu, select Volume Operations, then Remove Mirror.**
   If the plex is associated, it is automatically dissociated. The plex icon then disappears entirely, along with any associated subdisk icons.

To remove a plex (but not its subdisks):

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **If the plex to be removed is still associated with its volume, dissociate the plex as described in the section on dissociating plexes.**
3. **Select the plex to be removed by clicking the LEFT button on its icon.**

4. **From the Advanced-Ops menu, select Plex, then Remove Plexes.** The plex icon momentarily shrinks and appears next to its previous position, then disappears entirely along with the subdisk icon. The plex is now removed, but its subdisk remains on its disk and is now categorized as a free subdisk.

To remove multiple plexes simultaneously, follow the previous steps, but use the MIDDLE button to select multiple plexes for removal at once.

5.2.9 Dissociating a Subdisk

Associated subdisks can be dissociated from their parent plexes. A subdisk becomes a free subdisk when it is dissociated. A subdisk must be dissociated before it can be removed.

There are two methods of dissociating a subdisk from a plex:

- Selecting the subdisk(s) and then using menus
- Dragging the subdisk icon beyond the borders of the plex icon and then dropping the subdisk in an open area of the view window

To dissociate a subdisk from its plex by using menus:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select the subdisk to be dissociated from its plex by clicking the LEFT button on its icon.**
3. **From the Advanced-Ops menu, select Subdisk, then Dissociate Subdisks.** The subdisk icon disappears from its plex icon and is only visible in its VM disk icon. The subdisk is now dissociated from its plex.

To dissociate multiple subdisks from a plex, select multiple subdisks (using the MIDDLE button) and repeat the previous procedure.

5.2.10 Removing a Subdisk

You can remove a subdisk in order to free its space for reuse. Selected subdisks can be permanently removed. Once you have dissociated a subdisk from a plex, it is generally better to remove that subdisk than to allow it to continue to consume disk space. The space occupied by free subdisks cannot be used by the Volume Manager's free space pool until those subdisks are removed.

Note – You cannot remove a subdisk that is associated with a plex. You must dissociate the subdisk from its plex beforehand.

---

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **If the subdisk to be removed is still associated with its plex, dissociate the subdisk as described in Section 5.2.9, "Dissociating a Subdisk," on page 5-32.**
3. **Identify the subdisk icon to be removed.**
   Since the subdisk is no longer associated with a plex, its icon should now exist only on its disk. If projection is set to show free subdisks (using the Projection menu), then this subdisk icon should be highlighted on its disk.
4. **Select the subdisk to be removed by clicking the LEFT button on its icon.**
5. **From the Advanced-Ops menu, select Subdisk, then Remove Subdisks.**
   The subdisk icon disappears, leaving a gap or open area on its disk.

To remove multiple subdisks simultaneously, follow the previous steps, but use the MIDDLE button to select multiple subdisks for removal at once.

5.3 Resizing the Components of a Configuration

5.3.1 Resizing a File System

You can resize a file system on a selected volume. Resizing allows a file system to grow according to current needs. You can increase a file system in size if more space is required.

The resize operation involves resizing both the file system and its underlying volume. If new disk space is needed during the resize, it is allocated as necessary; if space becomes unused, it is added to the free space pool. You must select a volume icon containing a mounted file system in order for this operation to succeed. Resizing can take one of two forms:

- Increase the volume and file system *to* a given length.
- Increase the volume and file system *by* a given length.

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select the volume whose file system is to be resized.**

3. **From the Basic-Ops menu, select File System Operations, then Resize.**
   The File System Resize form appears. Most of the form fields are already set to the defaults for resizing a file system on a volume.

4. **Complete the File System Resize form as follows:**

Table 5-7 File System Resize Form

Mount point:	This field displays the mount point for the file system to be resized.
Volume:	This field contains the block device of the volume on which the file system resides.
Current size:	This field displays the current size of the file system to be resized.
Option:	Select the type of resize operation to be performed. This will determine whether the file system is grown or shrunk to a certain size, or grown or shrunk by a given amount.
Size/Amount:	Enter either the length to which or the amount by which the file system is to be resized. If Grow To is selected, this field should reflect the final size. If Grow By is selected, this field should reflect the amount by which the size should change.

5. **When the form is properly completed, select Apply to resize the file system.**
   As the file system and underlying volume are resized, the corresponding volume icon may change visibly.

5.3.2 Resizing a Volume

You can resize an existing volume if necessary. Resizing allows a volume to grow or shrink according to current needs. You can increase the volume size if a file system or application on the volume requires additional space.

If new disk space is needed during the resize, it is allocated as necessary; if space becomes unused, it is added to the free space pool. You must select a volume icon in order for this operation to succeed.

When you shrink a volume, be aware that the volume may contain data that could be lost. Volume Manager does not permit a volume containing a mounted file system to be shrunk using the volume resize operation offered through the Basic-Ops menu. This prevents the possibility of losing the contents of the end of a file system whose volume has been shrunk.

If a file system resides on a volume, you should use the resize operation offered through the File System Operations menu to adjust both the file system and the underlying volume. Resizing can take one of four forms:

- Increase the volume *to* a given length.
- Increase the volume *by* a given length.
- Reduce the volume *to* a given length.
- Reduce the volume *by* a given length.

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select the volume (that doesn't have a mounted file system) to be resized.**

3. **From the Basic-Ops menu, select Volume Operations, then Resize.**
   The Volume Resize form appears (see Figure 5-17). Most of the form fields are set to the defaults for resizing a volume.

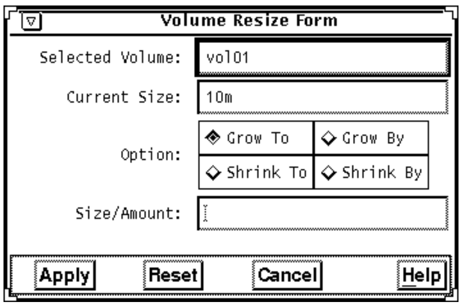

Figure 5-17 Volume Resize Form

4. Complete the Volume Resize form as follows:

Selected Volume:	This field displays the name of the volume to be resized. This field cannot be changed.
Current Size:	This field displays the current size of the volume to be resized. This field cannot be changed.
Option:	Select the type of resize operation to be performed. This will determine whether the volume is grown or shrunk to a certain size, or grown or shrunk by a given amount.
Size/Amount	Enter either the length to which, or the amount by which, the volume is to be resized. If Grow To or Shrink To is selected, this field should reflect the final size. If Grow By or Shrink By is selected, this field should reflect the amount by which the size should change.

5. When the form is properly completed, select Apply to activate the resize operation.

As the volume is resized, the corresponding volume icon may change visibly.

5.3.3 Splitting Subdisks

In some situations, it is useful to be able to split subdisks into a series of smaller pieces. This makes it possible to more accurately identify those areas of a disk where hot spots exist. Analysis can be used to identify any subdisk areas associated with high activity (for more information on analysis, see Section 6.5, “Displaying Activity Levels on Objects,” on page 6-11). The subdisks occupying the hot spot location can then be moved to other disks for improved overall performance. Contiguous subdisks that were not occupying hot spots can later be rejoined to form larger subdisks.

Note – The log subdisks and subdisks associated with striped plexes cannot be split, moved, or joined.

---

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select the subdisk to be split by clicking LEFT on its icon.**
3. **From the Advanced-Ops menu, select Subdisk.**
4. **Select either Into 2 Subdisks or Into More Than 2 Subdisks.**
   The Subdisk Split form appears.
5. **Complete the Subdisk Split form, specifying the number of new subdisks that you want the subdisk split into.**
6. **When the form is properly completed, select Apply to activate the subdisk split.**
   The subdisk icon is converted into the number of subdisk icons that you specified.

You can now perform analysis on the resulting subdisks (using the Analyze menu) to determine which of those subdisks is the hottest spot.

Once a subdisk is identified as a hot spot, it can be moved to another disk.

5.3.4 Joining Subdisks

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select both subdisks by clicking the MIDDLE button on their icons within their disk icon.**
   The subdisks must be adjacent on a single disk in order for joining to occur.
3. **From the Advanced-Ops menu, select Subdisk, then Join subdisks.**
   The subdisk icons rewrite themselves on their disk and reappear as a single subdisk icon. The subdisks are now joined. The remaining subdisk retains the name of the topmost original subdisk.

5.4 Stopping and Starting Components of a Configuration

5.4.1 Spinning Up and Spinning Down Disk Drives

You can spin up or spin down the physical disk drives in your SPARCstorage Array at three levels:

- Spinning Up and Spinning Down All Drives in an Array — page 5-38
- Spinning Up and Spinning Down All Drives in a Disk Tray — page 5-40
- Spinning Up and Spinning Down Individual Disk Drives — page 5-41

5.4.1.1 Spinning Up and Spinning Down All Drives in an Array

Spinning Up All Drives in a SPARCstorage Array

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 5-1).**
   This will bring up the View of SPARCstorage Array window.
2. **From the Commands menu, select Controller, then Start All Drives (see Figure 5-18).**

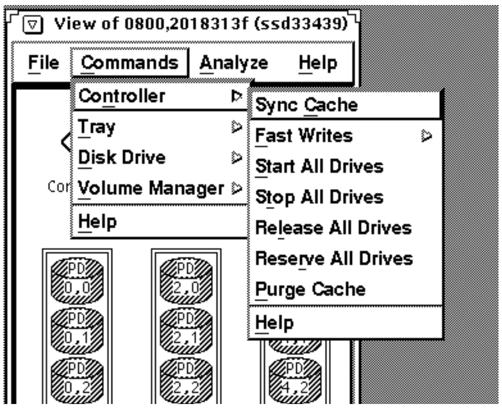

Figure 5-18 Accessing the Controller Menu from the Commands Menu

All the drives in the window will lose the asterisk (*) next to the PD icon as they start.

Spinning Down All Drives in a SPARCstorage Array

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 5-1).**
   This will bring up the View of SPARCstorage Arrays window.

2. **From the Commands menu, select Controller, then Stop All Drives (see Figure 5-18).**
   You will see the following confirmation window:

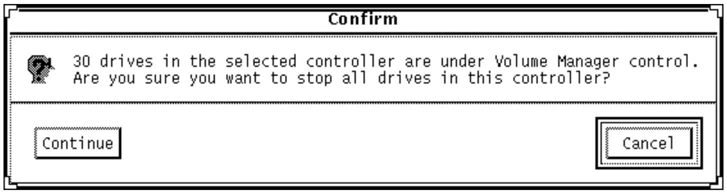

Figure 5-19 Stop Drives Confirmation Window

3. **Click LEFT over the Continue screen button.**
   All the drives in the window will show an asterisk (*) next to the PD icon as they stop.

5.4.1.2 Spinning Up and Spinning Down All Drives in a Disk Tray

Spinning Up All Drives in a Disk Tray

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 5-1).**
   This will bring up the View of SPARCstorage Array window.

2. **Determine which drive tray you want to start the drives in and click LEFT over that tray icon (see Figure 5-20).**

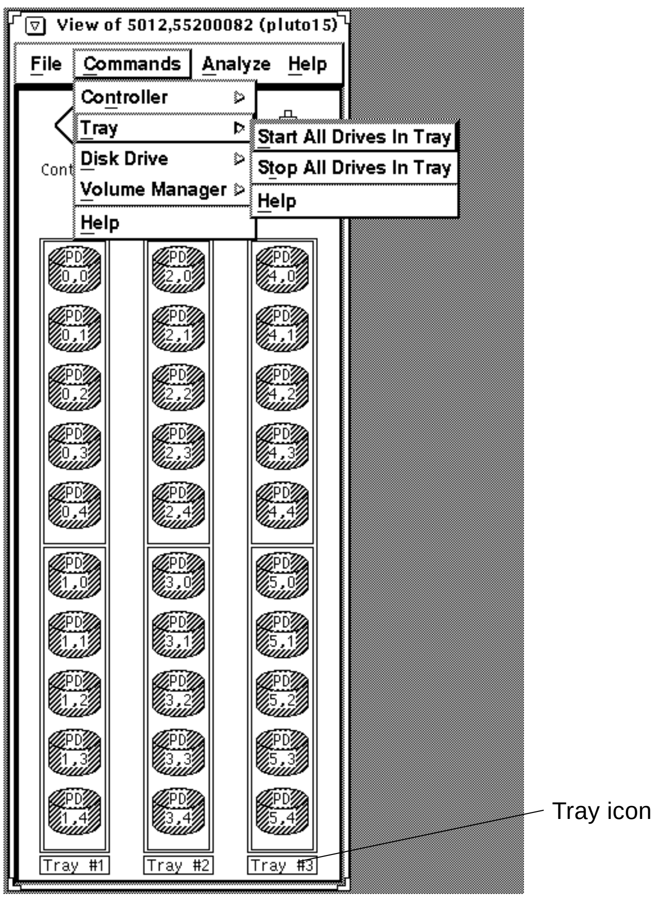

Figure 5-20 Accessing the Tray Menu from the Commands Menu

3. **From the Commands menu, select Tray, then Start All Drives In Tray (see Figure 5-20).**
   All the drives in the tray will lose the asterisk (*) next to the PD icon as they start.

Spinning Down All Drives in a Disk Tray

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 5-1).**
   This will bring up the View of SPARCstorage Array window.
2. **Determine which drive tray you want to stop the drives in and click LEFT over that tray icon (see Figure 5-20).**
3. **From the Commands menu, select Tray, then Stop All Drives In Tray (see Figure 5-20).**
4. **Click LEFT over the Continue screen button.**
   All the drives in the tray will show an asterisk (*) next to the PD icon as they stop.

5.4.1.3 Spinning Up and Spinning Down Individual Disk Drives

Spinning Up Individual Disk Drives

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 5-1).**
   This will bring up the View of SPARCstorage Array window.
2. **Determine which disk drive you want to start and click LEFT over that disk drive icon (see Figure 5-21).**

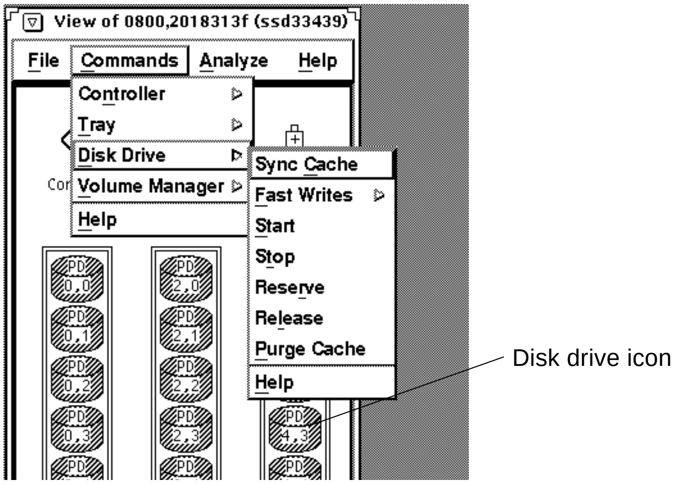

Figure 5-21 Accessing the Disk Drive Menu from the Commands Menu

3. **From the Commands menu, select Disk Drive, then Start (see Figure 5-21).**
   The drive will lose the asterisk (*) next to the PD icon when it starts.

Spinning Down Individual Disk Drives

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 5-1).**
   This will bring up the View of SPARCstorage Array window.

2. **Determine which disk drive you want to stop and click LEFT over that disk drive icon (see Figure 5-21).**

3. **From the Commands menu, select Disk Drive, then Stop (see Figure 5-22).**
   You will see the following confirmation window:

Figure 5-22 Stop Drives Confirmation Window

4. **Click LEFT over the Continue screen button.**
   The drives will show an asterisk (*) next to the PD icon when it stops.

5.4.2 Starting and Stopping Volumes

At times, it may be necessary to start or stop a volume. Starting involves enabling a disabled volume, while stopping involves disabling an enabled one. Volume Manager sometimes notifies you that a volume needs to be started before a desired operation can be performed.

Volumes can switch between the following states:

Table 5-8 Volume States

Started	The volume is started and functioning.
Unstartable	The volume has never been started.
Startable	The volume is now stopped, but was once started.

You should initialize volumes before you start them. Volumes must have been started before they can be stopped.

You can start or stop volumes all at one time or individually:

- Starting and Stopping All Volumes — page 5-43
- Starting and Stopping One Volume — page 5-44

5.4.2.1 Starting and Stopping All Volumes

Starting All Volumes

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **From the Advanced-Ops menu, select Volume, then Start Volumes, and then Start All.**
   All unstarted volume icons should rewrite themselves as they start. Volumes that were already started are not affected.

3. **Check the properties forms of any of the newly started volumes to confirm that they are indeed started.**

Stopping All Volumes

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **From the Advanced-Ops menu, select Volume, then Stop Volumes, and then Stop All.**
   All started volume icons should rewrite themselves as they stop. Volumes that were already stopped are not affected.

3. **Check the Utility State: field in the properties forms of any of the newly stopped volumes to confirm that they are indeed startable.**

5.4.2.2 Starting and Stopping One Volume

Starting One Volume

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Before starting the volume, confirm that it is not started.**
   To do this, access its properties form by clicking RIGHT on its icon. The Utility State: field should indicate that the volume is unstartable (never been started) or startable (currently unstarted).

3. **Select the volume to be started by clicking the LEFT button on its icon.**

4. **From the Advanced-Ops menu, select Volume, then Start Volumes, and then Start.**
   The volume icon should rewrite itself as it starts.

5. **Confirm that the volume has now been started by accessing its properties form again.**
   The Utility State: field should now indicate that the volume is started.

Stopping One Volume

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Before stopping the volume, confirm that it is started.**
   To do this, access its properties form by clicking RIGHT on its icon. The Utility State: field should indicate that the volume is started.

3. **Select the volume to be stopped by clicking the LEFT button on its icon.**

4. **From the Advanced-Ops menu, select Volume, then Stop Volumes, and then Stop.**
   The volume icon should rewrite itself as it stops.

5. **Confirm that the volume has now been stopped by accessing its properties form again.**
   The Utility State: field should now indicate that the volume is startable.

5.5 Changing Components of a Configuration

5.5.1 Changing Ownership and Permissions

If you want to change the ownership or permissions on a file or disk that is under Volume Manager control, you cannot use the `chgrp`, `chown` or `chmod` commands. However, you can change the ownership or permissions on the corresponding volume using Volume Manager:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

2. **Select the volume that you want to change ownership and/or permissions on and click the RIGHT mouse button over that volume icon.**
   This will bring up the Volume Properties form (see Figure 5-23).

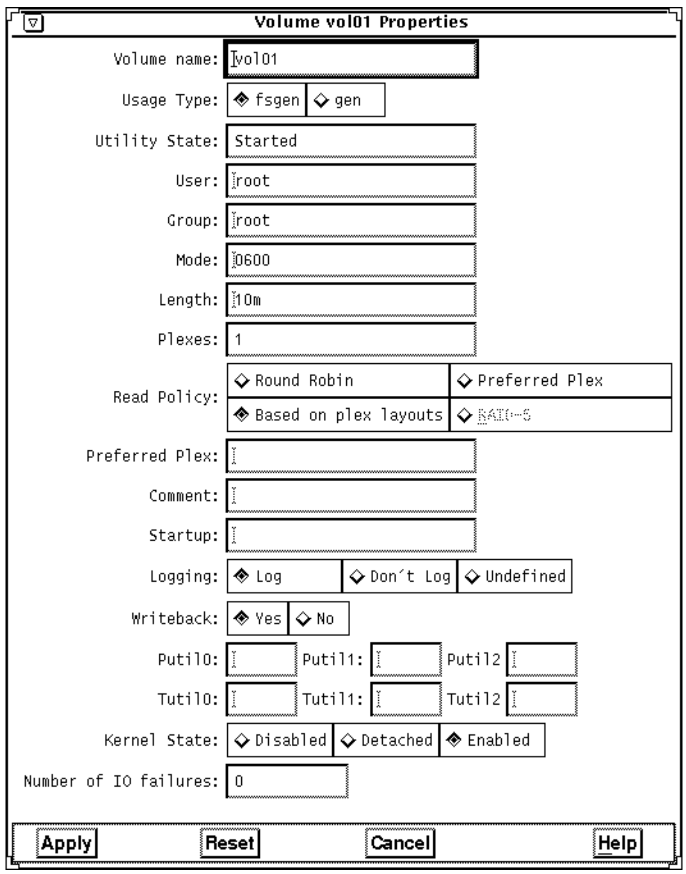

Figure 5-23 Volume Properties Form

3. **In the User: field, enter the name of the user who will own this volume.**
   This must be a valid user name on the system. The maximum length of the field is 14 characters.

4. **In the Group: field, enter the name of the group that will own this volume.**
   This must be a valid group name on the system. The maximum length of the field is 14 characters.

5. **In the Mode: field, enter the permissions mode for the volume in the same format as you would if you were changing the permissions using the `chmod` command.**
   The maximum length of this field is 4 characters. For example, to change the permissions mode to allow for only writes and only by the owner, you would enter *0200* in this field.

6. **Click the LEFT mouse button over the Apply screen button to set the ownership and permissions.**

5.5.2 Changing Names of Objects

1. **Move the pointer over the icon and click the RIGHT button (if the icon happens to be under analysis, you must use Shift-RIGHT instead).**
   This opens the properties window for that object.

2. **Alter the name field for that object.**

3. **Click the LEFT button on the Apply screen button at the bottom of the form.**
   The icon corresponding to the renamed object will adjust its name accordingly.

5.5.3 Changing the Properties of Icons

You can access a set of icon-related options using the Icon menu located in the menu bar of each view window (except the View of SPARCstorage Array window). Table 5-9 gives a summary of the icon-related options available through this menu:

Table 5-9 Icon-Related Options

Icon	Description
Maximize Icons	Maximizes the selected minimized icon(s), so that it shows all of its sub-icons.
Minimize Icons	Minimizes the selected icon(s), so that it shrinks down in size and hides all of its sub-icons.

Table 5-9 Icon-Related Options (Continued)

Icon	Description
Maximize All Icons	Maximizes all icons in the current view window at once.
Create Icons	Creates a copy of the icon(s) selected from another view and places the icon copy in the current user-created view.
Remove Icons	Removes the selected icon(s) from the current user-created view.

5.5.4 Changing Options

Most Volume Manager windows contain an Options menu, which allows you to specify preferences on how the Visual Administrator should operate.

♦ **Click the LEFT mouse button on Options in a window's menu bar within a window displaying this option.**
Some of the preferences listed have associated cascading menus that require further selection.

User preferences are saved in the file `.vxva_pref`, which is automatically created in your home directory. Each user has a personal `.vxva_pref` file.

The preferences that can be set through the Options menu are:

- Show Command
- When Commands are Ready
- Logging
- Popup the Command Window
- Format of Size
- Help

Some of the available preference settings relate to the Command Info Window. This is a special window that displays the command history, along with the status and results of those commands.

5.5.4.1 Show Command

This specifies whether the Command Info Window is to be shown before every command is executed (Show at Start), or only when a command fails (Show on Error). If no preference is indicated, the default is Show on Error.

5.5.4.2 When Commands are Ready

This specifies what the Visual Administrator should do when it is ready to run a Volume Manager or system command. The command can either be run immediately (Execute Commands), or brought up in the Command Info Window for inspection (Show Commands Only). If Show Commands Only is selected, the command can be executed directly from the Command Info Window once it is approved. If no preference is indicated, the default is Execute Commands.

5.5.4.3 Logging

This indicates whether commands are to be logged to a file. Logging keeps a record on disk of all commands sent to the Volume Manager or the system by the Visual Administrator. Logging can be started (Start) or stopped (Stop) at any time. When started, a window requesting a log file name appears and a file name must be entered; the `Apply` button must then be selected to start logging with the designated file. If no preference is indicated, the default is Stop and no command logging is in effect.

5.5.4.4 Popup the Command Window

This brings up the Command Info Window on demand. This window displays the command history, along with the status and results of those commands. Commands can also be executed or repeated from this window. The Command Info Window is discussed in more detail in Section 6.6 on page 6-18.

5.5.4.5 Format of Size

This specifies the units in which size-related output should be displayed (megabytes, kilobytes, or sectors). The unit of size applies to output only and is set to megabytes until the user resets it.

In properties forms, length values are displayed as a number followed by an `s`, `m`, or `k` (representing sectors, megabytes, or kilobytes, respectively). If the size cannot be cleanly converted into megabyte or kilobyte format, it is displayed in sectors instead.

The preferred format of size does not apply to input. Input typically defaults to sectors, unless megabytes or kilobytes are specified.

Maintaining the Data 6

This chapter gives the procedures for maintaining the data for your system:

6.1 Replacing Physical Disks

Disks most often need replacing when they fail or start to behave strangely. Replacing an existing disk with a new one (if the disk has failed, for instance) involves a set of procedures including the following:

- initializing, analyzing, and partitioning the raw disk
- initializing the new disk for Volume Manager use (installing header and configuration information on the disk)
- replacing the old disk with the new one (connecting the existing disk media record to the new disk)

These underlying steps are handled automatically when the disk replacement is performed through the GUI's Basic-Ops menu.

Note – This operation alters the partitioning of a disk and should therefore be used with caution.

The disk replacement procedure outlined here can only be performed on a disk that has failed. The GUI identifies such a disk by altering the disk icon's color or pattern when failure occurs.

Replace a VM disk as follows:

1. **Go to the View window corresponding to the disk group containing the VM disk to be replaced.**
   Select the failed disk by clicking the LEFT button in its icon. The GUI should portray the failed disk icon in a different color or pattern.

2. **From the Basic-Ops menu, select Disk Operations, then Replace Disks.**
   The Replace Disks form appears.

3. **Complete the Replace Disks Form.**

Table 6-1 Replace Disks Form

Field	Description
Old VM disk name:	Displays the name of the failed VM disk in this disk group. It is read only and cannot be changed.
New physical disk name:	Enter the name of the new physical disk that is to replace the existing one. The name should be in `c#t#d#` form and must be unique.

4. **When the form is properly completed, select Apply to activate the disk replacement.**
   If the replacement is successful, the VM disk icon associated with the failed physical disk returns to a normal state (the special icon coloring or pattern goes away). In the View of Disks window, a physical disk icon with a new name and a new partition appears.

6.2 Moving Data from a Failed Physical Disk to a Different Physical Disk

If a physical disk fails or starts to behave strangely, you can move the data from the failed physical disk to a different physical disk as follows:

1. **Determine which physical disk has failed.**
   You may be able to determine which physical disk has failed by clicking the RIGHT button over the failed VM disk icon to access the VM Disk Properties form. The GUI portrays a failed VM disk icon in a different color or pattern.

   The `Physical Disk` text field on the VM Disk Properties form gives the device name for the failed disk that the VM disk is assigned to. Write down the device name of the failed disk; you will need it later to locate the failed physical disk so that you can replace it.

2. **Evacuate subdisks from the failed disk as follows:**

Note – This operation can only be performed between two disks in the same disk group.

---

   a. **Go to the view window corresponding to the disk group in which you want to perform this operation.**

b. **Select the disk *from* which subdisks are to be evacuated by clicking LEFT on its icon.**

c. **From the Basic-Ops menu, select Disk Operations, then Evacuate Subdisks.**
The Evacuate Subdisks form appears.

d. **Complete the Evacuate Subdisks Form.**

Table 6-2 Evacuate Subdisks Form

Field	Description
Disk group name:	Enter the name of the disk group to which both disks belong. Both disks must share the same disk group.
Evacuate From:	Displays the name of the VM disk from which the subdisks are to be evacuated, which can be changed. Both disks must belong to the same disk group and subdisks must exist on the disk from which the evacuation is to be conducted.
To:	Enter the name of the VM disk to which the subdisks are to be moved. This field is optional. However, if no target disk is specified, the subdisks are evacuated to one or more random disks (depending on disk space availability).

e. **When the form is properly completed, select Apply to activate the subdisk evacuation.**
The subdisks move from their original disk icon to the targeted disk icon.

3. **Refer to the device name that you wrote down earlier in this procedure to find the physical disk that needs to be replaced.**

6.3 Displaying Properties of an Object

The properties of the object corresponding to a given icon can be displayed (and potentially adjusted) in a properties form. A properties form provides detailed information about the characteristics of a particular object.

There are properties forms for the following:

- physical disks
- partitions
- VM disks
- volumes
- plexes
- subdisks
- file systems

Note – If you want to display the properties of a file system, follow the instructions given in Section 6.3.1, “Displaying File System Properties,” on page 6-6.

♦ **Move the pointer over the icon and click the RIGHT button (if the icon happens to be under analysis, Shift-RIGHT must be used instead).**
When the properties form appears, details of the icon’s configuration are displayed. Some of the displayed properties can be altered directly through this form by altering the appropriate fields and then selecting the form’s Apply button. Selecting the Reset button converts any altered form fields back to their values at the time when the form was popped up.

Figure 6-1 shows a typical properties form.

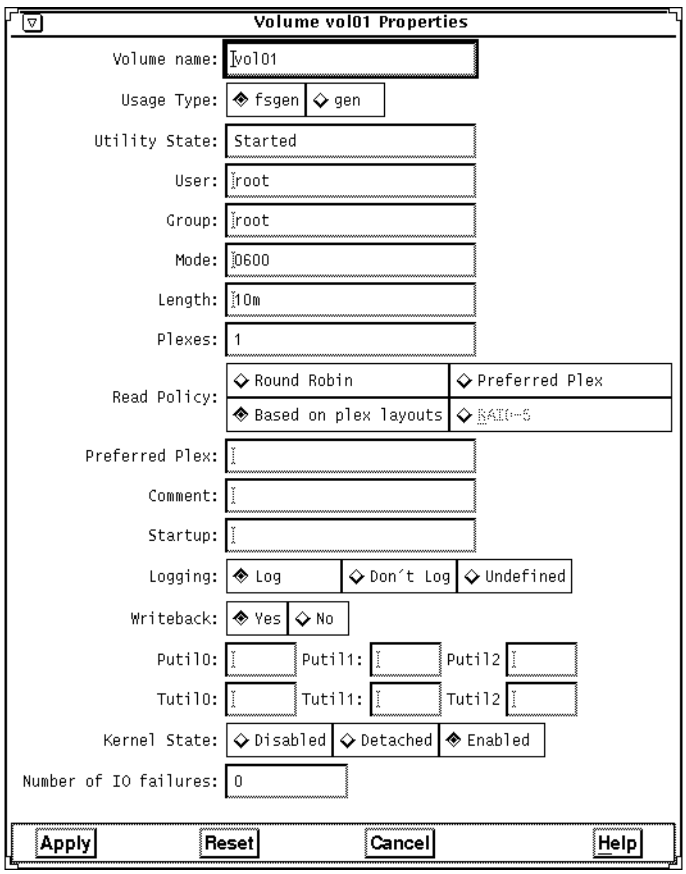

Figure 6-1 Properties Form

6.3.1 Displaying File System Properties

The File System Properties form displays useful details on the attributes of a particular file system. A single File System Properties form provides access to all file systems known to Volume Manager at any given time.

Since file systems do not have associated icons, file system properties cannot be accessed by clicking RIGHT on an icon. Instead, the File System Properties form is accessed in either of the following ways:

- Click the RIGHT button on the file system mount point, which appears below the volume icon on which the file system is mounted.
- Select Display Properties from the File System Operations menu.

Figure 6-2 illustrates the File System Properties form. All fields in this properties form are read-only, so this form cannot be used to alter the attributes of file systems.

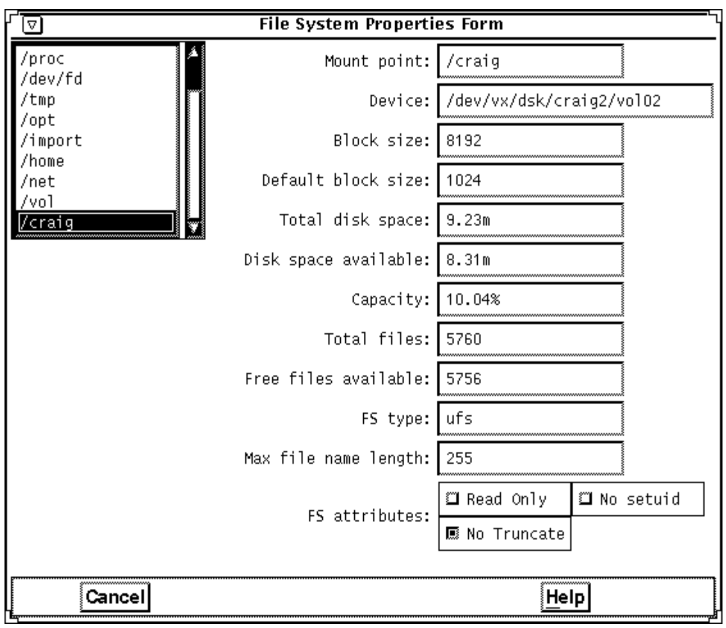

Figure 6-2 File System Properties Form

You can display information on a specific file system via the File System Properties form as follows:

1. **Optionally select a volume whose mounted file system's properties are to be displayed.**

2. **From the Basic-Ops menu, select File System Operations, then Display Properties.**
   The File System Properties form appears. If a volume has been selected, the properties for the file system that resides on that volume are displayed by default. Select a different mounted file system from the menu box in the upper left corner of this form, if desired.

The properties displayed on the right side of the form correspond to the selected mount point name from the list on the left.

The File System Properties form contains the following fields of information:

Table 6-3 File System Properties Form

Mount point:	The mount point of this file system.
Device:	The block device on which this file system resides.
Block size:	The block size of this file system.
Default block size:	Fundamental file system block size.
Total disk space:	Number of megabytes of disk storage this file system occupies.
Disk space available:	Number of megabytes of disk storage on this file system that is available for use.
Capacity:	Percentage of the total disk storage space still available for use. This is the “disk space available” divided by the “total disk space.”
Total files:	The maximum number of files allowed on this file system.
Free files available:	The number of files that still may be created on this file system.
FS type:	The file system type (such as `ufs`).
Max file name length:	This is the maximum number of characters that a file name may be on this file system. This restriction is imposed by the file system.

Table 6-3 File System Properties Form (Continued)

FS attributes:		This field indicates attributes associated with this file system.
	Read Only	Indicates a file system that cannot be written to.
	No setuid	Indicates a file system that does not support setuid/setgid semantics.
	No Truncate	Indicates a file system that does not truncate file names longer than NAME_MAX, the maximum filename length as defined in the system header files.

6.4 Displaying Relationships between Volume Manager Objects

Volume Manager uses projection to show relationships between icons that represent Volume Manager objects. Projection is illustrated using color (yellow is the default) or bitmap patterns. Projection highlights those objects that the selected object is composed of and illustrates the relationship between the objects. For example, if a volume is selected for projection, the corresponding subdisks are highlighted within the volume icon and also on the appropriate disk icons. If the selected icon has no associated objects, Volume Manager issues a warning to this effect.

To show the projection of a particular icon:

♦ **Click the MIDDLE button on the icon while holding down the Shift key (Shift-MIDDLE).**

To stop the projection of a particular icon:

♦ **Click Shift-MIDDLE on the icon again.**

You can also start or stop projection by selecting an icon and then using the Icon Projection submenu of the Projection menu.

You can request projection in any view. When an icon is highlighted by projection, all icons representing that object in all view windows where it appears are highlighted.

Table 6-4 summarizes the projection relationships that are highlighted for particular icon types. If no icons of the correct type are associated with the selected icon, then nothing is highlighted.

Table 6-4 Projection Behavior

Icon Selected	Icons Highlighted
Volume	All subdisks associated with any plex associated with the volume
Plex	All subdisks associated with the plex
Subdisk	Associated plex and volume, and all other subdisks associated with the plex
VM Disk	All plexes associated with the subdisks that reside on the disk
Physical Disk	All associated partition and VM disk icons
Partition	The VM disk associated with the partition

When projection is turned on and left on from two different objects, any icon that happens to be related to both of these objects receives two "layers" of projection highlighting. Projection must then be turned off from both objects that started it (or all projection in the current session must be stopped) before the double-highlighted icon returns to its normal state. For example, if projection is turned on from both a volume and a plex related to the same subdisk, then that subdisk is highlighted twice even though it only appears to have one layer of highlighting.

Figure 6-3 illustrates highlighting that results from the selection of a volume icon for projection.

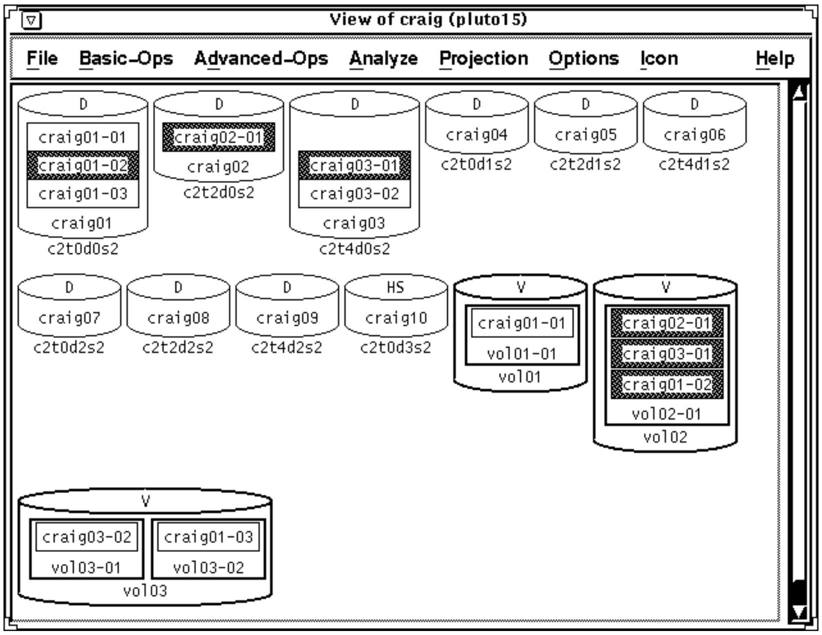

Figure 6-3 Projection

6.5 Displaying Activity Levels on Objects

You can display the activity levels on objects from the View of SPARCstorage Arrays window or from a Volume Manager window.

6.5.1 Displaying Activity Levels on Objects in a SPARCstorage Array Window

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 6-4).**

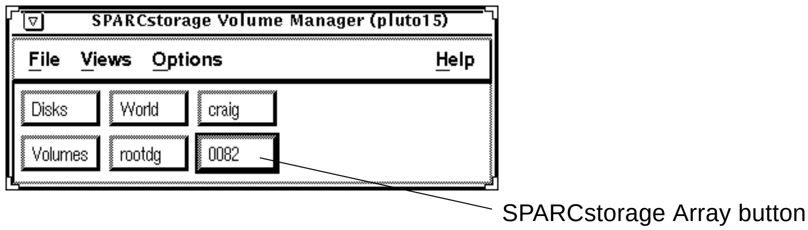

Figure 6-4 Root Window

This will bring up the View of SPARCstorage Array (see Figure 6-5).

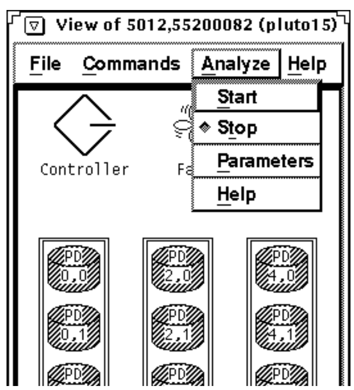

Figure 6-5 View of SPARCstorage Array

2. From the Analyze menu, select Start (see Figure 6-5).
The disk icons in the view will change their colors or patterns to reflect their activity level. Table 6-5 gives the default colors and patterns for analysis.

Table 6-5 Default Analysis Colors and Patterns

Monochrome Pattern	**Color**	**Analysis Activity Level**
Cross_weave	Green	Low activity level
Root_weave	Yellow	Medium activity level
Wide_weave	Red	High activity level

To stop analysis on certain objects:

1. **Select the SPARCstorage Array icons that you want to stop analysis on.**
2. **Select Stop from the Analyze menu.**

To set the parameters for the analysis:

1. **From the Analyze menu, select Parameters (see Figure 6-5).**
   This will bring up the Analysis Parameters form (see Figure 6-6).

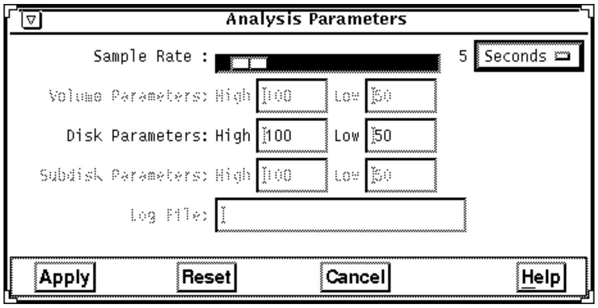

Figure 6-6 Analysis Parameters Form

Note – The Volume Parameters, Subdisk Parameters, and Log File lines will be grayed over in the SPARCstorage Array Analysis Parameters form, but will not be grayed out in the generic Volume Manager Analysis Parameters form.

---

2. **Change the parameters on the form and press LEFT over the Apply button.**
   Table 6-6 gives an explanation of each of the parameters in the Analysis Parameters Form.

Table 6-6 Analysis Parameters Form

Parameters	**Meaning**
Sample Rate	Use the slider bar to select the time interval, in seconds, between data samples. The interval should be from 1 to 60 in either seconds or minutes. A shorter interval means that data will be updated more often, but is also a higher load on the system. The default is 5 seconds. To convert the interval from seconds (default) to minutes, click LEFT on the box to the right of the radio button; when the pop-up menu appears, select minutes.
Volume Parameters	Enter the high and low values that will decide the coloring/patterning of the volume icons. The units for the values are total number of read/write operations. Although high and low values appear in this field, they should be changed according to the disk type.
Disk Parameters	Enter the high and low values that will decide the coloring/patterning of the VM disk icons. The units for the values are total number of read/write operations. Although high and low values appear in this field, they should be changed according to the disk type.
Subdisk Parameters	Enter the high and low values that will decide the coloring/patterning of the subdisk icons. The units for the values are total number of read/write operations. Although high and low values appear in this field, they should be changed according to the disk type.
Log File	Enter the name of the file to be used for statistics logging. If the file does not already exist, it will be created. To stop logging to the file, the filename text in this field must be erased. If logging is left in effect too long, a very large log file could result. Unlike the other fields, this one does not persist across sessions. This log file is a binary file. In order to view the log file, /opt/vxva/bin/vxvalog2text *filename* must be run on this file to process it for viewing.

You can also bring up the Analysis Statistics form for an object that is being analyzed by clicking the RIGHT mouse button over that object's icon (the object must be undergoing analysis). Table 6-7 gives an explanation of each of the fields in the Analysis Statistics form.

Table 6-7 Analysis Statistics Form

Field	**Description**
Reads:	The number of times the object was read from during the last interval.
Writes:	The number of times the object was written to during the last interval.
Total R/W:	The total number of reads and writes during the last interval.
Blocks Read:	The number of disk blocks read from the object during the last interval.
Blocks Written:	The number of disk blocks written to the object during the last interval.
Total Blocks:	The total number of blocks read from or written to the object during the last interval.
Avg Read Time:	The average time, in milliseconds, that it took for a read operation to complete. This is equal to the number of reads during the last interval divided by the total time spent on reads.
Avg Write Time:	The average time, in milliseconds, that it took for a write operation to complete. This is equal to the number of writes during the last interval divided by the total time spent on writes.
Interval:	The actual time, in seconds, since the last data was sampled. This may vary slightly from the specified interval time due to uncontrollable variances from system to system.

6.5.2 Displaying Activity Levels on Objects in a Volume Manager Window

Through analysis, the Volume Manager displays statistics about the performance of Volume Manager objects. These statistics are displayed both visually (with different colors or patterns), and numerically (with pop-up statistics forms). Analysis displays activity levels on selected icons representing Volume Manager objects.

Note – Only volume and VM disk icons can be selected for analysis.

---

1. **Select one or more VM disk and/or volume icons.**

2. **Select Start from the Analyze menu.**
   Volumes, VM disks, and subdisks that are associated with the selected icon(s) change their colors or patterns to reflect their relative activity levels (high, medium, or low). Figure 6-7 shows a typical analysis for a selected volume.

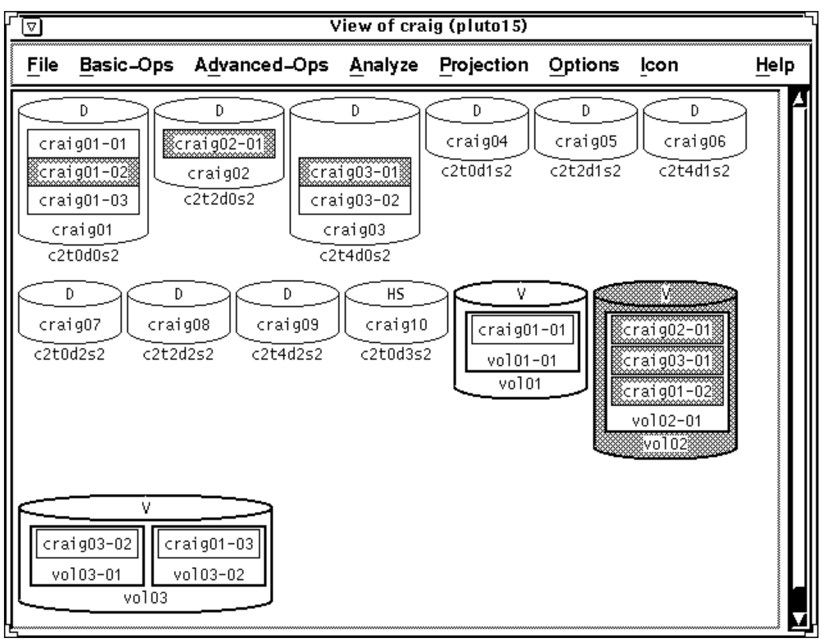

Figure 6-7 Analysis

To stop analysis on certain objects:

1. **Select the icons that you want to stop analysis on.**
2. **Select Stop from the Analyze menu.**

To stop analysis on all objects:

1. **Select the icons that you want to stop analysis on.**
2. **Select Stop All from the Analyze menu.**

You can also set the parameters for the analysis as follows:

1. **Select Parameters from the Analyze menu.**
   You will see the Analysis Parameters form (see Figure 6-6).
2. **Change the parameters on the form and press LEFT over the Apply button.**
   Table 6-6 on page 6-14 gives an explanation of each of the parameters in the Analysis Parameters Form.

You can bring up the Analysis Statistics form for an object that is being analyzed by clicking the RIGHT mouse button over that object's icon (the object must be undergoing analysis). Table 6-7 gives an explanation of each of the fields in the Analysis Statistics form

6.6 Displaying Past and Present Commands

The Command Info Window displays commands that are currently being executed by Volume Manager, as well as previous commands. Both Volume Manager and system commands are displayed in this window as they are invoked by Volume Manager. The status and output of these commands are also displayed here. Previously executed commands can be executed again directly from this window.

♦ **From the Options menu, select Popup the Command Window (see Figure 6-8).**

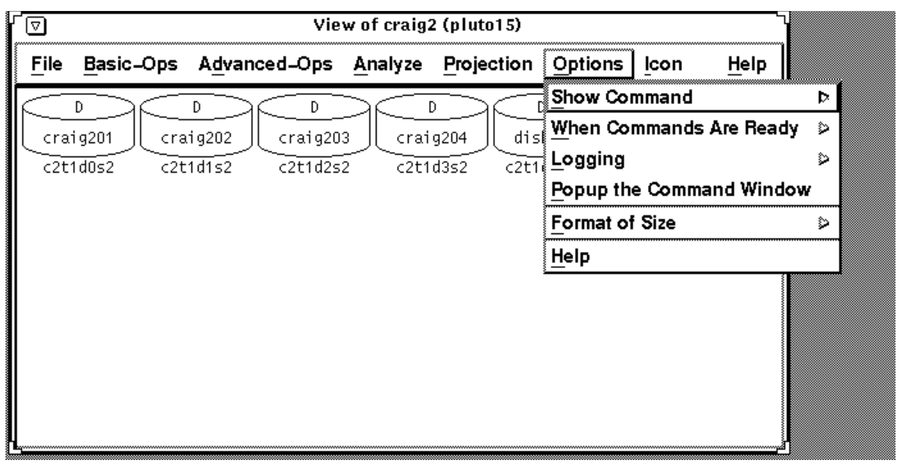

Figure 6-8 Accessing the Options Menu

Figure 6-9 shows the Command Info Window.

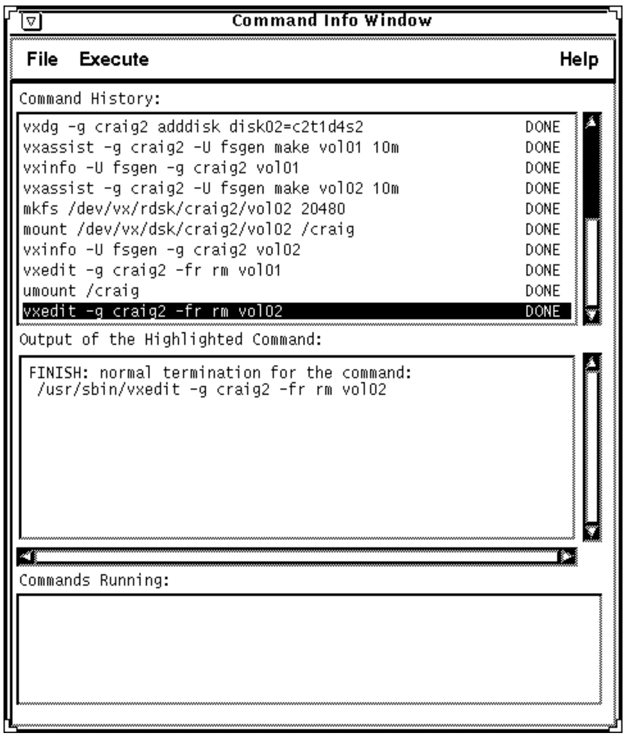

Figure 6-9 Command Info Window

6.6.1 Command Info Sections

This window is divided into three sections:

- Command History
- Output of the Highlighted Command
- Commands Running

6.6.1.1 Command History

This section displays a chronological listing (or history) of the commands sent to the Volume Manager or the system for execution. The last ten commands are saved and displayed, with the most recent command at the bottom. If a command is too long, only the first few arguments of the command are shown. Each of these commands is listed along with its status, which is displayed to the right of the command. The status of a command is one of the following:

- DONE — Command successfully completed
- BROWSE — Command not executed, just displayed here
- ERROR — Command terminated with error condition
- UNKNOWN — Command status cannot be determined by the Visual Administrator (this rarely occurs, but generally results from an internal error or a command being interrupted unexpectedly)

If a command is selected in this window, it is shown in its entirety in the middle section of this window, along with its output.

6.6.1.2 Output of the Highlighted Command

When a command is highlighted in the Command History section, its information is displayed in the Output of the Highlighted Command section of the window. The results of the command are indicated here (regardless of whether it succeeded or failed), along with the full command. If the command terminated abnormally or exited with an error condition, the error message is also displayed.

6.6.1.3 Commands Running

This section displays the command that is currently running. This command has been sent to the system or the Volume Manager, but has not yet terminated. As soon as the command completes, it disappears from this section of the window.

6.6.2 Using the Command Info Window

You access the Command Info Window by selecting Popup from the Command Window from the Options menu. This window may also pop up automatically when an error occurs with a command or operation. You can close the Command Info Window by selecting Close from its File menu.

You can execute a command directly from the Command Info Window:

1. **Select a command by clicking LEFT on the desired command in the Command History window.**
   Once selected, the command is highlighted.

2. **Access the Execute menu and select the Execute option.**

This procedure sends the selected (highlighted) command to the system or the Volume Manager for execution. This is useful for executing a command again, re-executing a failed command that should now succeed, or executing a command that was only shown (in Browse status) before.

In some circumstances, the Execute with Force option may be used rather than Execute. This option adds `-f` to the executed command, which forces the Volume Manager to complete an operation that is considered unsafe and to disregard error messages. The `-f` option is available with some Volume Manager commands only and does not apply to file system operations.

Caution – The Execute with Force option is a *very* dangerous operation, which can result in irreparable loss of data; it should only be used when the user is sure that an operation should succeed, even though Volume Manager error checking prevents it.

6.7 Determining Amount of Free Space on VM Disks

For operations that require free disk space, Volume Manager usually allows you to designate disks with sufficient free space. If no disks are specified, disks with available space are automatically used if an operation is performed using Basic-Ops. With Advanced-Ops, however, you are expected to designate disks with sufficient space. Before designating a disk to be used, you should verify the amount of free space on that disk:

♦ **Click the RIGHT button on a portion of the Volume Manager disk icon that is not covered by another icon.**
The disk properties form appears. At the bottom of this form is a field that indicates the maximum free space available for use on this disk.

It is sometimes possible to obtain information about a specific region of a VM disk that contains free space. If subdisks have been removed from a disk, resulting in visual gaps between the subdisk icons located on that disk, it is possible to check the amount of space freed by the subdisk removal.

Figure 6-10 illustrates a VM disk icon containing a gap created by the removal of a subdisk.

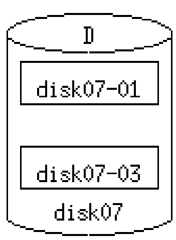

Figure 6-10 VM Disk with a Gap

To check the amount of free space in a specific region of a disk:

1. **Go to the view window corresponding to the disk group in which you want to perform the operation.**

2. **Click the RIGHT button on the gap between subdisk icons on the desired VM disk icon.**
   The Free Space form appears. This form provides the following read-only information:

Table 6-8 Free Space Form

Device:	The name of the device where this free space resides.
Hole offset:	The offset, in sectors, into the VM disk where this free space extent begins.
Hole size:	The extent size of this free space.

3. **Close the Free Space form by selecting Cancel.**

6.8 Checking a File System for Consistency

You can check a file system for consistency using `fsck` in the Volume Manager. You must select a volume icon containing an unmounted file system in order for this operation to succeed.

1. **Go to the view window corresponding to the disk group in which you want to perform this operation (View of rootdg, by default).**
2. **Select the volume whose unmounted file system is to be checked.**
3. **From the Basic-Ops menu, select File System Operations, then Check File System (fsck).**
4. **Access the Command Info Window to see if `fsck` had any problems.**
   Highlight the `fsck` command line listed in the Command History section of this window and then examine any corresponding output in the Output of the Highlighted Command section. This should reveal any `fsck`-related errors or inconsistencies. If the `fsck` line is accompanied by the word `DONE`, there are not likely to be any errors or inconsistencies present.

6.9 Backing Up Your Data

With Volume Manager, the snapshot operation creates a new volume that is a snapshot of an existing volume. This is done by creating a mirror of the existing volume (creating and associating a plex) using disk space from the pool of free disk space. The mirror is brought up to date (this may take some time) and a separate (snapshot) volume is then created for it. The snapshot volume represents a consistent copy of the original volume at the time the snapshot was begun. The snapshot volume can be used to make a backup of the original volume at a convenient time without stopping it. After the backup is made, the snapshot volume can be removed without losing any data.

The snapstart operation creates a write-only backup mirror, which is attached to and synchronized with the volume to be backed up. The amount of synchronization time depends on the size of the volume and the amount of data to be copied. When synchronized with the volume, the backup mirror is ready to be used as a snapshot mirror. However, it continues being updated until it is detached during the actual snapshot portion of the procedure.

The on-line backup procedure is completed by running the command on the volume with the snapshot mirror. This operation detaches the finished snapshot mirror, creates a new normal volume, and attaches the snapshot mirror to it. The snapshot then becomes a read-only, functioning mirror.

The amount of time involved in creating the snapshot mirror is long in contrast to the brief amount of time that it takes to create the snapshot volume.

6.9.1 Creating a Snapshot

1. **Go to the view window corresponding to the disk group in which you want to perform this operation (View of rootdg, by default).**

2. **Select the volume to be used for backup/snapshot purposes.**

3. **From the Basic-Ops menu, select Volume Operations, then Snapshot, then Snapstart.**
   A snapshot mirror appears within the selected volume. As the snapshot mirror is updated, its icon is grayed out. This synchronization takes a varied amount of time, which could be significant (depending on the size of the selected volume).

4. **Once the snapshot mirror is fully updated (and its icon is no longer grayed out), prepare to make the snapshot volume by optionally asking users to save files and temporarily reduce activity.**

5. **Unmount any file system or disable any database activity for that volume.**

6. **Select the volume containing the snapshot mirror.**

7. **From the Basic-Ops menu, select Volume Operations, then Snapshot, then Snapshot again.**
   The Snapshot form appears.

8. **Complete the Snapshot form as follows:**

Table 6-9 Snapshot Form

Selected volume:	This field displays the name of the volume to be used as the snapshot source. This field cannot be changed.
Snapshot name:	Enter the name of the snapshot volume to be created as a backup. Although a default name appears in this field, you should use a name that more closely resembles that of the selected volume for easier association. The maximum length is 14 characters.

9. **When the form is properly completed, select Apply to activate the snapshot operation.**
   When the backup plex is synchronized with the mirror, the snapshot operation begins. This portion of the procedure should take only a brief amount of time. The snapshot operation detaches the finished snapshot plex, creates a new normal volume, and attaches the snapshot plex to it. As this occurs, the original volume icon returns to its former state and the backup plex moves over into the new snapshot volume (which has the name specified in the Snapshot name: field of the Snapshot form). The new snapshot volume remains as a consistent copy of the selected volume at the time the snapshot was begun.

10. **Mount the file system or start up data base activity for that volume.**

11. **Back up the snapshot volume to tape (or some other medium).**

12. **Remove the snapshot volume when it is no longer necessary, as it takes up as much space as a normal volume.**

Figure 6-11 illustrates a volume and its snapshot.

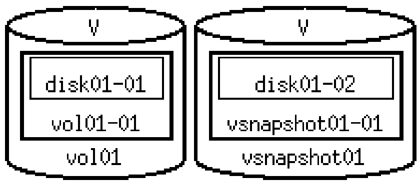

Figure 6-11 Volume and Snapshot

6.9.2 *Removing a Snapshot*

Once it has served its purpose, a snapshot volume can be removed in the same way as a regular volume is removed.

1. **Go to the view window corresponding to the disk group in which you want to perform this operation (View of rootdg, by default).**
2. **Select the snapshot volume to be removed.**
3. **From the Basic-Ops menu, select Volume Operations, then Remove Volumes Recursively.**
   The snapshot volume and its components disappear.

6.10 *Showing Free Subdisks*

The Projection menu provides access to a feature that highlights any free subdisk icons. This is useful for identifying subdisks that are not currently associated with any plexes and should either be used or removed to free up the space that they occupy.

♦ **From the Projection menu, select Show Free Subdisks, then Start (see Figure 6-12).**

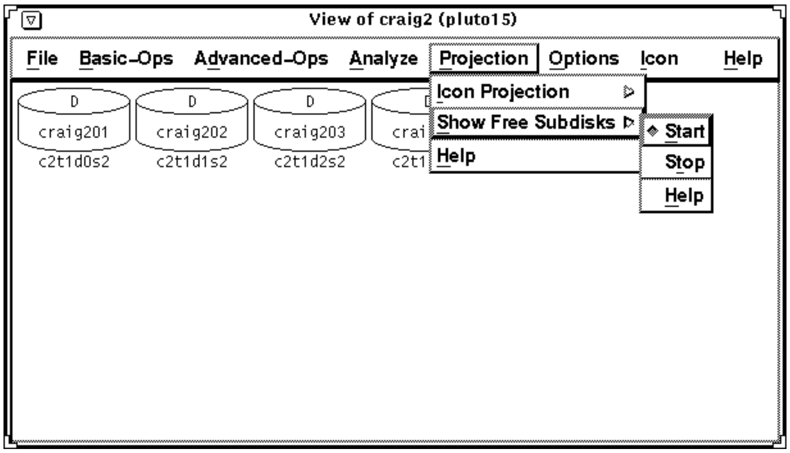

Figure 6-12 Accessing the Show Free Subdisks Menu

Once turned on, free subdisks will continue to be highlighted until this feature is turned off.

6.11 Logging Recent Disk Activity

A subdisk can be associated with a plex as a log subdisk (with Dirty Region Logging in effect). A log subdisk is used to log recent disk activity. Only one log subdisk can exist on a given plex. For mirrored striped or concatenated volumes, this will add a Dirty Region Log; for RAID-5 volumes, this will add a RAID-5 log.

Note – Logs are only useful for mirrored volumes or RAID-5 volumes.

To create a log subdisk for a new volume:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Create a volume using the instructions given in Chapter 4, "Setting Up the Software Configuration."**
3. **In the Volume/FS Create form, locate the Create log subdisk line and click LEFT on Yes.**

To create a log subdisk for an existing volume:

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Select a mirrored or RAID-5 volume by clicking LEFT on the icon.**
3. **From the Basic-Ops menu, select Volume Operations, then Add Log.**
   A log associated with a subdisk will appear in the volume icon. A DRL log subdisk for a mirrored volume can be identified by its double border, unlike RAID-5 logs.

Performing Service-Related Software Tasks 7

This chapter gives instructions on performing software tasks necessary for certain service procedures:

7.1 Flushing or Purging Outstanding Writes from NVRAM

Before you replace a FRU (Field Replaceable Unit), you must first determine if your system is set up for fast writes.

- If your system is set up for fast writes, go to "Flushing or Purging Outstanding Writes from NVRAM" on page 7-1.
- If your system is *not* set up for fast writes, go directly to the appropriate section to replace the FRU.

Note – If you are not sure if your system is set up for fast writes or not, you should go through the procedures given in "Flushing or Purging Outstanding Writes from NVRAM" anyway, since it won't affect drives that aren't set up for fast writes.

7.1.1 Flushing or Purging Outstanding Writes from NVRAM

You must first determine if you want to flush or purge the outstanding writes from the NVRAM:

- The *flush* option flushes any outstanding writes from the NVRAM to the disk drives. Follow the instructions given in “Flushing Outstanding Writes from the NVRAM” on page 7-2 to do this.
- If you get an error message when you tried to flush the writes from the NVRAM, then you must *purg*e the writes from the NVRAM. The purge option gives the controller permission to completely throw away any outstanding writes currently residing in the NVRAM for a disk drive. You should only use this option if you are replacing a failed disk drive, since you cannot flush the outstanding writes to the failed disk drive. Follow the instructions given in “Purging Writes from NVRAM” on page 7-3 to do this.

7.1.1.1 Flushing Outstanding Writes from the NVRAM

To flush writes from the NVRAM:

1. **Click SELECT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 7-2).**
   This will bring up the View of SPARCstorage Arrays window.

2. **From the Commands menu, select Controller, then Sync Cache (see Figure 7-1).**
   - If you did not get an error message after selecting Sync Cache, then all outstanding writes were flushed from the NVRAM. Go directly to the appropriate section to replace the FRU.
   - If you got an error message after selecting Sync Cache, you will have to purge the writes from the NVRAM. Refer to “Purging Writes from NVRAM” on page 7-3 for those instructions.

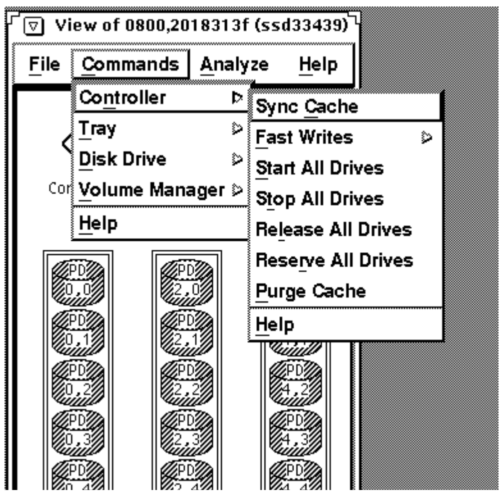

Figure 7-1 Flushing Outstanding Writes from NVRAM

7.1.1.2 *Purging Writes from NVRAM*

The purge option gives the controller permission to throw away any outstanding writes that are currently pending in NVRAM for the physical disk you select.

Caution – This option will destroy all data currently on the NVRAM for the selected disk drive. Use this option *only* if you can no longer access the disk drive; if you can still access the disk drive, you should flush the data to the drives using the instructions given in “Flushing Outstanding Writes from the NVRAM” on page 7-2.

To purge writes for the selected disk drive:

1. **Click SELECT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 7-2).**
   This will bring up the View of SPARCstorage Arrays window (see Figure 7-1).

2. **Click SELECT over the physical disk icon that you want to purge writes from (see Figure 7-1).**
   The physical disk icon will turn black, indicating that it has been selected.

3. **From the Commands menu, select Disk Drive, then Purge (see Figure 7-1).**

All outstanding writes for the disk drive are now purged. Go directly to the appropriate section to replace the FRU.

7.2 When Powering Down the SPARCstorage Array

If you are powering down the SPARCstorage Array, you must first stop all drives in the SPARCstorage Array. The procedures for these software tasks are given in the following sections.

7.2.1 Preliminary Software Tasks

1. **Unmount the file system(s) on all the disks in the array.**

2. **Stop all database processes that are accessing any of the disks in the array.**

3. **Stop all other processes that may be accessing any of the disks in the array.**

4. **Flush or purge the outstanding writes from the NVRAM, if necessary.**
   Refer to Section 7.1, “Flushing or Purging Outstanding Writes from NVRAM,” on page 7-1 for more information.

7.2.2 Spinning Down All Drives in the SPARCstorage Array

1. Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 7-2).

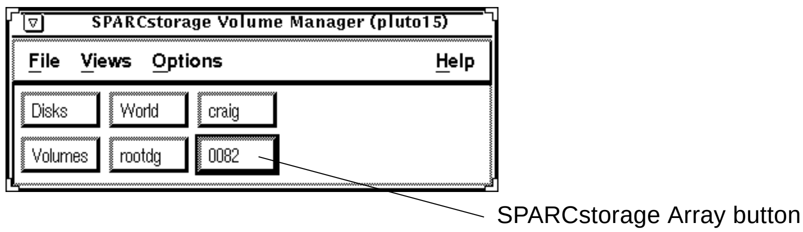

Figure 7-2 Accessing a SPARCstorage Array

This will bring up the View of SPARCstorage Arrays window.

2. From the Commands menu, select Controller, then Stop All Drives (see Figure 7-3).

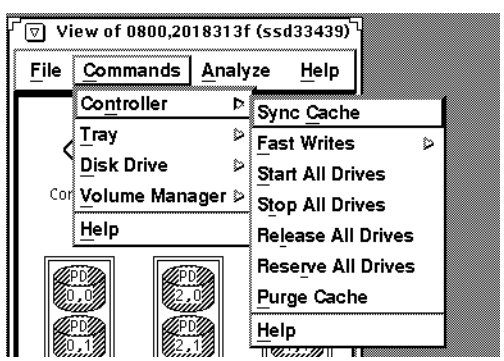

Figure 7-3 Accessing the Controller Menu from the Commands Menu

You will see the following confirmation window:

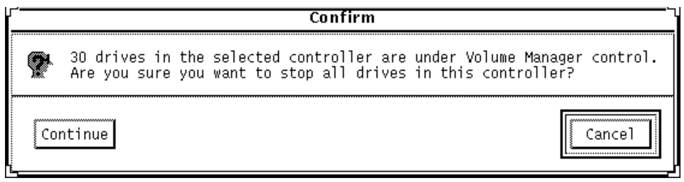

Figure 7-4 Stop Drives Confirmation Window

Click LEFT over the Continue screen button. All the disk drives in the SPARCstorage Array are now stopped. You can now power down the SPARCstorage Array safely.

Note – If you make any changes to the Volume Manager configuration at this point (for example, if you create a subdisk), the stopped drives may start up again, even if the change that you made was not related to the stopped drives. If this happens, simply stop the started drives again.

7.3 When Replacing a Physical Disk Drive in the SPARCstorage Array

There are two sets of tasks you should perform when you replace a drive tray in the SPARCstorage Array:

- *Before* replacing a drive — Section 7.3.1 on page 7-7
- *After* replacing a drive — Section 7.3.2 on page 7-16

7.3.1 Before Replacing a Drive

Before replacing the drive, you must first perform the preliminary software tasks.

7.3.1.1 Preliminary Software Tasks

Before replacing the drive, you must first perform the following software tasks:

1. **Unmount the file system(s) on the disks in the tray with the faulty drive.**
2. **Stop all *database* processes that are accessing the disks in the tray with the faulty drive.**
3. **Stop all *other* processes that may be accessing the disks in the tray with the faulty drive.**
4. **Flush or purge the outstanding writes from the NVRAM, if necessary.** Refer to Section 7.1, "Flushing or Purging Outstanding Writes from NVRAM," on page 7-1 for more information.
5. **Determine if the failed drive is under Volume Manager control or not.**
   - If the failed drive is not under Volume Manager control, stop all the drives in the tray using the instructions in Section 7.3.1.9, "Spinning Down All Drives in a Drive Tray," on page 7-14.
   - If the failed drive is under Volume Manager control, look at the subdisks in the failed VM disk and determine what kind of configuration each of those subdisks falls into. Then look at the flowcharts in Figure 7-5 and Figure 7-6 for the tasks that you must perform for the subdisks in each configuration.

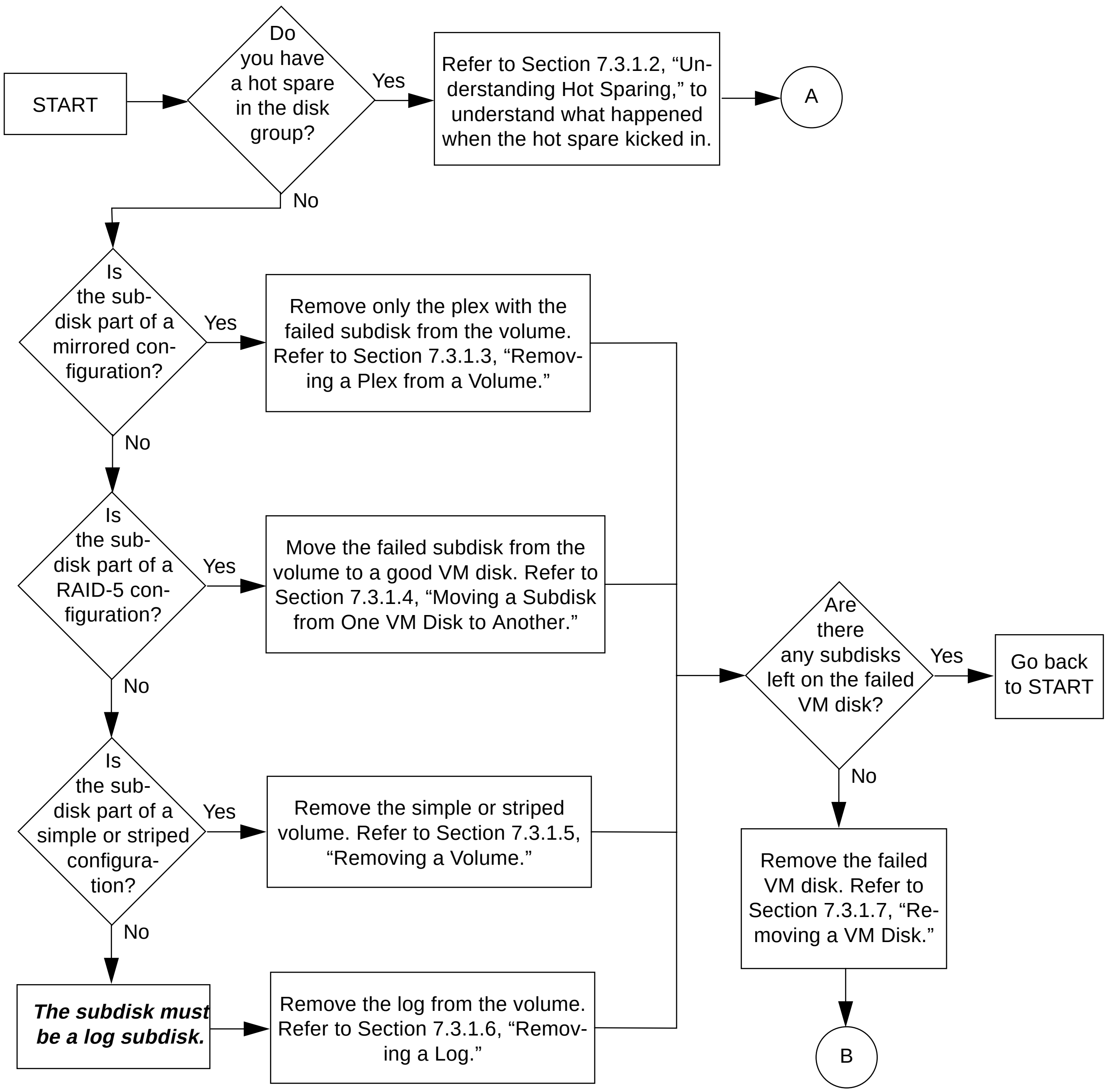

Figure 7-5 Disk Replacement Flowchart (1 of 2)

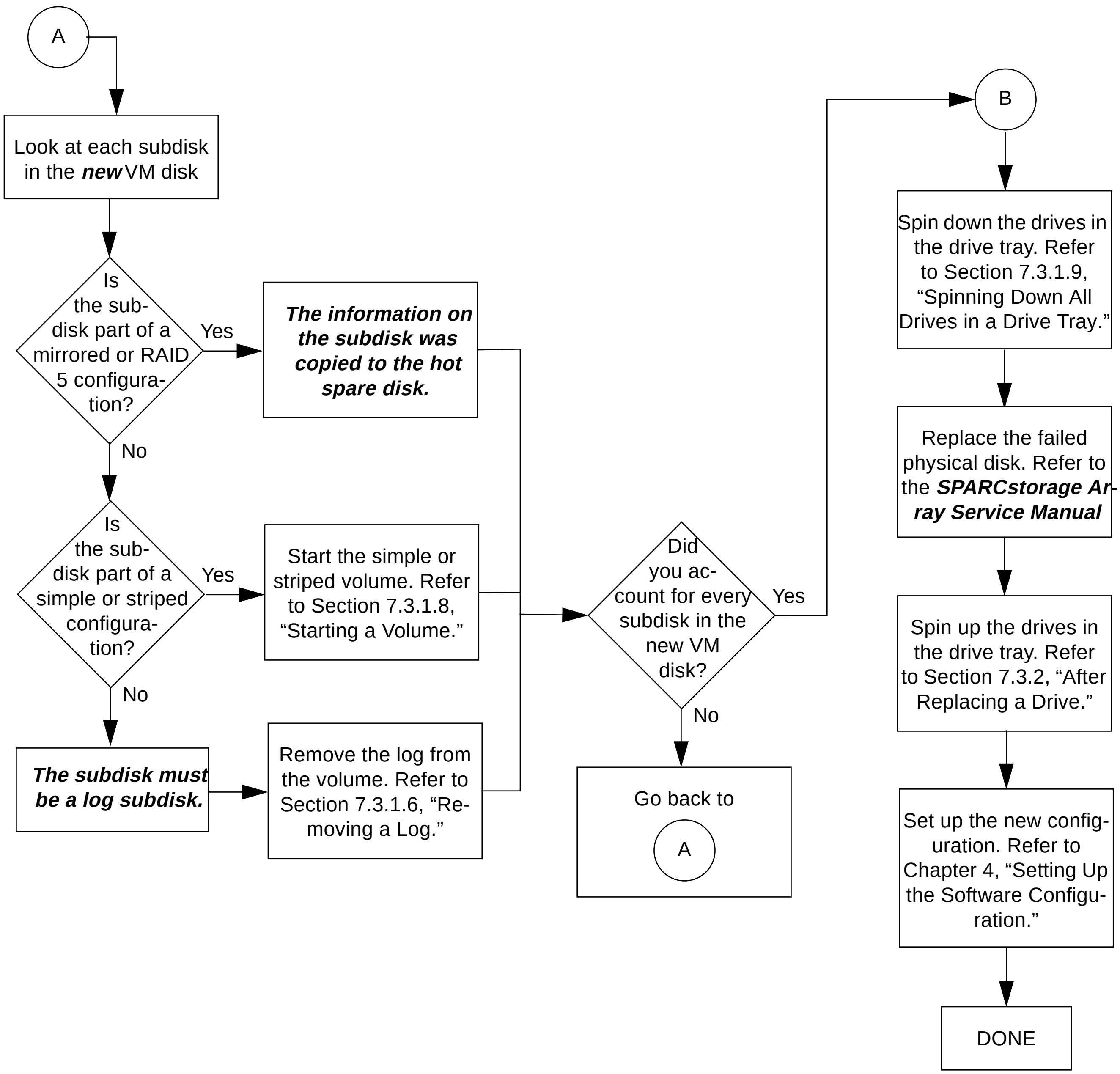

Figure 7-6 Disk Replacement Flowchart (2 of 2)

7.3.1.2 Understanding Hot Sparing

Hot-sparing can be used to automatically rebuild mirrored or RAID-5 data when a disk fails. If you have a hot spare in a disk group and a drive fails, Volume Manager will go through the following steps:

1. The VM disk assigned to the failed physical disk will disappear from the disk group view.
2. The hot spare disk will change to a VM disk and will be renamed with the failed VM disk's name.
3. The new VM disk will take all the subdisks from the failed VM disk.

For example, looking at the disk group in Figure 7-7, you can see that there is one hot spare disk (craig10) and several VM disks (craig01 through craig09) in that disk group.

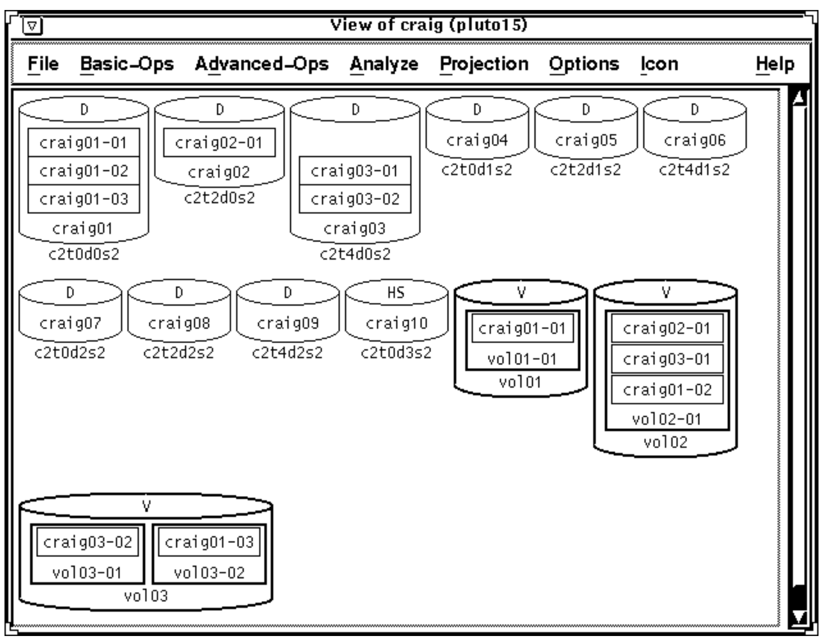

Figure 7-7 Example of Hot Sparing

If the VM disk craig01 failed, then Volume Manager would do the following:

1. The VM disk craig01 would disappear from the disk group view.
2. The hot spare disk craig10 would change to a VM disk and would be renamed to craig01.
3. The new VM disk craig01 would take all the subdisks from the old VM disk craig01.

Figure 7-8 gives a graphical representation of what happens when a hot spare kicks in for a failed VM disk.

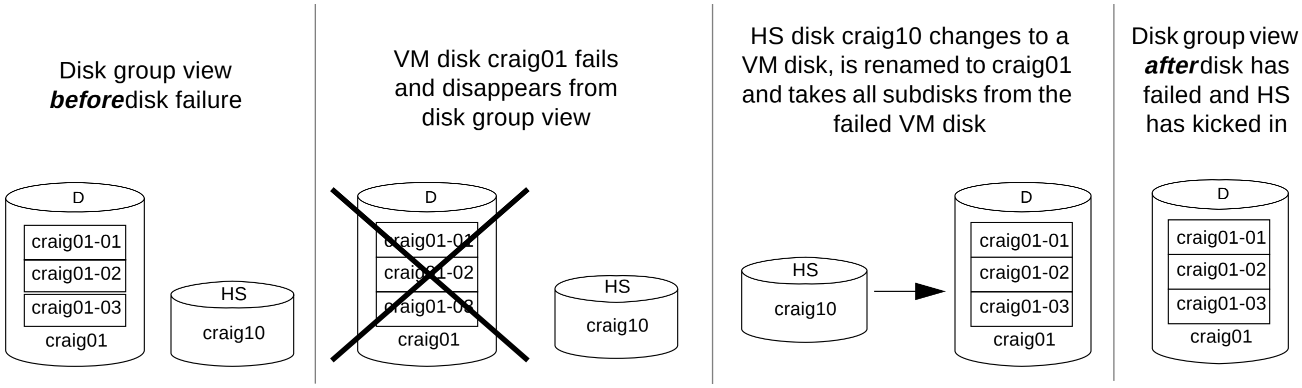

Figure 7-8 Graphical Representation of Hot Sparing

7.3.1.3 Removing a Plex from a Volume

1. **Select the plex with the failed subdisk by clicking the LEFT button on its icon.**
   The selected plex can be either associated or dissociated.
2. **From the Basic-Ops menu, select Volume Operations, then Remove Mirror or Log.**
   If the plex is associated, it is automatically dissociated. The plex icon then disappears entirely, along with any associated subdisk icons.
3. **Go back to the flowchart to determine if you have to do anything to any other subdisks from the failed VM disk.**

7.3.1.4 Moving a Subdisk from One VM Disk to Another

1. **Drag the subdisk icon from the failed VM disk to a second VM disk and drop it there.**
2. **Go back to the flowchart to determine if you have to do anything to any other subdisks from the failed VM disk.**

7.3.1.5 Removing a Volume

1. **Select the volume with the failed subdisk by clicking LEFT on its icon.**
2. **From the Basic-Ops menu, select Volume Operations, then Remove Recursively.**
   If the selected volume is enabled and therefore in danger of losing valuable data, a Volume Manager Warning window appears to announce this fact. The warning window requires input before Volume Manager will proceed with anything (see Figure 7-9).

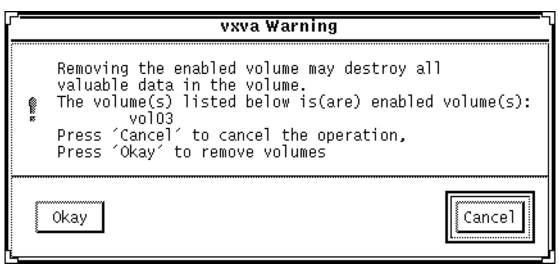

Figure 7-9 Volume Removal Warning

   Select `Okay` to proceed with the removal. The volume disappears when the removal is complete.

3. **Go back to the flowchart to determine if you have to do anything to any other subdisks from the failed VM disk.**

7.3.1.6 Removing a Log

1. **Go to the volume that contains the log from the failed VM disk.**
2. **Select the log plex to be removed by clicking LEFT on its icon.**

3. **From the Basic-Ops menu, select Volume Operations, then Remove Mirror or Log.**
   The log and any associated subdisk icons disappear, indicating that these objects have been removed.

7.3.1.7 Removing a VM Disk

1. **Select the failed VM disk by clicking the LEFT button on its icon.**

2. **From the Advanced-Ops menu, select Disk Group, then Remove Disks (see Figure 7-10).**
   The VM disk will disappear from the screen.

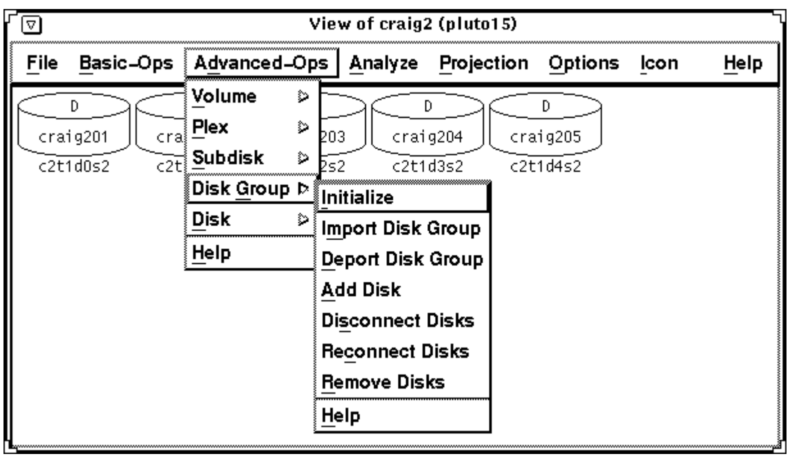

Figure 7-10 Accessing the Disk Group Menu

Note – This procedure removes a VM disk from a disk group only until the next reboot or re-initialization, at which time it will be brought back under Volume Manager control because the private region still exists on the physical disk. If you want to completely remove the disk from Volume Manager control, you must relabel the physical disk and remove the private region.

7.3.1.8 Starting a Volume

1. **Locate the simple or striped volume that contains the subdisk from the failed VM disk.**

2. **Before starting the volume, confirm that it is not started.**
   To do this, access its properties form by clicking RIGHT on its icon. The Utility State: field should indicate that the volume is unstartable (never been started) or startable (currently unstarted).

3. **Select the volume to be started by clicking the LEFT button on its icon.**

4. **From the Advanced-Ops menu, select Volume, then Start Volumes, and then Start.**
   The volume icon should rewrite itself as it starts.

5. **Confirm that the volume has now been started by accessing its properties form again.**
   The Utility State: field should now indicate that the volume is started.

7.3.1.9 Spinning Down All Drives in a Drive Tray

1. **Determine which drive tray you want to stop the drives in and click LEFT over that tray icon (see Figure 7-11).**
   The drive tray icon will change its color or pattern, indicating that it has been selected.

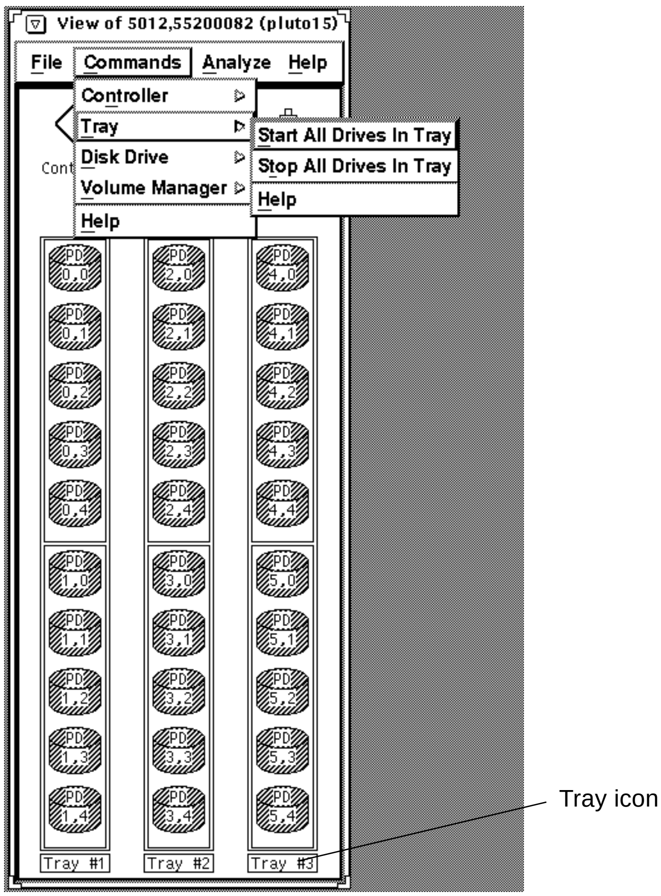

Figure 7-11 Accessing the Tray Menu

2. **From the Commands menu, select Tray, then Stop All Drives In Tray (see Figure 7-11).**

   All the drives in the tray will show an asterisk (*) next to the PD icon as they stop. When you see all the drives in the tray with an asterisk, you can remove the drive tray safely by following the instructions in the *SPARCstorage Array Service Manual*.

Note – If you make any changes to the Volume Manager configuration at this point (for example, if you create a subdisk), the stopped drives may start up again, even if the change that you made was not related to the stopped drives. If this happens, simply stop the started drives again.

7.3.2 After Replacing a Drive

After you have replaced the drive and secured the drive tray in the SPARCstorage Array, follow these instructions to start the drives in the drive tray:

1. **Determine which drive tray you want to start the drives in and click LEFT over that tray icon (see Figure 7-11).**

2. **From the Commands menu, select Tray, then Start All Drives In Tray (see Figure 7-11).**
   All the drives in the tray will lose the asterisk (*) next to the PD icon as they start.

7.4 When Replacing a Drive Tray in the SPARCstorage Array

There are two sets of tasks you should perform when you replace a drive tray in the SPARCstorage Array:

- *Before* replacing a drive tray — Section 7.4.1 on page 7-16
- *After* replacing a drive tray — Section 7.4.2 on page 7-17

7.4.1 Before Replacing a Drive Tray

If you are replacing a drive tray in the SPARCstorage Array, you must first perform the following software tasks and then stop all drives in the drive tray.

1. **Unmount the file system(s) on the disks in the tray that you are replacing.**

2. **Stop all *database* processes that are accessing the disks in the tray that you are replacing.**

3. **Stop all *other* processes that may be accessing the disks in the tray that you are replacing.**

4. **Flush or purge the outstanding writes from the NVRAM, if necessary.**
   Refer to Section 7.1, “Flushing or Purging Outstanding Writes from NVRAM,” on page 7-1 for more information.

7.4.1.1 Spinning Down All Drives in a Drive Tray

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 7-2).**
   This will bring up the View of SPARCstorage Arrays window.
2. **Determine which drive tray you want to stop the drives in and click LEFT over that tray icon (see Figure 7-11).**
   The drive tray icon will change its color or pattern, indicating that it has been selected.
3. **From the Commands menu, select Tray, then Stop All Drives In Tray (see Figure 7-11).**
   All the drives in the tray will show an asterisk (*) next to the PD icon as they stop. When you see all the drives in the tray with an asterisk, you can remove the drive tray safely.

Note – If you make any changes to the Volume Manager configuration at this point (for example, if you create a subdisk), the stopped drives may start up again, even if the change that you made was not related to the stopped drives. If this happens, simply stop the started drives again.

7.4.2 After Replacing a Drive Tray

You may have to start all the drives in the drive tray after you’re replaced the tray.

1. **Determine which drive tray you want to start the drives in and click LEFT over that tray icon (see Figure 7-11).**
2. **From the Commands menu, select Tray, then Start All Drives In Tray (see Figure 7-11).**
   All the drives in the tray will lose the asterisk (*) next to the PD icon as they start.

7.5 When Replacing an Array Controller

There are two sets of tasks you should perform when you replace an array controller in the SPARCstorage Array:

- *Before* replacing an array controller — page 7-18
- *After* replacing an array controller — page 7-19

7.5.1 Before Replacing an Array Controller

1. **Unmount the file system(s) on all disks in the array.**
2. **Stop all *database* processes that are accessing any disks in the array.**
3. **Stop all *other* processes that may be accessing any disks in the array.**
4. **Flush or purge the outstanding writes from the NVRAM, if necessary.**
   Refer to Section 7.1, "Flushing or Purging Outstanding Writes from NVRAM," on page 7-1 for more information.

7.5.1.1 Spinning Down All Drives in the SPARCstorage Array

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 7-2).**
   This will bring up the View of SPARCstorage Arrays window.
2. **From the Commands menu, select Controller, then Stop All Drives (see Figure 7-3).**
   All disk drives in the SPARCstorage Array are now stopped. You can now replace the array controller safely. Refer to the *SPARCstorage Array Service Manual* for those instructions.

Note – If you make any changes to the Volume Manager configuration at this point (for example, if you create a subdisk), the stopped drives may start up again, even if the change that you made was not related to the stopped drives. If this happens, simply stop the started drives again.

7.5.2 After Replacing an Array Controller

The worldwide name for a SPARCstorage Array is stored on a PROM in the array controller. When you replace an array controller, the worldwide name for the SPARCstorage Array will change to the worldwide name stored on the PROM on the new controller.

Because paths to volumes contain the worldwide name from the *old* array controller, you must go through the following procedure so that the paths to the volumes will be updated with the *new* array controller's worldwide name:

Caution – Do *not* change the worldwide name for the new controller back to the worldwide name used by the old controller.

1. **Become superuser.**

2. **Shut down your system:**

```
# shutdown -g0 -y -i0
```

Note – This is the correct shutdown command for most systems. However, if this does not work for your system, shut down your system as you normally would.

3. **Verify that the system goes to the** `ok` **prompt after the shutdown is complete.**
   If the system goes to the > prompt after the shutdown, enter `n` to get to the `ok` prompt.

4. **Reboot your system using the following command:**

```
ok boot -rs
```

Once the boot cycle is completed, you should see a screen similar to the following:

```
Type Ctrl-d to proceed with normal startup,
(or give root password for system maintenance):
```

5. **Enter your root password to become superuser, then enter the following command to boot the system a second time:**

```
# reboot
```

Once the system completes the second boot-up cycle, all the paths to the volumes should be updated to contain the new controller's worldwide name.

7.6 When Replacing All Other FRUs in the SPARCstorage Array

The software tasks contained in this section apply if you are replacing any of the following SPARCstorage Array FRUs:

- Fan tray
- Power supply
- Array controller
- FC/OM
- Battery module
- SPARCstorage Array backplane

You only need to perform certain software tasks *before* replacing a FRU in the SPARCstorage Array; no software tasks are necessary after you replace a FRU.

7.6.1 Before Replacing All Other FRUs

1. **Unmount the file system(s) on all disks in the array.**
2. **Stop all *database* processes that are accessing any disks in the array.**
3. **Stop all *other* processes that may be accessing any disks in the array.**
4. **Flush or purge the outstanding writes from the NVRAM, if necessary.**
   Refer to Section 7.1, "Flushing or Purging Outstanding Writes from NVRAM," on page 7-1 for more information.

7.6.1.1 Spinning Down All Drives in the SPARCstorage Array

1. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array that you want to access (see Figure 7-2).**
   This will bring up the View of SPARCstorage Arrays window.

2. **From the Commands menu, select Controller, then Stop All Drives (see Figure 7-3).**
   All disk drives in the SPARCstorage Array are now stopped. You can now replace the FRU safely.

Note – If you make any changes to the Volume Manager configuration at this point (for example, if you create a subdisk), the stopped drives may start up again, even if the change that you made was not related to the stopped drives. If this happens, simply stop the started drives again.

Part 3— Using the Command Line Interface

Getting Started 8

This chapter gives the basic information you will need to get your SPARCstorage Array up and running.

At this point, you can treat the physical disks in your SPARCstorage Array as you would any other set of physical disks. If you want to take full advantage of the disk management tools offered through the SPARCstorage Array, however, there are several options available:

- If you do not want to use the Volume Manager software but you want to perform the standard SPARCstorage Array software tasks, go to Section 8.1, "Performing Standard SPARCstorage Array Tasks," on page 8-2. Note that you can also use the options given in this section in conjunction with the Volume Manager software.
- If you want to take advantage of the disk management features offered through Volume Manager (such as striping and mirroring) on one or more of your SPARCstorage Array disks, you must bring these disks under Volume Manager control. (A complete explanation of these disk management features is given in Section 1.2, "Disk Management With the SPARCstorage Volume Manager," on page 1-4.) Read the instructions given in Section 8.2, "Bringing Physical Disks under Volume Manager Control," on page 8-6.

8.1 Performing Standard SPARCstorage Array Tasks

The standard SPARCstorage Array tasks are:

- Reserving and releasing drives
- Setting fast writes

These software features come standard with the SPARCstorage Array software, so you can perform them whether you want to bring the SPARCstorage Array disks under Volume Manager control or not.

- If you plan on having multiple hosts on your system and you want to set up your SPARCstorage Array so that all or some of the drives in the SPARCstorage Array report to only one host, refer to Section 8.1.1, "Reserving and Releasing Drives," on page 8-2.

- If you want to set up your system for fast writes, follow the instructions given in Section 8.1.2, "Setting Fast Writes," on page 8-5 (for an explanation of what fast write does, refer to Section 1.1.1, "Description of Fast Write," on page 1-2).

8.1.1 Reserving and Releasing Drives

The reserve command allows a host system to reserve individual drives or all the drives in a SPARCstorage Array so that no other host systems can use those drives. This is useful if you have more than one host system connected to a SPARCstorage Array but you want only one system to be able to access certain drives. The release command releases these drives from their reserved state. These commands implement the reserve and release functions as defined by the industry standard SCSI-2 specification.

8.1.1.1 Reserving and Releasing All Drives in a SPARCstorage Array

Following are the instructions for reserving and releasing all drives in a SPARCstorage Array.

Reserving All Drives in a SPARCstorage Array

The reserve option reserves all drives in a SPARCstorage Array for exclusive use by the issuing host.

♦ **Enter the following command at the prompt:**

```
ssaadm reserve controller
```

where `controller` is the logical name of the controller, given in the form `c#`. If any of the drives had previously been reserved by another system, this command will fail.

Note – For more information on the `ssaadm` command, see the `ssaadm(1M)` man page.

Releasing All Drives in a SPARCstorage Array

The release option releases all drives in a SPARCstorage Array from exclusive use by the issuing host. You would use this option if you had previously reserved all drives for exclusive use using the instructions given in the section entitled “Reserving All Drives in a SPARCstorage Array.”

♦ **Enter the following command at the prompt:**

```
ssaadm release controller
```

where `controller` is the logical name of the controller, given in the form `c#`. This command will have no affect on drives that were not reserved earlier.

8.1.1.2 Reserving and Releasing Single Disks

Following are the instructions for reserving and releasing a single disk.

Reserving a Physical Disk

The reserve option reserves a physical disk for exclusive use by the issuing host.

♦ **Enter the following command at the prompt:**

```
ssaadm reserve drive
```

where `drive` is the logical disk address for the drive, given in the form `/dev/rdsk/c#t#d#s#`. If the drive had previously been reserved by another system, this command will fail.

Releasing a Physical Disk

The release command releases a physical disk from exclusive use by the issuing host. You would use this option if you had previously reserved the physical disk for exclusive use using the instructions given in the section entitled “Reserving a Physical Disk.”

♦ **Enter the following command at the prompt:**

```
ssaadm release drive
```

where `drive` is the logical disk address for the drive, given in the form `/dev/rdsk/c#t#d#s#`. This command will have no affect on a drive that was not reserved earlier.

8.1.2 Setting Fast Writes

You can set fast writes from either the controller level, which would set fast writes for all the drives in the SPARCstorage Array, or from the drive level, which would set fast writes only for one specific drive.

- If you want to set fast writes from the controller level, refer to Section 8.1.2.1, “Setting Fast Writes at the Controller Level,” on page 8-5.
- If you want to set fast writes from the drive level, refer to Section 8.1.2.2, “Setting Fast Writes at the Drive Level,” on page 8-6.

8.1.2.1 Setting Fast Writes at the Controller Level

To set fast writes at the controller level:

♦ **Enter the following command at the prompt:**

```
ssaadm fast_write [-s] -e controller
```

The `-s` option causes the SPARCstorage Array to save the fast write option across power cycles. Your SPARCstorage Array is now set up so that all the drives within the array take advantage of the fast write option.

If you want to set fast writes for synchronous writes only, enter:

```
ssaadm fast_write [-s] -c controller
```

If at any point you want to stop fast writes for all the drives in the SPARCstorage Array, follow these instructions:

♦ **Enter the following command at the prompt:**

```
ssaadm fast_write [-s] -d controller
```

8.1.2.2 Setting Fast Writes at the Drive Level

To set fast writes at the drive level:

♦ **Enter the following command at the prompt:**

```
ssaadm fast_write [-s] -e drive
```

The `-s` option causes the SPARCstorage Array to save the fast write option across power cycles. The disk drive you selected is now set up so that it takes advantage of the fast write option. You can repeat this procedure for other drives, if you want set more than one drive for fast writes.

If you want to set fast writes for synchronous writes only, enter:

```
ssaadm fast_write [-s] -c drive
```

If at any point you want to stop fast writes for a single drive, follow these instructions:

♦ **Enter the following command at the prompt:**

```
ssaadm fast_write [-s] -d drive
```

8.2 Bringing Physical Disks under Volume Manager Control

When you add a disk to a system, you may wish to put the disk under control of the Volume Manager so that it can control the space allocation on the disk. If the disk was previously in use, but not under Volume Manager control, then you may wish to preserve existing data on the disk while still letting the Volume Manager take control of the disk. You can accomplish this by using the encapsulation function of the Volume Manager. If the disk is new, then it will need to be initialized. If a disk was previously not under Volume Manager control, but no data is required to be preserved, you should initialize it.

When you bring a disk under Volume Manager control, a small portion of space on that disk must be free so that the SPARCstorage Volume Manager software can create a private region, which is used to hold information necessary for Volume Manager to be run on that disk. If a disk has a swap partition (such as a boot disk), then Volume Manager will use part of the swap as the private region, so it doesn't need any free space on the boot disk in that case.

Note – If you have / and swap on separate disks and the boot device has no free space for the private region, you will not be able to encapsulate the boot disk.

When you perform disk administration, you need to recognize the difference between a *device name* and a *disk name.*

The *device name* (sometimes referred to as *devname* or *disk access name*) is the location of the disk. The syntax of a device name is `c`*n*`t`*n*`d`*n*`s`*n*, where:

- c*n* is the number of the controller to which the disk drive is attached.
- t*n* is the number of the target disk on that controller.
- d*n* is the number of the disk (or UNIX partition).

The full pathname of a device is `/dev/dsk/`*devicename*. In this document, only the device name is listed and `/dev/dsk` is assumed. An example of a device name is `c0t0d0`.

The *disk name* (sometimes referred to as *disk media name*) is an administrative name for the disk, such as *disk01*. If you do not assign a disk name, the disk name defaults to *disknn*, where *nn* is a sequence number if the disk is being added to `rootdg`. Otherwise, the default disk name is *groupnamenn*, where *groupname* is the name of the disk group to which the disk is added.

1. Add a disk by entering the command:

```
vxdiskadd devname
```

Note – For more information on the `vxdiskadd` command, see the `vxdiskadd(1M)` man page.

For example, to add the device *c1t0d0* to Volume Manager control, enter:

```
vxdiskadd c1t0d0
```

2. **`vxdiskadd` displays the following screen:**

```
Add or initialize a disk
Menu: Volume Manager/Disk/AddDisk

  Use this operation to add a disk to a disk group. You can select
  an existing disk group or create a new disk group. You can also
  initialize a disk without adding it to a disk group, which
  leaves the disk available for use as a replacement disk. This
  operation takes, as input, a disk device, for example c0t2d0,
  a disk group (or none to leave the disk available for as a
  replacement disk). If you are adding the disk to a disk group,
  you will be asked to give a name to the disk.

  Disk device c1t0d0 does not appear to have been initialized for
  use with the volume manager. However, it may have been
  initialized for other purposes. You may want to encapsulate the
  existing disk partitions as volumes instead of adding it as a
  new disk.

Do you wish to encapsulate c1t0d0? [y,n,q,?] (default: y)
```

If your disk is already partitioned and you want to keep the data as it is, press Return. If the disk has not been partitioned or if you don't need to keep the data on a previously partitioned disk, enter `n` and press Return.

3. **Regardless of your answer in the previous screen, the Volume Manager prompts you for a disk group name.**

```
  You can choose to add this disk to an existing disk group, to
  create a new disk group, or to leave the disk available for use
  by future add or replacement operations. To create a new disk
  group, select a disk group name that does not yet exist. To
  leave the disk available for future use, specify a disk group
  name of “none”.

Which disk group [<group>,none,list,q,?] (default: rootdg)
```

Press Return to assign the disk to the default disk group `rootdg`. Otherwise, enter the name of the disk group to which you want the disk assigned or enter `none` to assign the disk as a spare. You can create other disk groups as desired.

4. **The Volume Manager now prompts you for a disk name (unless you entered `none` for a disk group, since spare disks do not get named).**

```
  You must now select a disk name for the disk. This disk name
  can be specified to disk removal, move, or replacement
  operations. If you move the disk, such as between host bus
  adapters, the disk will retain the same disk name, even though
  it will be accessed using a different disk device address name.

Enter disk name [<name>,q,?] (default: disk01)
```

Enter a name that makes sense from an administrative point of view or press Return to accept the default disk name. If the disk you are adding is not the first disk in the disk group, the following screen is displayed:

```
Preserve this disk as hot-spared? [y,n,q?] (default: n)
```

Enter `y` if you want this disk to be reserved as an automatic replacement disk in case another disk should fail.

5. **If you selected to encapsulate the disk and chose the default disk group `rootdg` and the default disk name (in this case `disk01`), the following screen is displayed.**

```
  The requested operation is to encapsulate disk device c1t0d0
  and to add this device to disk group rootdg as disk disk01. The
  system must be rebooted before this can take effect.

Continue with operation? [y,n,q,?] (default: y)
```

If you are initializing a nonpartitioned disk or re-initializing a previously partitioned disk and selected the default disk group and disk name, the following screen appears:

```
  The requested operation is to initialize disk device c1t0d0 and
  to add this device to disk group rootdg as disk disk01.

Continue with operation? [y,n,q,?] (default: y)
```

In either case, press Return if you wish to continue with the chosen operation.

6. If you are encapsulating a disk, the following screen appears next:

```
  The first stage of encapsulation for c0t0d0 has completed
  successfully. You should now reboot your system at the earliest
  possible opportunity. The encapsulation will require two or
  three reboots which will happen automatically after the next
  reboot. To reboot execute the command:

      shutdown -g0 -y -i6

  This will update the /etc/vfstab file so that volume devices
  are used to mount the file systems on this disk device. You will
  need to update any other references such as backup scripts,
  databases, or manually created swap devices.

Encapsulate another disk? [y,n,q,?] (default: n)
```

Press Return to exit the `vxdiskadd` session. Volume Manager returns the following display to notify you that the initialization was a success.

```
  Disk initialization for c1t0d0 completed successfully.

Goodbye
```

Setting Up the Software Configuration 9

Note – You can only perform the tasks given in this chapter on disks that have been placed under Volume Manager control. Refer to Section 8.2, “Bringing Physical Disks under Volume Manager Control,” on page 8-6 for those instructions.

The different types of software configurations available are:

- Simple
- Striped (RAID 0)
- Mirrored (RAID 1)
- Striped and mirrored (RAID 0 + 1)
- RAID-5

Table 9-1 gives a brief description of each type of configuration. For a more detailed description and the pros and cons of each type, refer to the *SPARCstorage Array Configuration Guide*.

Table 9-1 Software Configurations Offered Through Volume Manager

Configuration	**Description**
Simple	Data is arranged both sequentially and contiguously over one or more physical disks.
Striped	Data is spread across relatively small, equally-sized fragments that are allocated alternately and evenly across multiple physical disks. Throughput increases with the number of physical disks across which data is striped. Striping helps to balance I/O load in cases where high traffic areas exist on certain physical disks.
Mirrored	Data is duplicated, or mirrored, over two or more physical disks. If one physical disk fails, you can use the mirrored disk as you would the original.
Striped and Mirrored	Data is first striped across multiple physical disks, and is then mirrored over two or more separate physical disks. This is useful if you want to take advantage of both the speed of striping and the data redundancy of mirroring.
RAID-5	Data is spread across relatively small, equally-sized fragments that are allocated alternately and evenly across multiple physical disks. Parity fragments are also created on the same disks, so that each stripe run contains the striped data and a single parity fragment. Compared to the performance of a striped configuration, throughput of RAID-5 configurations decreases, since parity information needs to be updated each time data is accessed. However, compared to mirroring, the amount of space used in RAID-5 is significantly smaller.

Table 9-2 gives a brief explanation of the components of the Volume Manager that are used to set up the configurations. For a more detailed description of Volume Manager and its components, refer to Chapter 1, "Disk Management for the SPARCstorage Array."

Table 9-2 Components of the Volume Manager

Components	Description
VM disk	When a partition from a physical disk is brought under Volume Manager control, Volume Manager creates a *VM disk* that corresponds to that partition.
Disk group	A collection of VM disks that have been grouped together because they have something in common. For example, you could put all the disks from a single SPARCstorage Array into one disk group.
Subdisk	A VM disk can be divided into one or more *subdisks*. Since a VM disk is assigned to a partition on a physical disk, all the subdisks within a VM disk are part of a single physical disk.
Plex	One or more subdisks can be grouped together to form a *plex*. The subdisks in a plex can be located on different VM disks.
Volume	One or more plexes can be grouped together to form a *volume*.

Figure 9-1 gives a graphical representation of the relationships between these Volume Manager components.

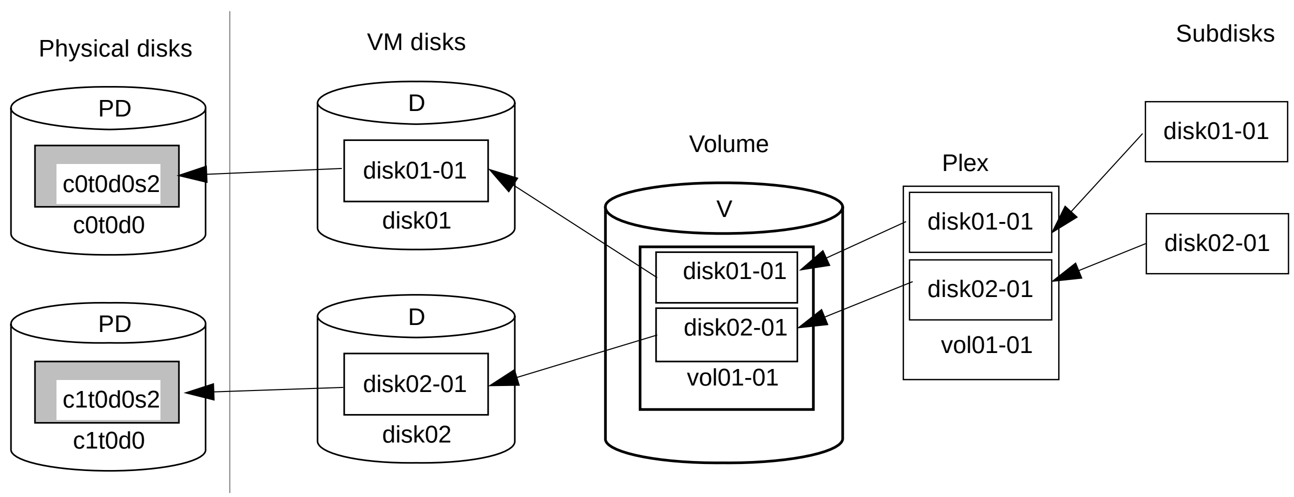

Figure 9-1 Graphical Representation of Volume Manager Components

Once you have decided upon the configuration that you want to use, you will use these components of the Volume Manager to set up that structure. For example, if you want to set up your system to stripe data across two physical disks, you and/or Volume Manager would:

1. Determine which physical disks you want the data striped across.
2. Locate the VM disks that are assigned to the partition on each physical disk.
3. Create one subdisk on each VM disk to be used for striping. Since you have two VM disks in this example, you would have two subdisks total, one on each VM disk.
4. Group the two subdisks into a single plex.
5. Create a volume to house this plex.

Following are the different software configurations you can create using the Volume Manager:

9.1 Setting Up a Simple Configuration

A simple configuration is one in which data is arranged both sequentially and contiguously over one or more physical disks. You can set up a simple configuration by creating a simple volume which can then be used for file systems, data bases, and so forth.

You can create a simple volume on a VM disk selected by Volume Manager or you can select the VM disk yourself.

9.1.1 Creating a Simple Volume on a VM Disk Selected by Volume Manager

♦ **To create a volume using system defaults, enter:**

```
vxassist make volume_name length
```

Note – For more information on the vxassist command, see the vxassist(1M) man page.

For example, to create the volume voldef enter:

```
vxassist make voldef 100m
```

This creates a 100 Mbyte volume named voldef. vxassist creates all the necessary subdisks, plex and volume objects. The objects are created on available disk space chosen by the Volume Manager.

9.1.2 Creating a Simple Volume on a VM Disk You Select

The Volume Manager automatically selects the disk or disks each volume will reside on, unless you specify otherwise. If you want a volume to reside on a specific disk, you must designate the disk for the Volume Manager.

♦ **To create a volume on a specific disk, enter:**

```
vxassist make volume_name length diskname [...]
```

You can specify more than one disk.

For example, to create the volume volspec on disk03, enter:

```
vxassist make volspec 30m disk03
```

vxassist creates all the necessary subdisks, plex and volume objects. The objects are created on available disk space chosen by the Volume Manager.

9.2 Setting Up a Striped Configuration

A striped configuration is one in which data is spread across relatively small, equally sized fragments that are allocated alternately and evenly across multiple physical disks. You can set up a striped configuration by creating a striped volume which can then be used for file systems, data bases, and so forth.

A striped volume consists of a number of equal sized subdisks, each located on a separate disk drive. For more information on striping, refer to Chapter 1, "Disk Management for the SPARCstorage Array."

Caution – There are several striped configurations that are not allowed or not recommended for the SPARCstorage Array. Refer to page 1-15 for general striping guidelines and to the section entitled "Boot Device Rules" on page 1-33 for boot device guidelines before proceeding with this section.

♦ **To create a striped volume, enter:**

```
vxassist make volume_name length layout=stripe
```

For example, to create the striped volume `volzebra`, enter:

```
vxassist make volzebra 100m layout=stripe
```

This creates a volume with the default stripe width on the default number of drives.

9.3 Setting Up a Mirrored Configuration

A mirrored configuration is one in which data is duplicated, or mirrored, over two or more physical disks. The mirror layout can be simple or striped.

A mirror is a copy of a volume. The mirror copy is not stored on the same disk(s) as the original copy of the volume. Mirroring a volume assures you that the data in that volume will not be lost if one of your disks fails.

Caution – There are several mirrored configurations that are not allowed or not recommended for the SPARCstorage Array. Refer to page 1-28 for general mirroring guidelines.

Note – If you are creating more than one mirror for a volume, you must wait until each mirror is completely made before you can create the next one.

♦ **To create a new volume with a mirror, enter:**

```
vxassist make volume_name length layout=mirror
```

For example, to create the mirrored volume, `volmir`, enter:

```
vxassist make volmir 50m layout=mirror
```

♦ **To create a mirror for an existing volume, enter:**

```
vxassist mirror volume_name
```

For example, to create a mirror of the volume `voltest`:

```
vxassist mirror voltest
```

♦ **To create a volume with dirty region logging enabled, create a mirrored volume with a dirty region log as follows:**

```
vxassist make volume_name length layout=mirror,log [nmirror=#] [nlog=#]
```

This specifies that the volume layout must be both mirrored and contain a DRL log. If you want more than two plexes or multiple log subdisks, the optional arguments `nmirror` and `nlog` can be used to specify the number of regular plexes and log plexes, respectively. `vxassist` creates one log plex per log subdisk, by default.

♦ **To create mirrors for all existing volumes using available disk space, enter:**

```
/etc/vx/bin/vxmirror -a
```

You can also configure the Volume Manager to create mirrored volumes by default. To do this, enter:

```
/etc/vx/bin/vxmirror -d yes
```

If you make this change, you can still make unmirrored volumes by specifying `nmirror=1` as an attribute to the `vxassist` command. For example, to create an unmirrored 20 Mbyte volume named `nomirror`, enter:

```
vxassist make nomirror 20m nmirror=1
```

9.4 Setting Up a RAID-5 Configuration

A RAID-5 configuration is one in which data is spread across relatively small, equally-sized fragments that are allocated alternately and evenly across multiple physical disks. Parity fragments are also created on the same disks, so that each stripe run contains the striped data and a single parity fragment.

♦ **To create a RAID-5 volume, enter:**

```
vxassist make volume_name length layout=raid5
```

For example, to create the RAID-5 volume `volraid`, enter:

```
vxassist make volraid 10m layout=raid5
```

This creates a RAID-5 volume with the default stripe width on the default number of drives.

Changing Components of the Software Configuration 10

This chapter gives the procedures for changing all or part of the software configuration. It is assumed that you understand the terminology presented in Chapter 9, “Setting Up the Software Configuration,” and set up your configuration based on the instructions given in that chapter.

10.1 Adding Components to a Configuration

10.1.1 Creating a Disk Group

1. **To create a disk group, enter:**

```
vxdiskadd devname
```

where *devname* is the device name of the disk that you will be adding to the new disk group.

Note – For more information on the `vxdiskadd` command, see the `vxdiskadd(1M)` man page.

For example, to create the disk group `newdg` associated with disk `c1t1d0`, enter:

```
vxdiskadd c1t1d0
```

2. **If c1t1d0 has already been initialized, the Volume Manager asks if you wish to reinitialize it.**

```
Add or initialize a disk Menu:
VolumeManager/Disk/AddDisk

  Use this operation to add a disk to a disk group. You can select
  an existing disk group or create a new disk group. You can also
  initialize a disk without adding it to a disk group, which
  leaves the disk available for use as a replacement disk. This
  operation takes, as input, a disk device, for example c0t2d0,
  a disk group (or none to leave the disk available for as a
  replacement disk). If you are adding the disk to a disk group,
  you will be asked to give a name to the disk.

  Disk device c1t1d0 appears to have been initialized already.
  The disk is currently available as a replacement disk.

Do you wish to reinitialize c1t1d0? [y,n,q,?] (default: y)
```

Enter n if you want to add the disk to the disk group without reinitializing it or press Return to reinitialize it.

3. **The Volume Manager prompts you for a disk group.**

```
  You can choose to add this disk to an existing disk group, to
  create a new disk group, or you can choose to leave the disk
  available for use by future add or replacement operations. To
  create a new disk group, select a disk group name that does not
  yet exist. To leave the disk available for future use, specify
  a disk group name of "none".

Which disk group [<group>,none,list,q,?] (default: rootdg) newdg
```

Enter the new disk group name and press Return (for the purposes of this exercise, newdg will be used as the name for the new disk group).

4. **The Volume Manager responds with:**

```
  There is no active disk group named newdg.

Create a new group named newdg? [y,n,q,?] (default: y)
```

Press Return to continue.

5. **The Volume Manager asks for a disk name.**

```
  You must now select a disk name for the disk. This disk name
  can be specified to disk removal, move, or replacement
  operations. If you move the disk, such as between host bus
  adapters, the disk will retain the same disk name, even though
  it will be accessed using a different disk device address name.

Enter disk name [<name>,q,?] (default: newdg01)
```

Enter the disk name of your choice or press Return to select the default name.

6. The Volume Manager displays a confirmations window:

```
  The requested operation is to create a new disk group named
  newdg containing disk device c1t1d0. The disk will be named
  newdg01 within the disk group.

Continue with operation? [y,n,q,?] (default: y)
```

Press Return to continue.

7. Once the operation is complete, the Volume Manager returns the following display:

```
  Disk initialization for c1t1d0 completed successfully.

Goodbye.
```

8. Enter:

```
vxdisk list
```

to see if the disk group was created.

```
DEVICE       TYPE       DISK          GROUP       STATUS
c0t0d0s2     sliced     c0t0d0s2      rootdg      online
c1t0d0s2     sliced     disk03        rootdg      online
c1t1d0s2     sliced     disk02        newdg       online
```

Note – For more information on the `vxdisk` command, see the `vxdisk(1M)` man page.

10.1.2 Moving Disk Groups Between Systems

An important feature of disk groups is that they can be moved between systems. If all disks in a disk group are moved from one system to another, then the disk group can be used by the second system without having to respecify the configuration.

To move a disk group between systems:

1. **On the first system, stop and unmount all volumes in the disk group.**
2. **Deport (disable local access to) the disk group with the command:**

```
vxdg deport diskgroupname
```

3. **Import (enable local access to) the disk group and its disks from the second system with the command:**

```
vxdg import diskgroupname
```

4. **After the disk group is imported, start all volumes in the disk group with the command:**

```
vxrecover -g diskgroupname -sb
```

You may want to move disks from a system that has crashed. In this case, you will not be able to deport the disk group from the first system. When a disk group is created or imported on a system, that system writes a lock on all disks in the disk group. The purpose of the lock is to ensure that dual-ported disks (disks that can be accessed simultaneously by two systems) will not be used by both systems at the same time. If two systems try to manage the same disks at the same time, configuration information stored on the disk will be corrupted and the disk will become unusable.

If you move disks from a system that has crashed or failed to detect the group before the disk is moved, the locks stored on the disks will remain and must be cleared. The system returns the following error message:

```
vxdg:disk group groupname: import failed: Disk is in use by
another host
```

To clear locks on a specific set of devices, use the command:

```
vxdisk clearimport diskdevicename ...
```

Be careful when using this command on systems that really do have dual-ported disks.

In some cases, you may want to import a disk group when some disks are not available. The `import` operation normally fails if some disks for the disk group cannot be found among the disk drives attached to the system. If the `import` operation fails, one of the following error messages will display:

```
vxdg: Disk group groupname: import failed: Disk for disk group
not found
```

or

```
vxdg: Disk group groupname: import failed: Disk group has no valid
configuration copies
```

If some of the disks in the disk group have failed, you can force the disk group to be imported with the command:

```
vxdg -f import diskgroupname
```

All of these operations can be done using `vxdiskadm`. To deport a disk group in `vxdiskadm` select item 9, `Remove access to (deport) a disk group`. To import a disk group, select item 8, `Enable access to (import) a disk` group. The vxdiskadm import operation checks for host import locks and asks if you want to clear any that are found. It also starts volumes in the disk group.

10.1.3 Adding a Physical Disk to a Disk Group

You may wish to add a new disk to an already established disk group. Perhaps the current disks have insufficient space for the application or work group requirements, especially if these requirements have changed.

1. To add an initialized disk to a disk group, enter:

```
vxdiskadd devname
```

where *devname* is the device name of the disk that you want to add to the disk group.

Note – For more information on the vxdiskadd command, see the vxdiskadd(1M) man page.

For example, to add device c1t1d0 to rootdg, enter:

```
vxdiskadd c1t1d0
```

2. **The Volume Manager displays the following message:**

```
Add or initialize a disk Menu:
VolumeManager/Disk/AddDisk

  Use this operation to add a disk to a disk group. You can select
  an existing disk group or create a new disk group. You can also
  initialize a disk without adding it to a disk group, which
  leaves the disk available for use as a replacement disk. This
  operation takes, as input, a disk device, for example c0t2d0,
  a disk group (or none to leave the disk available for as a
  replacement disk). If you are adding the disk to a disk group,
  you will be asked to give a name to the disk.

  Disk device c1t1d0 appears to have been initialized already.
  The disk is currently available as a replacement disk.

Do you wish to reinitialize c1t1d0? [y,n,q,?] (default: y)
```

Enter `n` if you want to add the disk to the disk group without reinitializing it or press Return to reinitialize it.

3. **The Volume Manager now asks you to choose a disk group.**

```
  You can choose to add this disk to an existing disk group, to
  create a new disk group, or you can choose to leave the disk
  available for use by future add or replacement operations. To
  create a new disk group, select a disk group name that does not
  yet exist. To leave the disk available for future use, specify
  a disk group name of "none".

Which disk group [<group>,none,list,q,?] (default: rootdg)
```

Enter a name that makes sense from an administrative point of view or press Return to accept the default disk group name.

Caution – If you are bringing your boot disk under Volume Manager control, you *must* put it into the `rootdg` disk group or a system error may result.

4. **The Volume Manager prompts you for a disk name.**

```
  You must now select a disk name for the disk. This disk name
  can be specified to disk removal, move, or replacement
  operations. If you move the disk, such as between host bus
  adapters, the disk will retain the same disk name, even though
  it will be accessed using a different disk device address name.

Enter disk name [<name>,q,?] (default: disk03)
```

Either enter a disk name or press Return to select the default disk name.

If the disk you are adding is not the first disk in the disk group, the following screen appears:

```
Preserve this disk as hot-spared? [y,n,q?] (default: n)
```

Enter `y` if you want this disk to be reserved as an automatic replacement disk in case another disk should fail. Disks designated as hot spares cannot have storage space allocated on them and do not participate in the free space model.

5. **The following verification screen is displayed:**

```
  The requested operation is to add disk device c1t1d0 to disk
  group rootdg as disk disk03.

Continue with operation? [y,n,q,?] (default: y)
```

Press Return to continue.

6. **The Volume Manager adds the disk to disk group `rootdg`, displays a success message, and exits.**

```
  Disk initialization for c1t1d0 completed successfully.

Goodbye.
```

To see that the disk has been added to the disk group, enter:

```
vxdisk list
```

The Volume Manager returns:

```
DEVICE       TYPE      DISK          GROUP      STATUS
c0t0d0s2     sliced    c0t0d0s2      rootdg     online
c1t0d0s2     sliced    disk01        rootdg     online
c1t1d0s2     sliced    disk03        rootdg     online
```

10.1.4 Moving a VM Disk to a Different Disk Group

If you want to reorganize by moving a disk between disk groups, remove the disk from one disk group and add it to the other. For example, to move the physical disk `c0t3d0` (attached with the disk name `disk04`) from disk group `rootdg` and add it to disk group `mktdg`, you could use the commands:

```
vxdg rmdisk disk04
vxdg -g mktdg adddisk mktdg02=c0t3d0
```

Note – This procedure does not save the configurations or data on the disks.

This can also be done using `vxdiskadm` by selecting item 3, `Remove a disk`, from the main menu, and then selecting item 1 (`Add or initialize a disk`).

10.1.5 Adding a VM Disk as a Hot Spare

You can add one or more VM disks to a disk group as hot spares. Hot spares are used to automatically recover data when a physical disk fails. The hot spare will only be effective for data that is on a mirrored volume; if one of the two disks from the mirrored volume fails and a hot spare is present in the same disk group, then the information from the failed disk is automatically placed on the hot spare. For more information on hot spares, refer to “Hot-Sparing” on page 1-29.

♦ **To add a disk as a hot spare, enter:**

```
vxedit set spare=on diskname
```

Note – For more information on the `vxedit` command, see the `vxedit(1M)` man page.

For example, to add disk `disk01` as a hot spare, enter:

```
vxedit set spare=on disk01
```

♦ **To remove the hot spare designation, enter:**

```
vxedit set spare=off diskname
```

For example, to make disk `disk01` available for normal use (remove the hot spare designation), enter:

```
vxedit set spare=off disk01
```

10.1.6 Adding a File System to an Existing Volume

Note – The instructions in this section use the `mkfs` command to create a file system. You can also use the `newfs` command to create a file system, since this is a front-end to the `mkfs` program. For more information on the `newfs` command, refer to the `newfs(1M)` man page.

A file system cannot be larger than the volume on which it is created.

♦ **To create a file system on a volume, enter:**

```
mkfs [generic_options] [-o specific_options] special_file size
```

where *generic_options* are the common file system options (such as -g), *specific_options* are options specific to this file system, *special_file* is the full pathname of the volume on which to create the file system (in the form `/dev/vx/<dsk or rdsk>/diskgroup/volume`), and *size* is the size of the new file system.

Note – For more information about the options and variables available for use with the `mkfs` command, see the `mkfs(1M)` manual page.

For example, to create a file system on a volume called `/dev/vx/rdsk/pubsgroup/pubs`, first create the volume called `pubs` using `vxassist`. Once the volume is created, enter:

```
mkfs /dev/vx/rdsk/pubsgroup/pubs 12288
```

to create the file system.

Note – The size of the file system (in this case 12288 sectors) must be the same size or smaller than the volume on which the file system is mounted.

You should see the following message:

```
12288 sectors, 6144 blocks of size 1024
1320 inodes, 5294 data blocks, log size 512
1 allocation units of 5630 blocks, 1320 inodes, 5294 data blocks
first allocation unit starts at block 514
overhead per allocation unit is 336 blocks
```

10.1.7 Mounting a File System

♦ **To mount a file system, enter:**

```
mount [generic_options] [-o specific_options] block_special
mount_point
```

where block_special is in the form /dev/vx/<dsk or rdsk>/diskgroup/volume.

For example, to mount the file system /dev/vx/rdsk/pubsgroup/pubs, enter:

```
mount /dev/vx/rdsk/pubsgroup/pubs /pubs
```

10.1.8 Adding a Mirror to a Volume

A mirror is a copy of a volume. The mirror copy is not stored on the same disk(s) as the original copy of the volume. Mirroring a volume assures you that the data in that volume will not be lost if one of your disks fails.

Note – If you are creating more than one mirror for a volume, you must wait until each mirror is completely made before you can create the next one.

♦ **To create a mirror for an existing volume, enter:**

```
vxassist mirror volume_name
```

For example, to create a mirror of the volume voltest, enter:

```
vxassist mirror voltest
```

Note – For more information on the vxassist command, see the vxassist(1M) man page.

Another way to mirror an existing volume is by first creating a plex and then associating it to a volume, using the following commands:

```
vxmake plex plex_name sd=subdisk_name ...
vxplex att volume_name plex_name
```

10.1.9 Adding a Mirror to a Root Volume

Adding a mirror to a root volume makes it possible to recover from failure of your boot disk by replacing it with the mirror of the boot disk.

1. **Select a disk that is at least as large as your boot disk.**
2. **Use the vxdiskadd command to add the selected disk as a new disk (if it is not already added).**
3. **Enter:**

```
/etc/vx/bin/vxrootmir alternate_disk
```

where *alternate_disk* is the disk name assigned to the disk. `vxrootmir` will create mirrors for the following volumes:

- `rootvol` — the volume for the root file system
- `swapvol` — the primary paging device
- `usr` — the volume for the `/usr` file system
- `var` — the volume for the `/var` file system

The alternate boot disk will be configured to enable booting from it if the primary boot disk fails.

There may be other volumes on the boot disk, such as volumes for `/home` or `/tmp` file systems. These must be mirrored separately using the `vxassist` utility. For example, if you have a `/home` file system on a volume `homevol`, you can mirror it to *alternate_disk* using the command:

```
vxassist mirror homevol alternate_disk
```

If you do not have space for a copy of some of these file systems on your alternate boot disk, you can mirror them to other disks. You can also span or stripe these other volumes across other disks attached to your system.

♦ **To list all volumes on your primary boot disk, enter:**

```
vxprint -t -v -e 'aslist.aslist.sd_disk="boot_disk_name"'
```

♦ **To mirror all of the simple volumes on this disk to your alternate boot disk, use the command:**

```
/etc/vx/bin/vxmirror boot_disk_name alternate_disk
```

10.1.10 Moving a Plex

Moving a plex copies the data content from the original plex onto a new plex, then the old plex is removed. In order for a move operation to be successful, the following criteria must be met:

- The old plex must be an active part of an active (enabled) volume.
- The new plex should be at least the same size or larger than the old plex.
- The new plex must not be associated with another volume.

The size of the plex has several important implications. If the new plex is smaller, or more sparse, than the original plex, you will get an incomplete copy of the data on the original plex. If this is the desired action, then the `-o force` option is required. If the new plex is longer, or less sparse, than the original plex, the data that exists on the original plex will be copied onto the new plex. Any area that was not on the original plex but is represented on the new plex will be filled from any other complete plexes which may be associated with the same volume. If the new plex is longer than the volume itself, then the remaining area of the new plex above the size of the volume will not be initialized.

The command to move data from one plex to another is:

```
vxplex mv original_plex new_plex
```

10.1.11 Copying a Plex

This operation copies the contents of a volume onto a specified plex. The volume to be copied must not be enabled. The plex must not be associated with any other volume. To copy a plex, enter:

```
vxplex cp vol_name new_plex
```

After the copy operation is complete, *new_plex* will not be associated with the specified volume *vol_name*. The plex contains a complete copy of the volume data. The plex that is being copied should be the same size or larger than the volume, otherwise you will get an incomplete copy of the data. For this same reason, *new_plex* also should not be sparse.

10.1.12 Attaching a Plex

When a disk has been repaired or replaced and is again ready for use, the plexes must be put back online (plex state set to active).

If the volume is currently enabled, use the following command:

```
vxplex att volume_name plex_name ...
```

For example, the command line for a plex labeled `vol01-02` on a volume labeled `vol01` looks like this:

```
vxplex att vol01 vol01-02
```

This revives the plex and, after the revival is complete, sets the plex utility state to active.

If the volume in not in use (not enabled), enter:

```
vxmend on plex_name
```

For example, the command line for a plex labeled `vol01-02` looks like this:

```
vxmend on vol01-02
```

In this case, the state of `vol01-02` is set to stale, so that when the volume is next started, the data on the plex will be revived from the other plex, and incorporated into the volume with its state set to active.

If it becomes necessary to manually change the state of a plex, refer to Section 11.2, “Recovering Volumes,” on page 11-8. See the `vxmake` and `vxmend` manual pages for more information about these commands.

10.1.13 Creating a Plex

Plexes are created by identifying subdisks and associating them to the plex that you want to create. To create a plex from existing subdisks, enter:

```
vxmake plex plex_name sd=subdisk_name,...
```

For example, the command to create a plex labeled `vol01-02` using two existing subdisks labeled `disk02-01` and `disk02-02` would be:

```
vxmake plex vol01-02 sd=disk02-01,disk02-02
```

10.1.14 Associating a Plex

Once a plex has been created and has at least one associated subdisk, you can associate that plex with a volume. A volume is of little use until it has at least one associated plex. When multiple plexes are associated with a volume, mirroring is in effect. If the volume is already started and an additional plex is being associated, the Volume Manager will update the newer plex by copying all necessary data to that plex. This operation may take some time.

You can associate several plexes at a time with a single volume, but you can select only one volume per association operation.

To associate a plex with an existing volume, enter:

```
vxplex att volume_name plex_name
```

For example, the command to associate a plex labeled `vol01-02` with a volume labeled `vol01` would be:

```
vxplex att vol01 vol01-02
```

Alternately, if the volume has not been created, a plex (or multiple plexes) can be associated with the volume to be created as part of the volume create command:

```
vxmake -U usetype vol vol_name plex=plex_name1, plex_name2...
```

For example, the command to create a mirrored, fsgen-type volume labeled `home` and associate two existing plexes labeled `home-1` and `home-2` would be:

```
vxmake -Ufsgen vol home plex=home-1,home-2
```

10.1.15 Moving a Subdisk

Moving a subdisk copies the disk space contents of a subdisk onto another subdisk. If the subdisk being moved is associated with a plex, then the data stored on the original subdisk is copied to the new subdisk, the old subdisk is dissociated from the plex, and the new subdisk is associated with the plex, at the same offset within the plex as the source subdisk.

For the subdisk move operation to perform correctly, the following conditions must be met:

- The subdisks involved must be the same size.
- The subdisk being moved must be part of an active plex on an active (enabled) volume.
- The new subdisk must not be associated with any other plex.

To move a subdisk, enter:

```
vxsd mv old_subdisk_name new_subdisk_name
```

10.1.16 Creating a Subdisk

To create a subdisk, enter:

```
vxmake sd subdisk_name disk_name, offset, len
```

where:

- *subdisk_name* is the name of the subdisk
- *disk_name* is the disk media name
- *offset* is the starting point (offset) of the subdisk within the disk
- *len* is the length of the subdisk

For example, the command line to create a subdisk labeled `disk02-01` that starts at the beginning of disk `disk02` and has a length of 8000 blocks looks like this:

```
vxmake sd disk02-01 disk02,0,8000
```

Note – Commands take sizes in blocks. Adding a suffix changes the unit of measure. A k suffix specifies 1024-byte blocks. To preserve (encapsulate) data that exists on the disk, a plex and volume must be created to cover that data.

10.1.17 Associating a Subdisk

Associating a subdisk with a plex places the amount of disk space defined by the subdisk at a specific offset within the plex. In all cases, the entire area that the subdisk fills must not be occupied by any portion of another subdisk. There are several different ways that subdisks can be associated with plexes, depending on the overall state of the configuration.

If you have already created all the subdisks needed for a particular plex, subdisks are associated at plex creation by using entering the command:

```
vxmake plex plex_name sd=subdisk_name, ...
```

For example, the following command creates the plex home-1 and associates subdisks disk02-01, disk02-00 and disk02-02 with the plex home-1 during the plex creation process:

```
vxmake plex home-1 sd=disk02-01,0s02-00,0s02-01
```

Subdisks are associated in order starting at offset 0. Using a command like this eliminates the need to specify the multiple commands necessary to create the plex and then associate each of the subdisks with that plex. In the previous example, the subdisks are associated to the plex in the order they are listed (after the sd=); the disk space defined as disk02-01 will be first, then 0s02-00, and then 0s02-01.

This method of associating subdisks is convenient during initial configuration. Subdisks can also be associated with a plex that already exists. One or more subdisks can be associated with an existing plex as follows:

```
vxsd assoc plex_name sd_name [sd_name2 sd_name3 ...]
```

For example, the command line to associate subdisks labeled disk02-01, 0s02-00, and 0s02-01 with a plex labeled home-1 looks like this:

```
vxsd assoc home-1 disk02-01 0s02-00 0s02-01
```

If the plex is not empty, the new subdisks are added after any subdisks that are already associated with the plex, unless the -l option is specified with the command. The -l option provides a way to associate subdisks at a specific offset within the plex.

The -l option is needed in a case where you have created a sparse plex for a particular volume and you want to make this plex complete. To make the plex complete, it is necessary to create a subdisk of exactly the size needed to fill the hole in the sparse plex, and then associate the subdisk with the plex by specifying the offset of the beginning of the hole in the plex. Use the following command to accomplish this task:

```
vxsd -l offset assoc sparse_plex_name exact_size_subdisk
```

Note – The subdisk must be exactly the right size because Volume Manager does not allow for the space defined by two subdisks to overlap within a single plex.

10.2 Removing Components from a Configuration

10.2.1 Removing a VM Disk From a Disk Group

A disk that contains no subdisks can be removed from its disk group with the command:

```
vxdg rmdisk diskname
```

For example, to remove disk02, enter:

```
vxdg rmdisk disk02
```

If the disk has subdisks on it when you try to remove it, the following error message is displayed:

```
vxdg:Disk diskname is used by one or more subdisks
```

Use the -k option with the vxdg command to remove the device assignment. Using the -k option allows you to remove the disk in spite of the presence of subdisks.

In some cases, you may want to remove a disk on which some subdisks are defined. For example, you may have three disks on one system and you may want to consolidate all of the volumes onto one disk. If you use vxdiskadm to remove a disk, you can choose to move volumes off that disk. To do this, run vxdiskadm and select item 3 (Remove a disk) from the main menu. If the disk is used by some subdisks, then a screen resembling the following is displayed:

```
The following subdisks currently use part of disk disk02:

   home usrvol

Subdisks must be moved from disk02 before it can be removed.

Move subdisks to other disks? [y,n,q,?] (default: n)
```

If you choose y, then all subdisks will be moved off the disk, if possible. Some subdisks may not be movable. The most common reasons why a subdisk may not be movable are:

- There is not enough space on the remaining disks.
- Plexes or striped subdisks cannot be allocated on different disks from existing plexes or striped subdisks in the volume.

If vxdiskadm cannot move some subdisks, you may need to remove some plexes from some disks to free more space before proceeding with the disk removal operation.

10.2.2 Unmounting a File System

File system administration often requires the unmounting of file systems. You can unmount a file system from a volume as long as the mount point is not currently in use. A volume icon containing a mounted file system must be selected in order for this operation to succeed.

If you no longer need to access the data in a file system, you can unmount it.

♦ **To unmount a file system, enter:**

```
umount block_special | mount_point
```

where block_special is in the form /dev/vx/<dsk or rdsk>/diskgroup/volume.

For example, to unmount file system `/dev/vx/dsk/pubsgroup/pubs`, enter:

```
umount /dev/vx/dsk/pubsgroup/pubs
```

10.2.3 Deporting a Disk Group

Deporting a disk group does not actually remove a disk group; it disables access to that disk group. You can deport a disk group so that its last VM disk can be removed, thus allowing the physical disk assigned to that VM disk to be reused, reinitialized, or added to another disk group. Once deported, the disk group is inaccessible. However, the partition retains its disk access record and knowledge of the deported disk group until it is reused (assigned to another disk group) or removed. A deported disk group can be re-imported later to the same host system or can be imported by a different host system running the Volume Manager.

To deport a disk group, unmount and stop any volumes in the disk group and then run the command:

```
vxdg deport diskgroupname
```

10.2.4 Taking a Physical Disk Offline

Occasionally, you may need to take a physical disk offline. If the disk is corrupted, you need to disable it and remove it. You also must disable a disk before moving the physical disk device to another location to be connected to another system.

To take a physical disk offline, first remove the disk from its disk group. Then place a disk in an “offline” state:

```
vxdisk offline devname
```

Note – For more information on the `vxdisk` command, see the `vxdisk(1M)` man page.

To take the device `c1t1d0s2` offline, enter:

```
vxdisk offline c1t1d0s2
```

10.2.5 Removing a Physical Disk from Volume Manager Control

You can remove a disk to move it to another system or you can remove the disk because the disk is failing or has failed. However, before removing the disk from the current system, you must:

- Unmount the file system (see Section 10.2.2, "Unmounting a File System," on page 10-23).
- Stop the volumes.
- Move the volumes to other disks or back up the volumes to tape (see later in this chapter). To move a volume, mirror the volume on one or more other disks, then remove the original copy of the volume.

Alternatively, if the volumes are no longer needed, you can remove them.

Removing a disk involves the following steps:

1. Remove the disk from its disk group:

```
vxdg [-g groupname] rmdisk diskname
```

where *groupname* is the name of the group to which the disk belongs and *diskname* is the name of the disk to be removed.

For example, to remove `disk01` from `rootdg`, enter:

```
vxdg rmdisk disk01
```

Since `rootdg` is the default disk group, you do not need to specify it.

If the disk has subdisks on it when you try to remove it, the following error message is displayed:

```
vxdg:Disk diskname is used by one or more subdisks
```

Use the `-k` option to `vxdg` to remove device assignment. Using the `-k` option allows you to remove the disk in spite of the presence of subdisks. For more information, see the `vxdg`(1m) manual page.

Note – For more information on the `vxdg` command, see the `vxdg(1M)` man page.

2. Remove the disk from the Volume Manager and the system.
After removing a disk from its disk group, remove it from the system:

```
vxdisk rm devname
```

For example, to remove `c1t0d0` from Volume Manager control, enter:

```
vxdisk rm c1t0d0s2
```

Note – This procedure removes a VM disk from a disk group only until the next reboot or re-initialization, at which time it will be brought back under Volume Manager control because the private region still exists on the physical disk. If you want to completely remove the disk from Volume Manager control, you must relabel the physical disk and remove the private region.

In some cases, you may want to remove a disk on which some subdisks are defined. For example, you may have three disks on one system and you may want to consolidate all of the volumes onto one disk. If you use `vxdiskadm` to remove a disk, you can choose to move volumes off that disk. To do this, run vxdiskadm and select item 3 (Remove a disk) from the main menu. If the disk is used by some subdisks, then a screen resembling the following is displayed:

```
The following subdisks currently use part of disk disk02:

     home usrvol

Subdisks must be moved from disk02 before it can be removed.

Move subdisks to other disks? [y,n,q,?] (default: n)
```

If you choose y, then all subdisks will be moved off the disk, if possible. Some subdisks may not be movable. The most common reasons why a subdisk may not be movable are:

- There is not enough space on the remaining disks.
- Plexes or striped subdisks cannot be allocated on different disks from existing plexes or striped subdisks in the volume.

If vxdiskadm cannot move some subdisks, you may need to remove some plexes from some disks to free more space before proceeding with the disk removal operation. Refer to Section 10.2.6, “Removing a Volume,” and Section 10.2.7, “Removing a Plex/Mirror or RAID-5 Log,” for information on how to remove volumes and plexes.

10.2.6 Removing a Volume

Once a volume is no longer necessary (it is inactive and archived, for example), you can remove the volume and free up the disk space for other uses.

1. **Remove all references to the volume.**
2. **If the volume is mounted as a file system, unmount it:**

```
umount /dev/vx/dsk/volume_name
```

3. **If the volume is listed in** /etc/vfstab**, remove its entry.**

4. **Make sure the volume is stopped with the command:**

```
vxvol stop
```

The `vxvol stop` command stops all Volume Manager activity to the volume.

5. **Remove the volume:**

```
vxedit -rf rm volume_name
```

For example, to remove the volume `volspan`, enter:

```
vxedit -rf rm volspan
```

Note – For more information on the `vxedit` command, see the `vxedit(1M)` man page.

10.2.7 *Removing a Plex/Mirror or RAID-5 Log*

When a plex/mirror is no longer needed, it can be removed. Examples of operations that require a plex/mirror to be removed are:

- Providing free disk space.
- Reducing the number of plexes/mirrors in a volume in order to increase the length of another mirror and its associated volume. The plexes and subdisks are removed, then the resulting space can be added to other volumes.
- Removing a temporary plex/mirror that was created to backup a volume and is no longer required.
- Changing the layout of a plex from concatenated to striped, or vice versa.

Removing a plex/mirror or RAID-5 log involves first dissociating it from its volume and then removing the plex/mirror or log and any associated subdisks completely.

Note – The last valid plex/mirror in a started or enabled volume cannot be removed, unless the volume is stopped before the remove operation.

To identify a plex/mirror or RAID-5 log, enter the following command:

```
vxprint -ht volume_name
```

The output would list a plex/mirror as the following:

```
pl vol02-01     vol02       ENABLED  5        -        ACTIVE    -       -
sd craig07-01   vol02-01    ENABLED  5        0        -         -       -
```

and a RAID-5 log would be displayed as the following:

```
pl vol01-02     vol01       ENABLED  1008     -        LOG       -       -
sd craig05-01   vol01-02    ENABLED  1008     0        -         -       -
```

To dissociate and remove the plex/mirror or RAID-5 log from its volume, enter:

```
vxplex -o rm dis plex_name
```

For example, to dissociate and remove the RAID-5 log `vol01-02`, enter:

```
vxplex -o rm dis vol01-02
```

10.2.8 Detaching a Plex

A detached plex is inaccessible for reads and writes, but is still associated with a volume. One or more plexes can be detached from their parent volume at a time. The volume must be started before a plex can be detached.

Note – This operation is not permitted when the specified plex is the last valid plex on the volume.

To temporarily detach one plex in a mirrored volume, use the following command:

```
vxplex det plex_name
```

For example, the command to temporarily detach a plex labeled `vol01-02` and place it in maintenance mode would be:

```
vxplex det vol01-02
```

This command temporarily detaches the plex, but maintains the association between the plex and its volume; however, the plex will not be used for I/O. A plex detached with the preceding command will be recovered on a system reboot. The plex state is set to stale, so that if a `vxvol start` command is run on the appropriate volume (for example, on system reboot), the plex will be revived and made active.

When you want the plex to return as an active part of its volume, follow this procedure:

If the volume is not enabled, start it using:

```
vxstart vol_name
```

If it is unstartable, set one of the plexes to CLEAN using:

```
vxmend mirror clean plex_name
```

and then start the volume.

If the plex does not yet have a *kernel state* of enabled, enter:

```
vxplex att volume_name plex_name ...
```

As with returning an offline plex to active, this command revives the plexes stated, and when each revive completes, sets the plex state to active.

10.2.9 Dissociating a Subdisk

To break an established relationship between a subdisk and the plex to which it belongs, the subdisk is *dissociated* from the plex. A subdisk is dissociated when the subdisk is to be removed or used in another plex. To dissociate a subdisk, enter:

```
vxsd dis subdisk_name
```

To dissociate a subdisk labeled disk02-01 from the plex with which it is currently associated, enter:

```
vxsd dis disk02-01
```

10.2.10 Removing a Subdisk

You would usually remove a subdisk when you are making changes to the system configuration. To remove a subdisk, first dissociate the subdisk by using the instructions in Section 10.2.9, "Dissociating a Subdisk." Then enter:

```
vxedit rm subdisk_name
```

For example, the command line to remove a subdisk labeled `disk02-01` would be:

```
vxedit rm disk02-01
```

10.3 Renaming Components of a Configuration

10.3.1 Renaming a Disk Group

Since only one disk group of a given name can exist per system, you must rename a disk group if you want to move it to a system already containing a disk group with the same name.

Every system running the Volume Manger must have a single `rootdg` disk group. `rootdg` can therefore only be moved across systems if it is renamed.

The following set of steps can be used if you want to temporarily move the `rootdg` disk group from one host to another (for repair work on the root volume, for instance) and then move it back:

1. **Identify the disk group ID of the root disk group to be imported:**

```
vxdisk -s list
```

2. **Import and rename the rootdg disk group:**

```
vxdg -tC -n newname import diskgroup
```

where:

- `-t` indicates a temporary import name
- `-C` clears import locks
- `-n` specifies a temporary name for the `rootdg` to be imported (so that it does not conflict with the existing `rootdg`)
- *`diskgroup`* is the ID of the `rootdg` disk group being imported

3. **After the necessary work has been done on the imported `rootdg`, deport the `rootdg`:**

```
vxdg -h newhost_ID deport diskgroup
```

where *`newhost_ID`* is the host ID of the system whose `rootdg` is being returned.

10.3.2 Renaming a VM Disk

It isn’t necessary to give your disks special names. The Volume Manager gives the disk a default name when you add the disk to Volume Manager control. The disk name is used by the Volume Manager to identify the disk’s location or type. If you wish to change the disk name to reflect a change of ownership or use, enter:

```
vxedit rename old_diskname new_diskname
```

To rename `disk01` to `disk03`, enter:

```
vxedit rename disk01 disk03
```

Note – For more information on the `vxedit` command, see the `vxedit(1M)` man page.

To see if the name change took place, enter:

```
vxdisk list
```

The Volume Manager returns:

```
DEVICE      TYPE      DISK          GROUP      STATUS
c0t0d0s2    sliced    c0t0d0s2      rootdg     online
c1t0d0s2    sliced    disk03        rootdg     online
c1t1d0s2    sliced    -             -          online
```

10.4 Resizing the Components of a Configuration

10.4.1 Resizing a Volume

10.4.1.1 Determining Largest Possible Size for Volumes

The `vxassist` command can provide information on the largest possible size for a volume that can currently be created with a given set of attributes. `vxassist` can also provide similar information for how much an existing volume can be extended under the current circumstances.

To determine the largest possible size for the volume to be created, use the command:

```
vxassist maxsize attributes...
```

For example, to find out the maximum size for new a RAID-5 volume on available disks, enter:

```
vxassist maxsize layout=raid5
```

This does not actually create the volume, but returns output such as:

```
Maximum volume size: 376832 (184Mb)
```

If, however, a volume with the specified attributes cannot be created, an error similar to the following might result:

```
vxvm:vxassist: ERROR: No volume can be created within the given constraints
```

To determine how much an existing volume can grow, use the command:

```
vxassist maxgrow volume_name
```

For example, the command:

```
vxassist maxgrow raidvol
```

might result in output similar to this:

```
Volume raidvol can be extended by 366592 to 1677312 (819Mb)
```

Notice that this output indicates both the amount *by* which the volume can be increased and the total size *to* which the volume can grow.

10.4.1.2 Extending a Volume Size

If the volume is not large enough for the amount of data that needs to be stored in it, you should extend the volume's length.

♦ **To extend a volume *to* a specific length, enter:**

```
vxassist growto volume_name length
```

For example, to extend `volcat` to 2000 512-byte sectors, enter:

```
vxassist growto volcat 2000
```

♦ **To extend a volume *by* a specific length, enter:**

```
vxassist growby volume_name length
```

For example, to extend `volcat` by 100 sectors, enter:

```
vxassist growby volcat 100
```

Note – For more information on the `vxassist` command, see the `vxassist(1M)` man page.

10.4.1.3 *Shrinking a Volume Size*

If you find that your volume is much larger than you really need it to be, you can shrink the volume's size.

Caution – Do not shrink a volume below the size of the file system on that volume.

♦ **To shrink a volume *to* a specific length, enter:**

```
vxassist shrinkto volume_name length
```

Make sure not to shrink the volume below the current size of the file system or database using the volume. This command can be safely used on empty volumes. For example, to shrink `volcat` to 1300 sectors, enter:

```
vxassist shrinkto volcat 1300
```

♦ **To shrink a volume *by* a specific length, enter:**

```
vxassist shrinkby volume_name length
```

For example, to shrink `volcat` by 300 sectors, enter:

```
vxassist shrinkby volcat 300s
```

Note – For more information on the `vxassist` command, see the `vxassist(1M)` man page.

10.4.1.4 Changing the Length of a Volume

To change the length of a volume using `vxvol set`, use the following command:

```
vxvol set len=value ... volume_name ...
```

For example, to change the length to 100000 sectors, use the following command:

```
vxvol set len=100000 vol01
```

Note – The `vxvol set len` command cannot increase the size of a volume unless the needed space is available in the plexes of the volume.

As with mirrored volumes, several attributes of RAID-5 volumes can be changed by the user. For RAID-5 volumes, the volume length and RAID-5 log length can be manipulated using the `vxvol set` command. To change the length of a RAID-5 volume, the following command can be used:

```
vxvol set len=10240 r5vol
```

The length of a volume can only be made as long as the mapped region (called the *contiguous length*, or *contiglen*) of the RAID-5 plex. In other words, the length can not be extended so as to make the volume unusable. If the RAID-5 volume is active and the length is being shortened, the operation must be forced using the `-o force` usage type option; this is done to prevent yanking space away applications using the volume.

The length of the RAID-5 logs can also be changed using `vxvol set`:

```
vxvol set loglen=2M r5vol
```

Remember that RAID-5 log plexes are only valid if they map the entire length of the RAID-5 volume's log length. If increasing the log length would make any of the RAID-5 logs invalid, the operation will not be allowed. Also, if the volume is not active and is dirty (i.e. was not shut down cleanly) the log length cannot be changed. This avoids the loss of any of the log contents (if the log length is decreased) or the introduction of random data into the logs (if the log length is being increased).

10.4.2 Splitting Subdisks

Splitting a subdisk divides an existing subdisk into two subdisks. The `–s` option is required to specify the size of the first of the two subdisks that will be created. To split a subdisk, enter:

```
vxsd –s size split sd newsd newsd2
```

where *sd* is the name of the original subdisk, *newsd* is the name of the first of the two subdisks that will be created, and *newsd2* is the name of the second subdisk to be created.

If the existing subdisk is associated with a plex before the operation, upon completion of the split, both of the resulting subdisks will be associated to the same plex.

10.4.3 Joining Subdisks

Joining a subdisk combines two or more existing subdisks into one subdisk. To join subdisks, the subdisks must be contiguous on the same disk; if the selected subdisks are associated, they must be associated with the same plex, and be contiguous in that plex. The command to join a subdisk is:

```
vxsd join subdisk1 subdisk2 new_subdisk
```

10.5 Stopping and Starting Components of a Configuration

10.5.1 Starting and Stopping Disk Drives

You can spin up or spin down the physical disk drives in your SPARCstorage Array at three levels:

- Spinning Up and Spinning Down All Drives in an Array — page 10-39
- Spinning Up and Spinning Down All Drives in a Disk Tray — page 10-40
- Spinning Up and Spinning Down Individual Disk Drives — page 10-40

10.5.2 Spinning Up and Spinning Down All Drives in an Array

To spin up all drives in a SPARCstorage Array:

♦ **Enter the following command at the prompt:**

```
ssaadm start controller
```

where `controller` is the logical name of the controller, given in the form `c#`.

Note – For more information on the `ssaadm` command, see the `ssaadm(1M)` man page.

To spin down all drives in a SPARCstorage Array:

♦ **Enter the following command at the prompt:**

```
ssaadm stop controller
```

10.5.3 Spinning Up and Spinning Down All Drives in a Disk Tray

To spin up all drives in a disk tray:

♦ **Enter the following command at the prompt:**

```
ssaadm start -t tray_number controller
```

where `tray number` is the number of the disk tray containing the drives and `controller` is the logical name of the controller, given in the form `c#`. For example, to start all drives in tray 3 on controller c2, you would enter:

```
ssaadm start -t 3 c2
```

To spin down all drives in a disk tray:

♦ **Enter the following command at the prompt:**

```
ssaadm stop -t tray_number controller
```

10.5.4 Spinning Up and Spinning Down Individual Disk Drives

To spin up individual disk drives:

♦ **Enter the following command at the prompt:**

```
ssaadm start drive
```

where `drive` is the logical disk address for the drive, given in the form `/dev/rdsk/c#t#d#s#`.

To spin down individual disk drives:

♦ **Enter the following command at the prompt:**

```
ssaadm stop drive
```

10.5.5 Starting and Stopping Volumes

Like mounting and unmounting a file system, starting and stopping a volume affects its availability to the user. Starting a volume changes its state and makes it available for use. Stopping a volume makes it unavailable.

Note – It is seldom desirable to stop a volume. Volume devices that are closed as part of the shutdown will not cause recovery operations to be performed the next time the volume is started.

Starting a volume changes the volume state from disabled or detached to enabled. The success of this operation depends on the ability to enable a volume. If a volume cannot be enabled, it remains in its current state. To start a volume, enter:

```
vxrecover -s volume_name ...
```

To start all disabled volumes, enter:

```
vxrecover -s
```

Stopping a volume changes the volume state from enabled or detached to disabled. If the command cannot stop it, the volume remains in its current state. To stop a volume, enter:

```
vxvol stop volume_name ...
```

For example, the command line to stop a volume labeled `vol01` looks like this:

```
vxvol stop vol01
```

To stop all enabled volumes, enter:

```
vxvol stopall
```

If all plexes of the volume become stale, put the volume in maintenance mode so that the plexes can be looked at while the volume is detached and determine which plex to use for reviving the others. To place a volume in maintenance mode, enter:

```
vxvol maint volume_name
```

To assist in choosing the revival source plex, list the unstarted volume and displays its plexes.

To take plex `vol01-02` offline, enter:

```
vxmend off vol01-02
```

For enabled volumes, save a step by using `vxplex att` without first invoking `vxmend on`. This command works on an offline plex of an enabled volume (designated as `vol01` in the example):

```
vxplex att vol01 vol01-02
```

The `vxmend` utility can change the state of an offline plex of a disabled volume to stale, after which a `vxvol start` on the volume would revive the plex. To put a plex labeled `vol01-02` in the stale state, enter:

```
vxmend on vol01-02
```

An unstartable volume is likely to be incorrectly configured or has other errors or conditions that prevent it from being started. To display unstartable volumes, use the command `vxinfo`, which displays information on the accessibility and usability of one or more volumes:

```
vxinfo volume_name
```

10.6 Changing Components of a Configuration

10.6.1 Changing Ownership and Permissions

If you want to change the ownership or permissions on a file or disk that is under Volume Manager control, you cannot use the `chgrp`, `chown` or `chmod` commands to do this. However, you can change the ownership or permissions on the corresponding volume using Volume Manager.

To change the ownership or permissions on a volume, enter:

```
vxedit set user=username group=groupname mode=modenumber volumename
```

For example, to change the owner of the group to `susan` and the permissions to read/write for owner, group, and world, you would enter:

```
vxedit set user=susan group=staff mode=0666 vol01
```

10.6.2 Changing the Information on an Object

10.6.2.1 Changing Volume Information

Volume attributes such as read policy, error policies, ownership, permissions, and the values in the comment and utility fields for existing volumes can be changed. These attributes are changed whenever the use of the volume or your needs change.

There are two Volume Manager commands associated with setting volume attributes:

- The `vxedit` command sets those attributes that are not usage-type-dependent.
- The `vxvol` command sets only those attributes that are usage-type-dependent.

Examples of how to use each of these commands follow:

```
vxvol set field=value0 ... volume_name ...
```

or

```
vxedit set field=value0 ... volume_name ...
```

Table 10-1 details which attributes can be set by each command.

Table 10-1 Setting Volume Attributes

Command	Attribute	Description
vxedit	comment	comment field
	tutil0, tutil1, tutil2 putil0, putil1,putil2	descriptive string of volume contents
	fstype	string indicating file system type
	writeback	boolean (on/off) specifying read error correction mode
	user	owner of volume
	group	group of volume
	mode	permission mode for volume
vxvol	len log type	numeric length of volume (drl/undef) specifier of dirty region logging mode for volume
	log len	length of the dirty region logging log
	start opts	options to be executed to the vxvol `start` operation

Volume Manager offers the choice of two read policies:

- `round` reads each plex in turn in “round-robin” fashion for each non-sequential I/O detected. Sequential access will cause only one plex to be accessed, thus taking advantage of the drive or controller read-ahead caching policies.
- `prefer` reads preferentially from a plex that has been labeled as the preferred plex.

The read policy can be changed from `round` to `prefer` (or vice versa in the case of prefer) or to a different preferred plex.

The `vxvol rdpol` command sets the read policy for a volume. To set a read policy, use one of the following commands:

```
vxvol rdpol round volume_name
```

or

```
vxvol rdpol prefer volume_name preferred_plex_name
```

For example, the command line to set the read policy for volume `vol01` to a round-robin read looks like this:

```
vxvol rdpol round vol01
```

The command line to set the policy for the same volume to read preferentially from the plex looks like this:

```
vxvol rdpol prefer vol01 vol01-02
```

10.6.2.2 Changing Plex Information

The `comment` field and the `putil` and `tutil` fields are used by the utilities after plex creation. `putil` attributes are maintained on reboot; `tutil` fields are temporary and are not retained on reboot. Both `putil` and `tutil` have three uses and are numbered according to those uses. These fields can be modified as needed. Volume Manager uses the utility fields marked `putil0` and `tutil0`. Other products use those marked `putil1` and `tutil1`; those marked `putil2` and `tutil2` are user fields. Table 10-2 details the uses for the `putil` and `tutil` fields.

Table 10-2 The putil[n] and tutil[n] Fields

Field	Description
`putil0`	This utility field is reserved for use by Volume Manager utilities and is retained on reboot.
`putil1`	This utility field is reserved for use by high-level utilities such as the GUI and the Administrative Script interface. This field is retained on reboot.
`putil2`	This utility field is reserved for use by the system administrator or site-specific applications. This field is retained on reboot.
`tutil0`	This utility field is reserved for use by Volume Manager utilities and is cleared on reboot.
`tutil1`	This utility field is reserved for use by high-level utilities such as the GUI and the Administrative Script interface. This field is cleared on reboot.
`tutil2`	This utility field is reserved for use by the system administrator or site-specific applications. This field is cleared on reboot.

To change plex information, enter:

```
vxedit set field=value ... plex_name ...
```

The command:

```
vxedit set comment="my plex" tutil2="u" uid="admin" vol01-02
```

uses `vxedit` to set the following information:

- set the comment field (identifying what the plex is used for) to `my_plex`
- set `tutil2` to `u` to indicate that the subdisk is in use
- change the user ID to `admin`

To prevent a particular plex from being associated with a volume, set the putil0 field to a non-null string as specified in the following command:

```
vxedit set putil0="DO-NOT-USE" vol01-02
```

10.6.2.3 *Changing Subdisk Information*

The `vxedit` utility changes information related to subdisks. To change information relating to a subdisk, enter:

```
vxedit set field=value ... subdisk_name
```

For example, the command to change the comment field of subdisk labeled `disk02-01` would be:

```
vxedit set comment= "New comment" disk02-01
```

The subdisk fields that can be changed using `vxedit` are:

- name
- the `putil[n]` fields
- the `tutil[n]` fields
- `len` (only if the subdisk is dissociated)
- comment

Note – Entering data in the `putil` and `tutil` fields prevents the subdisk from being used as part of a plex, if it is not already.

10.6.3 Changing the Plex Status

Once a volume has been created and placed online (enabled), Volume Manager provides mechanisms by which plexes can be temporarily disconnected from the volume. This is useful, for example, when the hardware on which the plex resides needs repair or when a volume has been left unstartable and a source plex for the volume revived must be chosen manually.

Resolving a disk or system failure involves:

- taking the volume offline
- detaching its plexes
- making any necessary hardware repairs
- attaching its plexes

To take a plex offline so that repair or maintenance can be performed on the physical disk containing that plex's subdisks, enter:

```
vxmend off plex_name ...
```

If a disk drive has failed terminally, you should take all plexes offline that have associated subdisks represented on the affected drive. For example, if plexes `vol01-02` and `vol02pl` had subdisks on a drive to be repaired, enter:

```
vxmend off vol01-02 vol02pl
```

This command places vol01-02 and vol02pl in the offline state, and they remain in that state until explicitly changed.

To detach the plex, refer to “Detaching a Plex” on page 10-29. To attach the plex, refer to “Attaching a Plex” on page 10-17.

Maintaining the Data 11

This chapter gives the procedures for maintaining the data for your system:

11.1 Repairing Physical Disks

This section discusses how to detect physical disk failures and how to replace failed disks.

11.1.1 Detecting Failed Physical Disks

If one mirror of a volume encounters a disk I/O failure (for example, because the disk has an uncorrectable format error), the Volume Manager may detach the mirror. If a disk fails completely, the Volume Manager may detach the disk from its disk group. If a mirror is detached, I/O stops on that mirror but continues on the remaining mirrors of the volume. If a disk is detached, all mirrors on the disk are disabled. If there are any unmirrored volumes on a disk when it is detached, those volumes are disabled as well.

If a volume, mirror, or disk is detached by failures, mail is sent to `root` indicating the failed objects. For example, if a disk containing two mirrored volumes fails, you might receive a mail message like:

```
To: root
Subject: Volume Manager failures on host minuet.tango.com

Failures have been detected by the Volume Manager on host
minuet.tango.com:

failed plexes:
  home-02
  src-02

No data appears to have been lost. However, you should replace
the drives that have failed.
```

See the `vxsparecheck`(1M) manual page for information on sending the mail to users other than `root`.

Note – If you miss the e-mail, the failure can be seen with the `vxprint` command output or by using the GUI to look at the status of the disks.

To determine which disks are causing the failures in the above message, enter:

```
vxstat -sf home-02 src-02
```

This will produce output such as:

```
                                  FAILED
TYP NAME                      READS    WRITES
sd disk01-04                      0         0
sd disk01-06                      0         0
sd disk02-03                      1         0
sd disk02-04                      1         0
```

This display indicates that the failures are on `disk02` (the basename for the displayed subdisks).

If there are any problems which do not require replacing a disk drive (for example, a loose cable), correct them and recover the mirrors with the command:

```
vxrecover -b home src
```

This command will start a recovery of the failed mirrors in the background (the command will return before the operation is done). If an error message appears later or if the mirrors become detached again, there is probably a hardware failure, so you may need to replace a component.

If a disk fails completely, the mail message will list the disks that have failed, all mirrors that use the disk, and all volumes defined on the disk that were disabled because the volumes were not mirrored. For example:

```
To: root
Subject: Volume Manager failures on host minuet.tango.com

Failures have been detected by the Volume Manager on host
minuet.tango.com:

failed disks:
  disk02

failed plexes:
  home-02
  src-02
  mkting-01

failed volumes:
  mkting

The contents of failed volumes may be corrupted, and should be
restored from any available backups. To restart one of these
volumes so that you can restore it from backup, replace disks as
appropriate then use the command:

     vxvol -f start <volume-name>

You can then restore or recreate the volume.
```

This message indicates that `disk02` was detached by a failure. When a disk is detached, I/O cannot get to that disk. Mirrors `home-02`, `src-02`, and `mkting-01` were also detached (probably because of the failure of the disk); and the volume `mkting` was disabled.

11.1.2 Replacing Physical Disks

Physical disks that have failed completely (that have been detached by failure) can be replaced by running `vxdiskadm` and selecting item 5, `Replace a failed or removed disk` from the main menu. If there are any initialized but unadded disks, you will be able to select one of those disks as a replacement. Do not choose the old disk drive as a replacement; it may appear in the selection list. If there are no suitable initialized disks, you can choose to initialize a new disk.

If a disk failure caused a volume to be disabled then the volume must be restored from backup after the disk is replaced. To identify volumes that wholly reside on disks that were disabled by a disk failure, use the command:

```
vxinfo
```

Any volumes that are listed as `Unstartable` must be restored from backup. For example, `vxinfo` might display:

```
home          fsgen     Started
mkting        fsgen     Unstartable
src           fsgen     Started
standvol      gen       Started
rootvol       root      Started
swapvol       swap      Started
```

To restart volume `mkting` so that it can be restored from backup, use the command:

```
vxvol -obg -f start mkting
```

The `-obg` option causes any mirrors to be recovered in a background task.

If failures are starting to occur on a disk, but the disk has not yet failed completely, you should replace the disk. This involves two steps: detaching the disk from its disk group and replacing the disk with a new one. To detach the disk, run `vxdiskadm` and select item 4, `Remove a disk for replacement`, from the main menu. If there are initialized disks available as replacements, you can specify the disk as part of this operation. Otherwise, you must specify the replacement disk later by selecting item 5, `Replace a failed or removed disk`, from the main menu.

When you select a disk to remove for replacement, all volumes that will be affected by the operation are displayed. For example, the following output might be displayed:

```
The following volumes will lose mirrors as a result of this
operation:

     lhome src

No data on these volumes will be lost.

The following volumes are in use, and will be disabled as a result
of this operation:

     mkting

Any applications using these volumes will fail future accesses.
These volumes will require restoration from backup.

Are you sure you want do this? [y,n,q,?] (default: n)
```

If any volumes would be disabled, you should quit from `vxdiskadm` and save the volume. Either backup the volume or move the volume off of the disk. To move the volume `mkting` to a disk other than `disk02`, use the command:

```
vxassist move mkting !disk02
```

After the volume is backed up or moved, run `vxdiskadm` again and continue to remove the disk for replacement.

After the disk has been removed for replacement, you can use a replacement disk by specifying item 5, `Replace a failed or removed disk`, from the main menu in `vxdiskadm`.

11.1.3 Hot-Sparing

Hot-sparing is used to automatically recover data when a physical disk fails. When a disk fails, Volume Manager looks for the closest spare disk and places the data from the failed disk on the spare disk. The replacement disk device must be initialized and placed on the system as a spare.

When a disk failure is detected, Volume Manager first attempts to correct the error that brought the failure to its attention. If Volume Manager is unable to correct the error, `vxconfigd` is notified. `vxconfigd` then tries to access the disk. If `vxconfigd` cannot access the disk, it considers the disk to have failed and sends root a mail message about the failure.

At the same time, `vxconfigd` notifies the `vxsparecheck`. `vxsparecheck` searches for a suitable replacement for the failed disk. The replacement selection is based upon what disks are available and what Volume Manager objects reside on the failed disk (for example, if the disk that failed was part of a striped volume, the replacement disk should be on a different controller than the other disks in that volume).

Note – If no spare disks large enough to contain all the data from the failed disk are available, Volume Manager notifies you about the failure, but no other automatic action is taken.

The disk access record of the failed disk is disassociated from its disk media record. The disk media record is then associated with the disk access record of the replacement disk device. This leaves the disk media record for the new device unassociated and the device's error flag is set. The failed device is also unassociated. The display from a `vxdisk list` will show this disk to have last been attached to the original disk media record.

For example, if the failed disk is called `disk01` and the hot spare is called `disk30`, then after the hot sparing operation is completed, the physical drive formerly known as `disk30` will be called `disk01`.

For information on designating a VM disk as a spare, see "Adding a VM Disk as a Hot Spare" on page 10-11.

11.2 Recovering Volumes

If a system crash or an I/O error corrupts one or more plexes of a volume and no plex is CLEAN or ACTIVE, mark one of the plexes CLEAN and instruct the system to use that plex as the source for reviving the others. To place a plex in a CLEAN state, enter:

```
vxmend fix clean plex_name
```

For example, the command line to place one plex labeled in the CLEAN state looks like this:

```
vxmend fix clean vol01-02
```

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

11.3 Initializing a Volume

During normal system operation, volume and plex states will be affected by system failures, shutdowns, and possible I/O failures. When a volume is first created, it is necessary to initialize the state of its one or more plexes according to what the state of the data is on each plex. Normally, if the user has created the volume using `vxassist` or one of the other higher-level interfaces, the state of the plexes will be properly set. However, when `vxmake` has been used to create a volume, the states of its plexes must be set manually before the volume can be made available for use through the `vxvol start` command. The command for setting the state of a volume's plexes is:

```
vxvol init state volume_name [plex_name]
```

where the *state* variable determines what the initialization does and what condition the volume and plexes will have after the volume has been initialized.

The most common form of manual initialization is setting the state of the volume to CLEAN. The following examples show how to do this for mirrored and non-mirrored volumes. In the simplest case, in which a volume has been created containing only one plex (mirror), the state of the plex is set to CLEAN. This is because there is no need for any synchronization of the data on the disk. Since there is only one plex in the volume, it is not necessary to specify the *plex_name* argument. The command to set the state of this volume to CLEAN is:

```
vxvol init clean volume_name
```

Note – The rest of this section applies only to mirrored volumes (not RAID-5 volumes, which are discussed later).

Under more complicated circumstances, where a newly created volume `vol01` has multiple mirrors associated with it, then one of the plexes must be chosen to which the other plexes are synchronized. For instance, if plex `vol01-02` has been created over disk space that contained data that needed to be accessed through the volume after it is made available, then the following command would ensure that the data is synchronized out to the other plexes when the volume is started:

```
vxvol init clean vol01 vol01-02
```

This command will set the state of `vol01-02` to CLEAN and the remainder of the plexes to STALE, so that they will be properly synchronized at the time the volume is made available. Sometimes, the administrator will wish to avoid the initial synchronization of the volume in order to save time, with the predefined knowledge that none of the plexes contain data that will be the final contents of the volume. Under such a situation, it is possible to temporarily initialize the state of the volume so that the data can be loaded without having to perform a synchronization first. The command to do this is:

```
vxvol init enable volume_name
```

This enables the volume and all its plexes, but leaves the plex utility states set to EMPTY. After the entire volume's contents have been restored, both mirrors contain exactly the same data and will not need to be synchronized using the `vxvol start` operation. Such a volume, for example `home1`, could be initialized for use and started at the same time using the following command:

```
vxvol init active home1
```

Caution – It is strongly recommended that data on each of the mirrors be exactly the same under these circumstances. Otherwise, the system may corrupt the data on both mirrors and, depending on the use for the volume, could crash the system. If you are not sure that the data is identical, then use the `vxvol init clean` method.

Sometimes it is necessary to remove all existing data from disks before new data is loaded. In this case, you can initialize every byte of the volume to zero by entering:

```
vxvol init zero volume_name
```

11.4 Initializing a RAID-5 Volume

A RAID-5 volume needs to be initialized if it was created by `vxmake` and has not yet been initialized or if it has somehow been set to an uninitialized state.

A RAID-5 volume can be initialized through `vxvol` in either of the following ways:

```
vxvol init zero volume_name
```

or

```
vxvol start volume_name
```

`vxvol init zero` writes zeroes to any RAID-5 logging plexes and to the entire length of the volume; it then leaves the volume in the ACTIVE state.

`vxvol start` recovers parity by XORing together corresponding data stripe units in all other columns. Although it is slower than a `vxvol init zero` operation, `vxvol start` makes the RAID-5 volume immediately available.

11.5 Displaying Properties of an Object

You can display the properties of the following objects:

- SPARCstorage Array
- Disk groups
- Physical disks
- Volumes
- File systems
- Plexes
- Subdisks

11.5.1 Displaying Properties of a SPARCstorage Array

♦ **Enter the following command at the prompt:**

```
ssaadm display controller
```

where `controller` is the logical name of the controller, given in the form `c#`.

You will see output similar to the following:

```
                    SPARCstorage Array Configuration
Controller path:/devices/io-
unit@f,e3200000/sbi@0,0/SUNW,soc@2,0/SUNW,pln@a0000000,1ab9:ctlr
                          DEVICE STATUS
      TRAY 1                 TRAY 2                 TRAY 3
slot
1     Drive: 0,0             Drive: 2,0             Drive: 4,0
2     Drive: 0,1             Drive: 2,1             Drive: 4,1
3     NO SELECT              NO SELECT              NO SELECT
4     NO SELECT              NO SELECT              NO SELECT
5     NO SELECT              NO SELECT              NO SELECT
6     Drive: 1,0             Drive: 3,0             Drive: 5,0
7     Drive: 1,1             Drive: 3,1             NO SELECT
8     NO SELECT              NO SELECT              NO SELECT
9     NO SELECT              NO SELECT              NO SELECT
10    NO SELECT              NO SELECT              NO SELECT

                          CONTROLLER STATUS
Vendor:         SUN
Product ID:     SSA100
Product Rev:    1.0
Firmware Rev:   1.9
Serial Num:     000000001AB9
Accumulate Performance Statistics: Enabled
```

Note – For more information on the ssaadm command, see the ssaadm(1M) man page.

11.5.2 Displaying Properties of Disk Groups

To use disk groups, you need to know what they are and what disks belong to each group.

♦ **To display information on existing disk groups, enter:**

```
vxdg list
```

The Volume Manager returns the following:

```
NAME      STATE     ID
rootdg    enabled   730344554.1025.harley
newdg     enabled   731118794.1213.harley
```

Note – For more information on the `vxdg` command, see the `vxdg(1M)` man page.

11.5.3 Displaying Properties of Physical Disks

You can get the properties of a physical disk from the SPARCstorage Array or the Volume Manager perspective.

11.5.3.1 Displaying SPARCstorage Array Disk Properties

♦ **To display the properties of a SPARCstorage Array disk, enter:**

```
ssaadm display drive
```

where `drive` is the logical disk address for the drive, given in the form `/dev/rdsk/c#t#d#s#`.

You will see output similar to the following:

```
DEVICE PROPERTIES for device /dev/rdsk/c1t5d0s0
 SCSI Port 5  Target 0
 Status:        O.K.
 Vendor:        CONNER
 Product ID:    CP30548  SUN0535
 Product Rev:   AEBX
 Firmware Rev:  9308
 Serial Num:    28QB
 Unformatted Capacity: 532956 KByte
 Fast Writes:   Disabled
```

Following are the different messages you could see on the status line:

- No Drive Found
- Drive did not respond to Select
- Drive not Ready
- Could not Read from Drive
- Drive Spun Down
- No UNIX Label
- O.K.

Note – For more information on the `ssaadm` command, see the `ssaadm(1M)` man page.

11.5.3.2 Displaying VM Disk Properties

Before you use a disk, you need to know if it has been initialized. You need to know if the disk is part of a disk group, since you cannot create volumes on a disk that is not part of a disk group. The `list` command displays device names for all recognized disks, the disk names, the disk group names associated with each disk, and the status of each disk.

♦ **To display information on all disks that are defined to the Volume Manager, enter:**

```
vxdisk list
```

The Volume Manager returns the following display:

```
DEVICE       TYPE      DISK         GROUP      STATUS
c0t0d0s2     sliced    c0t0d0s2     rootdg     online
c1t0d0s2     sliced    disk01       rootdg     online
c1t1d0s2     sliced    -            -          online
```

♦ **To display details on a particular disk defined to the Volume Manager, enter:**

```
vxdisk list disk01
```

Note – For more information on the `vxdisk` command, see the `vxdisk(1M)` man page.

11.5.4 Displaying Properties of Volumes

It is possible to list information related to volumes under Volume Manager control. This information includes the name of the volume, its usage type, state, length, user and group IDs, and mode. To list information on all volumes, use the following command:

```
vxprint -vt
```

To display detailed information about a specific volume, use the following command:

```
vxprint -l volume_name
```

If no volume is specified, detailed information is given for all volumes by using the following command:

```
vxprint -vl
```

11.5.5 Displaying Properties of File Systems

It is important to keep track of which file systems are mounted and which are not. This saves users from trying to access unmounted file systems. You can look at the status of file systems by entering:

```
mount
```

The `mount` command displays the file system information similar to that shown below:

```
/ on /dev/root read/write/setuid on Sun Feb 21 17:45:22 1993
/proc on /proc read/write on Sun Feb 21 17:45:24 1993
/dev/fd on /dev/fd read/write on Sun Feb 21 17:45:24 1993
/stand on /dev/vol/standvol read/write on Sun Feb 21 17:45:25
1993
/lhome on /dev/vol/homevol
read/write/delaylog/setuid/mincache=closesync \ on Sun Feb 21
17:47:12 1993
```

11.5.6 Displaying Properties of Plexes

To display detailed information about all plexes, enter:

```
vxprint -lp
```

To display detailed information about a specific plex, enter:

```
vxprint -l plex_name
```

11.5.7 Displaying Properties of Subdisks

To display general information for all subdisks, enter:

```
vxprint -st
```

The -s option gives information about subdisks. The -t option prints a single-line output record that depends on the type of object being listed.

To display complete information about a particular subdisk, enter:

```
vxprint -l subdisk_name
```

For example, the command line to obtain all database information on a subdisk labeled disk02-01 looks like this:

```
vxprint -l disk02-01
```

11.6 Displaying Activity Levels on Objects

You can display the activity levels on SPARCstorage Array disks or on Volume Manager objects.

11.6.1 Displaying Activity Levels on SPARCstorage Array Disks

♦ **To display the activity level on SPARCstorage Array disks, enter:**

```
ssaadm -p display controller
```

where controller is the logical name of the controller, given in the form c#.

You will see output similar to the following:

```
PERFORMANCE Log
BUSY: 84% IOPS: 1102
                              DEVICE IOPS
slot      TRAY 1        TRAY 2        TRAY 3
1         254           20            103
2         120           348           348
3         41            20            123
4         34            20            10
5         26            20            13
6         254           20            43
7         72            348           0
8         50            20            0
9         31            0             0
10        7             11            46
```

Note – For more information on the `ssaadm` command, see the `ssaadm(1M)` man page.

11.6.2 Displaying Activity Levels on Volume Manager Objects

There are two sets of priorities for a system administrator. One set is physical, concerned with the hardware; the other set is logical, concerned with managing the software and its operations.

The physical performance characteristics address the balance of the I/O on each drive and the concentration of the I/O within a drive to minimize seek time. Based on monitored results, it may be necessary to move subdisks around to balance the disks.

The logical priorities involve software operations and how they are managed. Based on monitoring, certain volumes may be mirrored or striped to improve their performance. Overall throughput may be sacrificed to improve the performance of critical volumes. The system administrator can decide what is important on a system and what tradeoffs make sense.

11.6.2.1 Getting Performance Data

Volume Manager provides two types of performance information: I/O statistics and I/O traces. Each type can help in performance monitoring. I/O statistics are retrieved using the `vxstat` utility, and I/O tracing can be retrieved using the `vxtrace` utility.

Obtaining I/O Statistics

The `vxstat` utility provides access to information for activity on volumes, mirrors, subdisks, and disks under Volume Manager control. `vxstat` reports statistics that reflect the activity levels of Volume Manager objects since boot time. Statistics for a specific Volume Manager object or all objects can be displayed at one time. A disk group can also be specified, in which case statistics for objects in that disk group only will be displayed.

The amount of information displayed depends on what options are specified to `vxstat`. For detailed information on available options, refer to the `vxstat(1M)` manual page.

Volume Manager records the following three I/O statistics:

- a count of operations
- the number of blocks transferred (one operation could involve more than one block)
- the total active time

Volume Manager records the preceding three pieces of information for logical I/Os: reads, writes, atomic copies, verified reads, verified writes, mirror reads, and mirror writes for each volume. As a result, one write to a two-mirror volume results in at least five operations being recorded: one for each mirror, one for each subdisk, and one for the volume. Similarly, one read that spans two subdisks shows at least four reads: one read for each subdisk, one for the mirror, and one for the volume.

Volume Manager also maintains other statistical data. For each mirror, read failures and write failures that appear are maintained. For volumes, corrected read failures and write failures accompany the read failures and write failures.

`vxstat` is also capable of resetting the statistics information to zero. Use the command `vxstat -r` to clear all statistics. This can be done for all objects or for only those objects that are specified. Resetting just prior to a particular operation makes it possible to measure the impact of that particular operation afterwards.

The following is an example of `vxstat` output:

```
                    OPERATIONS           BLOCKS            AVG TIME(ms)
TYP NAME          READ    WRITE      READ       WRITE     READ      WRITE
vol blop             0        0         0           0      0.0        0.0
vol foobarvol        0        0         0           0      0.0        0.0
vol rootvol      73017   181735    718528     1114227     26.8       27.9
vol swapvol      13197    20252    105569      162009     25.8      397.0
vol testvol          0        0         0           0      0.0        0.0
```

Tracing I/O

The `vxtrace` command is used to trace operations on volumes. Through the `vxtrace` utility, the system administrator can set I/O tracing masks against a group of volumes or to the system as a whole. The `vxtrace` utility can then be used to display ongoing I/O operations relative to the masks. Tracing can be applied to volumes, plexes, subdisks, and physical disks. Each separate user of the I/O tracing can specify how the desired trace mask is set, independent of all other users. For additional information, refer to the `vxtrace(1M)` man page.

11.6.2.2 Using Performance Data

Once performance data has been gathered, it can be used to determine an optimum system configuration in order to make the most efficient use of system resources. The following sections provide an overview of how this data can be used.

Using I/O Statistics

By examining the I/O statistics, you may realize that you should reconfigure your system. There are two primary statistics to look at: volume I/O activity and disk I/O activity.

Before obtaining statistics, consider clearing (resetting) all existing statistics.

♦ **To clear all statistics, enter:**

```
vxstat -r
```

Clearing statistics eliminates any differences between volumes or disks that might appear due to volumes being created, and also removes statistics from booting (which are not normally of interest).

After clearing the statistics, let the system run for a while and then display the accumulated statistics. Try to let it run during typical system activity. In order to measure the effect of a particular application or workload, it should be run specifically. When monitoring a system that is used for multiple purposes, try not to exercise any one application more than it would be exercised normally. When monitoring a time-sharing system with many users, try to let statistics accumulate during normal use for several hours during the day.

♦ **To display volume statistics, use the command `vxstat` with no arguments.**

This might display a list such as:

```
                    OPERATIONS          BLOCKS          AVG TIME(ms)
TYP NAME          READ   WRITE       READ     WRITE      READ   WRITE
vol archive        865     807       5722      3809      32.5    24.0
vol home          2980    5287       6504     10550      37.7   221.1
vol local        49477   49230     507892    204975      28.5    33.5
vol rootvol 102906 342664    1085520   1962946      28.1    25.6
vol src          79174   23603     425472    139302      22.4    30.9
vol swapvol      22751   32364     182001    258905      25.3   323.2
```

♦ **To display disk statistics, use the command `vxstat -d`.**
This might display a list such as:

```
                   OPERATIONS          BLOCKS          AVG TIME(ms)
TYP NAME         READ    WRITE      READ    WRITE     READ    WRITE
dm disk01       40473   174045    455898   951379     29.5     35.4
dm disk02       32668    16873    470337   351351     35.2    102.9
dm disk03       55249    60043    780779   731979     35.3     61.2
dm disk04       11909    13745    114508   128605     25.0    30.7
```

Look for volumes with an unusually large number of operations or excessive read or write times. It is normal for `rootvol` to have a large amount of activity relative to other volumes. However, `rootvol` and `swapvol` must be stored on the same disk and must each have single-subdisk plexes.

Note – Do not move `rootvol` or `swapvol` from the boot disks in an attempt to improve performance. Doing so may leave the system unbootable.

Try to avoid using the boot disks for volumes other than `rootvol` and `swapvol`. If it is necessary to store more volumes on the boot disks, look for volumes that are relatively unused (by looking at the I/O counts). For example, based on the previous I/O statistics, it is reasonable to move the volume archive onto the boot disks.

♦ **To move the volume archive onto the boot disk (disk01 here), identify which disk(s) it is on using `vxprint -tvh` archive.**
This might yield the output:

```
V   NAME        USETYPE     KSTATE   STATE    LENGTH  READPOL    PREFPLEX
PL  NAME        VOLUME      KSTATE   STATE    LENGTH  LAYOUT     NCOL/WDTH MODE
SD  NAME        PLEX        PLOFFS   DISKOFFS LENGTH  [COL/]OFF  FLAGS
v   archive     fsgen       ENABLED  ACTIVE   204800  SELECT     -
pl  archive-01  archive     ENABLED  ACTIVE   204800  CONCAT     -         RW
sd  disk03-03   archive-01  0        409600   204800  0          c1t2d0s0
```

Looking at the associated subdisks indicates that the archive volume is on disk disk03.

♦ **To move the volume off disk03, enter:**

```
vxassist move archive !disk03 dest_disk
```

where *dest_disk* is the disk you want to move the volume to. It is not necessary to specify a *dest_disk*. If you do not, the volume will be moved to any available disk with enough room to contain the volume.

For C-shell users, use:

```
vxassist move archive \!disk03 disk01
```

This command indicates that the volume should be reorganized such that no part is on disk03, and that any parts to be moved should be moved to disk01.

If there is one volume that is particularly busy (especially if it has unusually large average read or write times), consider striping the volume (or splitting the volume into multiple pieces, with each piece on a different disk). Converting a volume to use striping requires sufficient free space to store an extra copy of the volume. To convert to striping, create a striped mirror of the volume and then remove the old mirror. For example, to stripe the volume archive across disks disk02 and disk04, use:

```
vxassist mirror archive layout=stripe disk02 disk04
vxplex -o rm dis archive-01
```

After reorganizing any particularly busy volumes, check the disk statistics. If some volumes have been reorganized, clear statistics first and then accumulate statistics for a reasonable period of time.

If some disks appear to be excessively used (or have particularly long read or write times), it may be wise to reconfigure some volumes. If there are two relatively busy volumes on a disk, consider moving them closer together to reduce seek times on the disk. If there are too many relatively busy volumes on one disk, try to move them to a disk that is less busy.

Use I/O tracing (or perhaps subdisk statistics) to determine whether volumes have excessive activity in particular regions of the volume. If such regions can be identified, try to split the subdisks in the volume and to move those regions to a less busy disk.

Note – File systems and databases typically shift their use of allocated space over time, so this position-specific information on a volume is often not useful. For databases, it may be possible to identify the space used by a particularly busy index or table. If these can be identified, they are reasonable candidates for moving to non-busy disks.

Caution – Striping a volume, or splitting a volume across multiple disks, increases the chance that a disk failure will result in failure of that volume. For example, if five volumes are striped across the same five disks, then failure of any one of the five disks will require that all five volumes be restored from a backup. Use mirroring to substantially reduce the chances that a single disk failure will result in failure of a large number of volumes.

Examining the ratio of reads and writes helps to identify volumes that can be mirrored to improve their performance. If the read-to-write ratio is high, mirroring could increase performance as well as reliability. If a particularly busy volume has a ratio of reads to writes is as high as 5:1, it is likely that mirroring can dramatically improve performance of that volume.

Using I/O Tracing

I/O statistics provide the data for basic performance analysis; I/O traces serve for more detailed analysis. With an I/O trace, focus is narrowed to obtain an event trace for a specific workload. Exactly where a hot spot is, how big it is, and which application is causing it can be identified.

Using data from I/O traces, real work loads on disks can be simulated and the results traced. By using these statistics, the system administrator can anticipate system limitations and plan for additional resources.

11.7 Determining Amount of Free Space on Disk Groups

Before you add volumes and file systems to your system, make sure you have enough free disk space to adequately meet your needs. The Volume Manager lets you request a display of free space.

♦ **To display free space for a group, enter:**

```
vxdg -g disk_group free
```

For example, to see the free space in the default disk group, `rootdg`, enter:

```
vxdg free
```

The Volume Manager returns:

```
GROUP     DISK      DEVICE    TAG     OFFSET   LENGTH  FLAGS
rootdg    c0t0d0s2  c0t0d0s2  c0t0d0  726400   102672  -
rootdg    disk01    c1t0d0s2  c1t0d0  0        102128  -
rootdg    disk01    c1t0d0s2  c1t0d0  175856   26384   -
rootdg    disk02    c1t1d0s2  c1t1d0  26624    175616  -
```

The free space is measured in 512-byte sectors.

Note – For more information on the `vxdg` command, see the `vxdg(1M)` man page.

11.8 Determining Largest Possible Size for Volumes

The `vxassist` command can provide information on the largest possible size for a volume that can currently be created with a given set of attributes. `vxassist` can also provide similar information for how much an existing volume can be extended under the current circumstances.

To determine the largest possible size for the volume to be created, use the command:

```
vxassist maxsize attributes...
```

For example, to find out the maximum size for new a RAID-5 volume on available disks, enter:

```
vxassist maxsize layout=raid5
```

This does not actually create the volume, but returns output such as:

```
Maximum volume size: 376832 (184Mb)
```

If, however, a volume with the specified attributes cannot be created, an error similar to the following might result:

```
vxvm:vxassist: ERROR: No volume can be created within the given constraints
```

To determine how much an existing volume can grow, use the command:

```
vxassist maxgrow volume_name
```

For example, the command:

```
vxassist maxgrow raidvol
```

might result in output similar to this:

```
Volume raidvol can be extended by 366592 to 1677312 (819Mb)
```

Notice that this output indicates both the amount *by* which the volume can be increased and the total size *to* which the volume can grow.

11.9 Backing Up Your Data

It is very important to make backup copies of your volumes. Backup copies are used to restore volumes lost due to disk failure, or data destroyed due to human error. The Volume Manager allows you to back up volumes with minimal interruption of the volume's availability for users. Backing up a volume involves the following procedure:

1. Create a snapshot mirror of the volume to be backed up.
2. Select a suitable time to create a snapshot. You need to ensure that the data in the volume is in a consistent state and that no users are accessing the volume at the time the snapshot is taken.
3. Create a snapshot volume that reflects the original volume at the time of the snapshot.

The snapshot volume is a read-only volume that can now be used by back up utilities while the original volume continues to be available for applications and users.

1. To create a snapshot mirror for a volume, enter:

```
vxassist snapstart volume_name
```

For example, to create a snapshot mirror of a volume called `voldef`, enter:

```
vxassist snapstart voldef
```

2. After creating the snapshot mirror, unmount any file system or disable any database activity for that volume.

3. Create a snapshot volume that reflects the original volume:

```
vxassist snapshot volume_name new_volume_name
```

For example, to create a snapshot volume of `voldef`, enter:

```
vxassist snapshot voldef snapvol
```

You can now back up the snapshot volume by whatever means you prefer. To avoid wasting space, you can remove the snapshot volume, which occupies as much space as the original volume.

Note – For more information on the `vxassist` command, see the `vxassist(1M)` man page.

11.10 *Logging Recent Disk Activity*

You can add a log to a mirrored or a RAID-5 configuration.

11.10.1 *Adding a Log to a Mirror Configuration*

Logging subdisks are two, four, six, eight, or ten-block long subdisks that are defined for and added to a mirror that is to become part of a volume using dirty region logging. Dirty region logging is enabled for a volume when the volume has at least two active plexes that include a logging subdisk. For a description of dirty region logging, refer to “Dirty Region Logging” on page 1-31. Logging subdisks are ignored as far as the usual plex policies are concerned, and are only used to hold the *dirty region log*.

Note – Only one logging subdisk can be associated with a mirror. Because this subdisk is frequently written to, you should position it on a disk that is not heavily used. Placing a logging subdisk on a heavily-used disk can result in degradation of system performance.

♦ **To add a logging subdisk to a mirror, enter:**

```
vxsd aslog plex subdisk
```

For example, the command line to associate a subdisk labeled disk02-01 with a mirror labeled vol01-02 (which is already associated with volume vol01) would be:

```
vxsd aslog vol01-02 disk02-01
```

11.10.2 Adding a Log to a RAID-5 Configuration

Only one RAID-5 plex can exist per RAID-5 volume. Any additional plexes become RAID-5 log plexes, which are used to log information about data and parity being written to the volume. When a RAID-5 volume is created using `vxassist`, a log plex is created for that volume, by default.

♦ **To add a log to a RAID-5 configuration, enter:**

```
vxassist addlog volume_name
```

For example, to create a log for the RAID-5 volume `volraid`:

```
vxassist addlog volraid
```

11.11 Debugging the System

There are some environment variables that you can use to enable debug statements:

- S_DEBUG — Prints information when running Mode Sense command
- P_DEBUG — Prints information for each external library function call
- I_DEBUG — Prints progress during get status function
- O_DEBUG — Prints when files are opened

You can enable them by setting them in the same environment as the CLI or library program before starting the program. For example:

```
setenv S_DEBUG
```

Performing Service-Related Software Tasks 12

This chapter gives instructions on performing software tasks necessary for certain service procedures:

12.1 Flushing or Purging Outstanding Writes from NVRAM

Before you replace a Field Replaceable Unit (FRU), you must first determine if your system is set up for fast writes.

- If your system is set up for fast writes, go to “Flushing or Purging Outstanding Writes from NVRAM” on page 12-1.
- If your system is *not* set up for fast writes, go directly to the appropriate section to replace the FRU.

Note – If you are not sure if your system is set up for fast writes or not, you should go through the procedures given in “Flushing or Purging Outstanding Writes from NVRAM” anyway, since it won’t affect drives that aren’t set up for fast writes.

You must first determine if you want to flush or purge the outstanding writes from the NVRAM:

- The *flush* option flushes any outstanding writes from the NVRAM to the disk drive. Follow the instructions given in "Flushing Outstanding Writes from the NVRAM" on page 12-2 to do this.
- If you get an error message when you tried to flush the writes from the NVRAM, then you must *purge* the writes from the NVRAM. The purge option gives the controller permission to completely throw away any outstanding writes currently residing in the NVRAM for a disk drive. You should only use this option if you are replacing a failed disk drive, since you cannot flush the outstanding writes to the failed disk drive. Follow the instructions given in "Purging Writes from NVRAM" on page 12-3 to do this.

12.1.1 Flushing Outstanding Writes from the NVRAM

To flush writes from the NVRAM:

♦ **Enter the following command at the prompt:**

```
ssaadm sync_cache controller
```

where `controller` is the logical name of the controller, given in the form `c#`.

- If you did not get an error message after entering the command, then all outstanding writes were flushed from the NVRAM. Go directly to the appropriate section to replace the FRU.
- If you got an error message after entering the command, you will have to purge the writes from the NVRAM. Refer to "Purging Writes from NVRAM" for those instructions.

12.1.2 Purging Writes from NVRAM

The purge option gives the controller permission to throw away any outstanding writes that are currently pending in NVRAM for the physical disk you select.

Caution – This option will destroy all data currently on the NVRAM for the selected disk drive. Use this option *only* if you can no longer access the disk drive; if you can still access the disk drive, you should flush the data to the drives using the instructions given in "Flushing Outstanding Writes from the NVRAM" on page 12-2.

♦ **Enter the following command at the prompt:**

```
ssaadm purge drive
```

All outstanding writes for the disk drive are now purged. Go directly to the appropriate section to replace the FRU.

12.2 When Powering Down the SPARCstorage Array

If you are powering down the SPARCstorage Array, you must first stop all drives in the SPARCstorage Array. The procedures for this software task is given in the following section.

12.2.1 Preliminary Software Tasks

1. **Unmount the file system(s) on all disks in the array.**
2. **Stop all *database* processes that are accessing any disks in the array.**
3. **Stop all *other* processes that may be accessing any disks in the array.**
4. **Flush or purge the outstanding writes from the NVRAM, if necessary.** Refer to Section 12.1, "Flushing or Purging Outstanding Writes from NVRAM," on page 12-1 for more information.

12.2.2 Spinning Down All Drives in the SPARCstorage Array

♦ **At the prompt, enter:**

```
ssaadm stop controller
```

where `controller` is the logical name of the controller, given in the form `c#`.

Note – For more information on the `ssaadm` command, refer to the `ssaadm(1M)` man page.

12.3 When Replacing a Physical Disk Drive in the SPARCstorage Array

There are two sets of tasks to be performed when replacing a physical disk drive in the SPARCstorage Array:

- *Before* replacing a drive — Section 12.3.1 on this page
- *After* replacing a drive — Section 12.3.2 on page 12-13

12.3.1 Before Replacing a Drive

Before replacing the drive, you must first perform the preliminary software tasks.

12.3.1.1 Preliminary Software Tasks

Before replacing the drive, you must first perform the following software tasks:

1. **Unmount the file system(s) on the disks in the tray with the faulty drive.**
2. **Stop all *database* processes that are accessing the disks in the tray with the faulty drive.**
3. **Stop all *other* processes that may be accessing the disks in the tray with the faulty drive.**

4. **Flush or purge the outstanding writes from the NVRAM, if necessary.**
   Refer to Section 12.1, “Flushing or Purging Outstanding Writes from NVRAM,” on page 12-1 for more information.

5. **Determine if the failed drive is under Volume Manager control or not.**
   - If the failed drive is not under Volume Manager control, stop all the drives in the tray using the instructions in Section 12.3.1.9, “Spinning Down All Drives in a Drive Tray,” on page 12-13.
   - If the failed drive is under Volume Manager control, look at the subdisks in the failed VM disk and determine what kind of configuration each of those subdisks falls into. Then look at the flowcharts in Figure 12-1 and Figure 12-2 for the tasks that you must perform for the subdisks in each configuration.

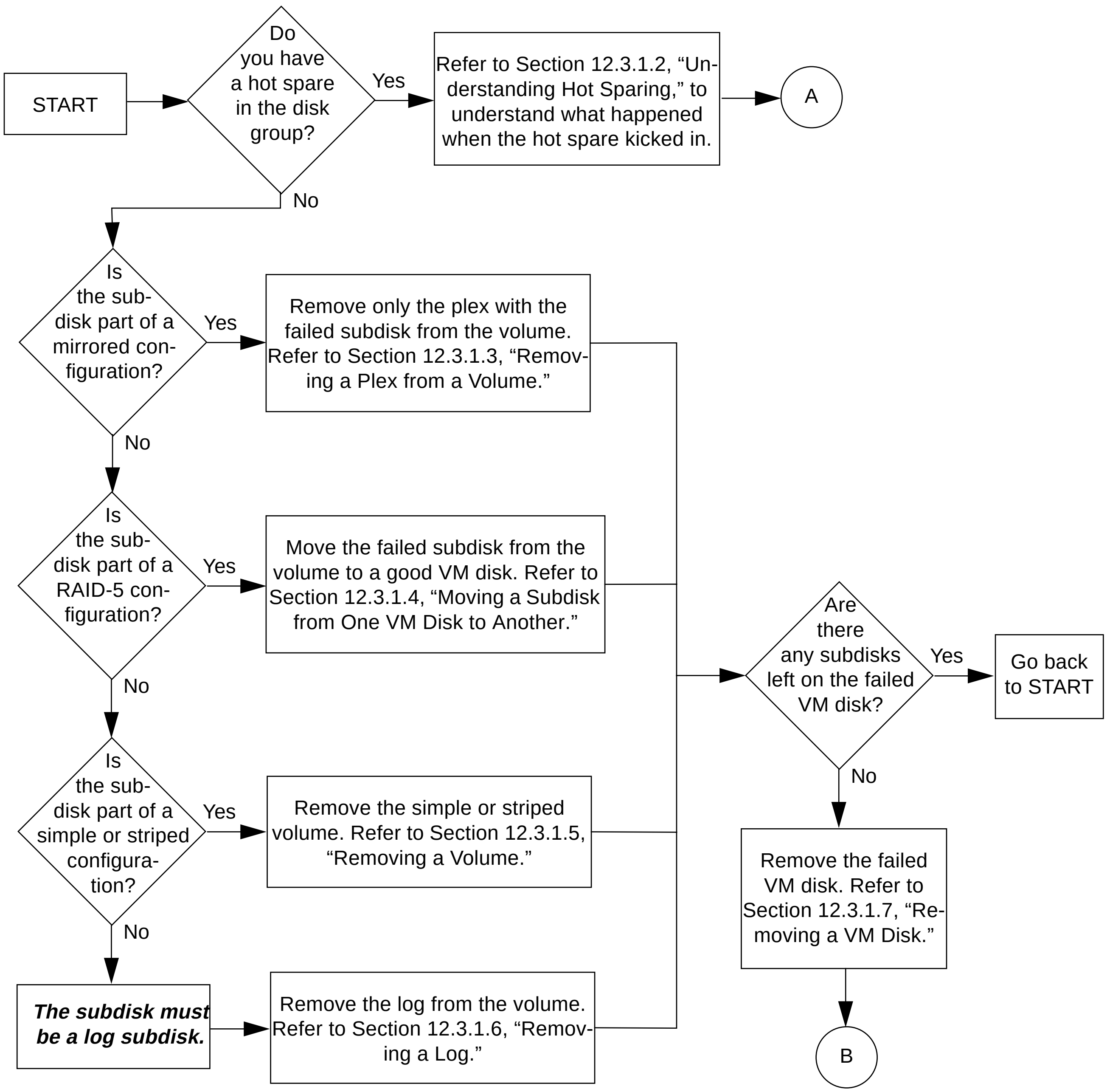

Figure 12-1 Disk Replacement Flowchart (1 of 2)

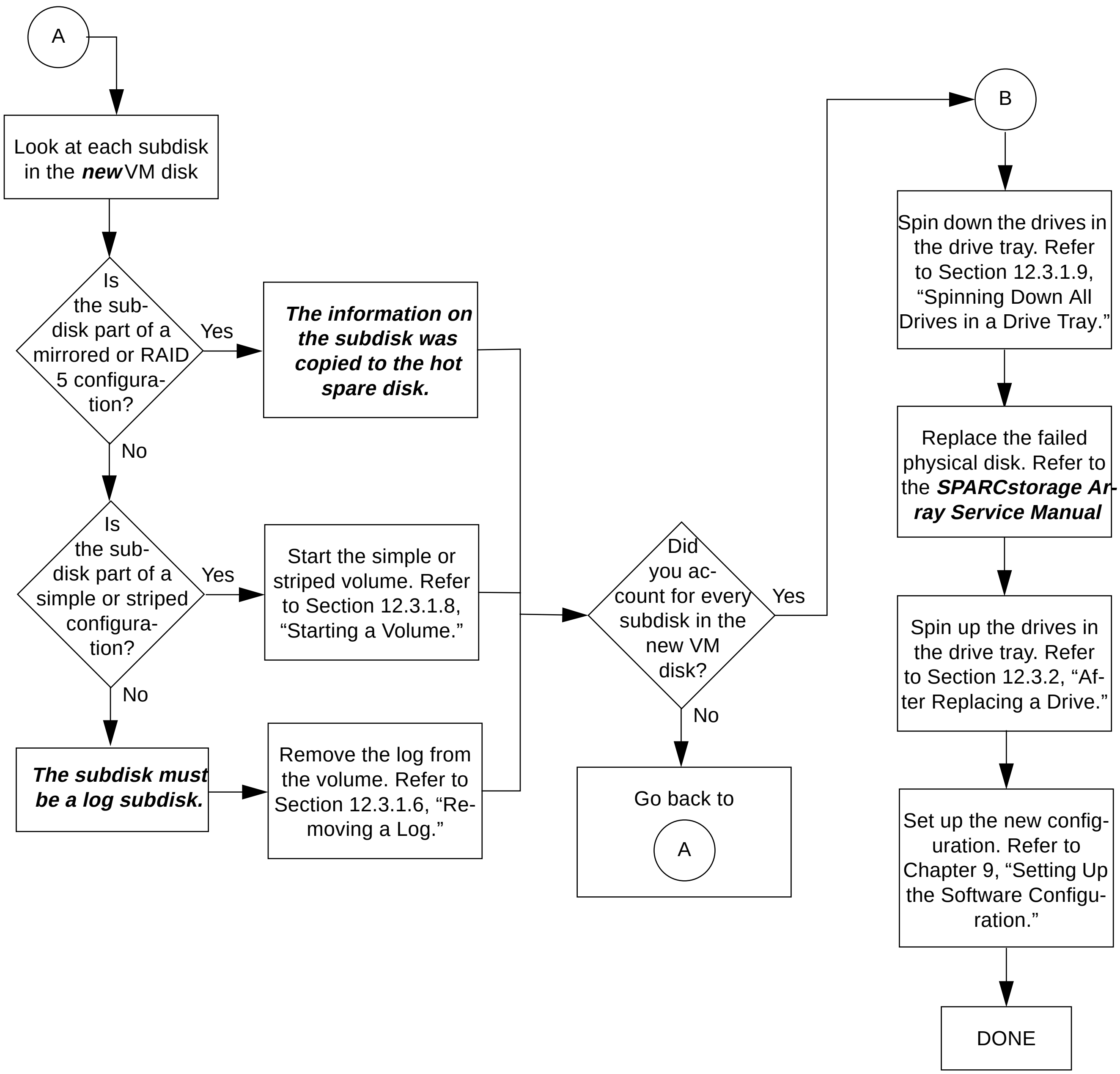

Figure 12-2 Disk Replacement Flowchart (2 of 2)

12.3.1.2 Understanding Hot Sparing

Hot-sparing can be used to automatically rebuild mirrored or RAID-5 data when a disk fails. If you have a hot spare in a disk group and a drive fails, Volume Manager will go through the following steps:

1. The VM disk assigned to the failed physical disk will disappear from the disk group view.
2. The hot spare disk will change to a VM disk and will be renamed with the failed VM disk's name.
3. The new VM disk will take all the subdisks from the failed VM disk.

For example, entering the following command for a disk group with a hot spare:

```
vxprint -g diskgroup
```

should give you output similar to the following:

```
TY NAME         ASSOC       KSTATE   LENGTH   PLOFFS   STATE    TUTIL0  PUTIL0
dg craig         craig        -        -        -        -        -       -

dm craig01       c2t0d0s2     -        2050272  -        -        -       -
dm craig02       c2t2d0s2     -        2050272  -        -        -       -
dm craig03       c2t4d0s2     -        2050272  -        -        -       -
dm craig04       c2t0d3s2     -        2050272  -        SPARE    -       -

v  vol01         fsgen        ENABLED  20480    -        ACTIVE   -       -
pl vol01-01      vol01        ENABLED  21168    -        ACTIVE   -       -
sd craig01-01    vol01-01     ENABLED  21168    0        -        -       -
```

Note that in this disk group, there is one hot spare disk (craig04) and several VM disks (craig01 through craig03).

If the VM disk craig01 failed, then Volume Manager would do the following:

1. The VM disk craig01 would disappear from the disk group view.
2. The hot spare disk craig04 would change to a VM disk and would be renamed to craig01.

3. The new VM disk craig01 would take all the subdisks from the old VM disk craig01.

12.3.1.3 *Removing a Plex from a Volume*

Removing a plex involves first dissociating it from its volume and then removing the plex and any associated subdisks completely.

Note – The last valid plex/mirror in a started or enabled volume cannot be removed, unless the volume is stopped before the remove operation.

To identify a plex, enter the following command:

```
vxprint -ht volume_name
```

The output would list a plex as the following:

```
pl vol02-01     vol02       ENABLED  5        -        ACTIVE   -       -
sd craig07-01   vol02-01    ENABLED  5        0        -        -       -
```

To dissociate and remove the plex from its volume, enter:

```
vxplex -o rm dis plex_name
```

For example, to dissociate and remove the plex `vol01-02`, enter:

```
vxplex -o rm dis vol01-02
```

12.3.1.4 *Moving a Subdisk from One VM Disk to Another*

Moving a subdisk copies the disk space contents of a subdisk onto another subdisk. If the subdisk being moved is associated with a plex, then the data stored on the original subdisk is copied to the new subdisk, the old subdisk is dissociated from the plex, and the new subdisk is associated with the plex, at the same offset within the plex as the source subdisk.

For the subdisk move operation to perform correctly, the following conditions must be met:

- The subdisks involved must be the same size.
- The subdisk being moved must be part of an active plex on an active (enabled) volume.
- The new subdisk must not be associated with any other plex.

To move a subdisk, enter:

```
vxsd mv old_subdisk_name new_subdisk_name
```

12.3.1.5 Removing a Volume

1. **Remove all references to the volume.**
2. **If the volume is mounted as a file system, unmount it:**

```
umount /dev/vx/dsk/volume_name
```

3. **If the volume is listed in** `/etc/vfstab`**, remove its entry.**
4. **Make sure the volume is stopped with the command:**

```
vxvol stop
```

   The `vxvol stop` command stops all Volume Manager activity to the volume.

5. **Remove the volume:**

```
vxedit -rf rm volume_name
```

   For example, to remove the volume `volspan`, enter:

```
vxedit -rf rm volspan
```

Note – For more information on the `vxedit` command, see the `vxedit(1M)` man page.

12.3.1.6 *Removing a Log*

Removing a log involves first dissociating it from its volume and then removing the log and any associated subdisks completely.

To identify a log, enter the following command:

```
vxprint -ht volume_name
```

The output would list a log as the following:

```
pl vol01-02     vol01       ENABLED  1008     -        LOG      -       -
sd craig05-01   vol01-02    ENABLED  1008     0        -        -       -
```

To dissociate and remove the log from its volume, enter:

```
vxplex -o rm dis plex_name
```

For example, to dissociate and remove the RAID-5 log `vol01-02`, enter:

```
vxplex -o rm dis vol01-02
```

12.3.1.7 *Removing a VM Disk*

To remove a VM disk from a disk group, enter:

```
vxdg [-g groupname] rmdisk diskname
```

where *`groupname`* is the name of the group to which the disk belongs and *`diskname`* is the name of the disk to be removed.

For example, to remove disk01 from rootdg, enter:

```
vxdg rmdisk disk01
```

Since rootdg is the default disk group, you do not need to specify it.

If the disk has subdisks on it when you try to remove it, the following error message is displayed:

```
vxdg:Disk diskname is used by one or more subdisks
```

Use the -k option to vxdg to remove device assignment. Using the -k option allows you to remove the disk in spite of the presence of subdisks. For more information, see the vxdg(1m) manual page.

Note – For more information on the vxdg command, see the vxdg(1M) man page.

Note – This procedure removes a VM disk from a disk group only until the next reboot or re-initialization, at which time it will be brought back under Volume Manager control because the private region still exists on the physical disk. If you want to completely remove the disk from Volume Manager control, you must relabel the physical disk and remove the private region.

12.3.1.8 *Starting a Volume*

1. **Locate the simple or striped volume that contains the subdisk from the failed VM disk.**

2. **Before starting the volume, confirm that it is not started by entering:**

```
vxprint -l volume_name
```

Starting a volume changes the volume state from disabled or detached to enabled. The success of this operation depends on the ability to enable a volume. If a volume cannot be enabled, it remains in its current state. To start a volume, enter:

```
vxrecover -s volume_name ...
```

To start all disabled volumes, enter:

```
vxrecover -s
```

12.3.1.9 *Spinning Down All Drives in a Drive Tray*

♦ **At the prompt, enter:**

```
ssaadm stop -t tray_number controller
```

where `tray_number` is the tray that contains the drives and `controller` is the logical name of the controller, given in the form `c#`.

Note – For more information on the `ssaadm` command, refer to the `ssaadm(1M)` man page.

12.3.2 *After Replacing a Drive*

♦ **At the prompt, enter:**

```
ssaadm start -t tray_number controller
```

where `tray_number` is the tray that contains the drives and `controller` is the logical name of the controller, given in the form `c#`.

12.4 When Replacing a Drive Tray in the SPARCstorage Array

There are two sets of tasks to be performed when replacing a drive tray in the SPARCstorage Array:

- *Before* replacing a drive tray — Section 12.4.1 on this page
- *After* replacing a drive tray — Section 12.4.2 on page 12-15

12.4.1 Before Replacing a Drive Tray

If you are replacing a drive tray in the SPARCstorage Array, you must first stop all drives in the drive tray.

12.4.1.1 Preliminary Software Tasks

1. **Unmount the file system(s) on the disks in the tray.**
2. **Stop all *database* processes that are accessing the disks in the tray.**
3. **Stop all *other* processes that may be accessing the disks in the tray.**
4. **Flush or purge the outstanding writes from the NVRAM, if necessary.**
   Refer to Section 12.1, "Flushing or Purging Outstanding Writes from NVRAM," on page 12-1 for more information.

12.4.1.2 Spinning Down All Drives in a Drive Tray

♦ **At the prompt, enter:**

```
ssaadm stop -t tray_number controller
```

where `tray_number` is the tray that contains the drives and `controller` is the logical name of the controller, given in the form `c#`.

Note – For more information on the `ssaadm` command, refer to the `ssaadm(1M)` man page.

12.4.2 After Replacing a Drive Tray

♦ **At the prompt, enter:**

```
ssaadm start -t tray_number controller
```

where `tray_number` is the tray that contains the drives and `controller` is the logical name of the controller, given in the form `c#`.

12.5 When Replacing an Array Controller

There are two sets of tasks you should perform when you replace an array controller in the SPARCstorage Array:

- *Before* replacing an array controller — page 12-15
- *After* replacing an array controller — page 12-16

12.5.1 Before Replacing an Array Controller

1. **Unmount the file system(s) on all disks in the array.**
2. **Stop all *database* processes that are accessing any disks in the array.**
3. **Stop all *other* processes that may be accessing any disks in the array.**
4. **Flush or purge the outstanding writes from the NVRAM, if necessary.** Refer to Section 12.1, "Flushing or Purging Outstanding Writes from NVRAM," on page 12-1 for more information.

12.5.1.1 Spinning Down All Drives in the SPARCstorage Array

♦ **At the prompt, enter:**

```
ssaadm stop controller
```

where `controller` is the logical name of the controller, given in the form `c#`.

Note – For more information on the `ssaadm` command, refer to the `ssaadm(1M)` man page.

12.5.2 After Replacing an Array Controller

The worldwide name for a SPARCstorage Array is stored on a PROM in the array controller. When you replace an array controller, the worldwide name for the SPARCstorage Array will change to the worldwide name stored on the PROM on the new controller.

Because paths to volumes contain the worldwide name from the *old* array controller, you must go through the following procedure so that the paths to the volumes will be updated with the *new* array controller's worldwide name:

Caution – Do *not* change the worldwide name for the new controller back to the worldwide name used by the old controller.

1. **Become superuser.**
2. **Shut down your system:**

```
# shutdown -g0 -y -i0
```

Note – This is the correct shutdown command for most systems. However, if this does not work for your system, shut down your system as you normally would.

3. **Verify that the system goes to the `ok` prompt after the shutdown is complete.**
   If the system goes to the > prompt after the shutdown, enter `n` to get to the `ok` prompt.
4. **Reboot your system using the following command:**

```
ok boot -rs
```

Once the boot cycle is completed, you should see a screen similar to the following:

```
Type Ctrl-d to proceed with normal startup,
(or give root password for system maintenance):
```

5. **Enter your root password to become superuser, then enter the following command to boot the system a second time:**

```
# reboot
```

Once the system completes the second boot-up cycle, all the paths to the volumes should be updated to contain the new controller's worldwide name.

12.6 When Replacing All Other FRUs in the SPARCstorage Array

The software tasks contained in this section apply if you are replacing any of the following SPARCstorage Array FRUs:

- Fan tray
- Power supply
- FC/OM
- Battery module
- SPARCstorage Array backplane

You only need to perform certain software tasks *before* replacing the FRU in the SPARCstorage Array; no software tasks are necessary after you replace the FRU.

12.6.1 Preliminary Software Tasks

1. **Unmount the file system(s) on any disk in the array.**
2. **Stop all *database* processes that are accessing all disks in the array.**
3. **Stop all *other* processes that may be accessing all disks in the array.**
4. **Flush or purge the outstanding writes from the NVRAM, if necessary.** Refer to Section 12.1, "Flushing or Purging Outstanding Writes from NVRAM," on page 12-1 for more information.

12.6.2 Spinning Down All Drives in the SPARCstorage Array

♦ **At the prompt, enter:**

```
ssaadm stop controller
```

where `controller` is the logical name of the controller, given in the form `c#`.

Note – For more information on the `ssaadm` command, refer to the `ssaadm(1M)` man page.

Part 4— Reference for the Graphical User Interface

Reference for the Root Window 13

This chapter provides information on screen buttons, menus, and forms relating to the root window, shown in Figure 13-1.

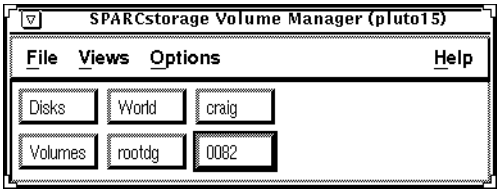

Figure 13-1 Root Window

13.1 Screen Buttons

The root window contains the following screen buttons:

- SPARCstorage Arrays
- Disks
- Volumes
- World
- rootdg

Table 13-1 Root Window Screen Buttons

Button	Description
SPARCstorage Array	Accesses the View of SPARCstorage Array screen, which gives a graphical representation of the major components of a SPARCstorage Array, including the controller, fans, battery, drive trays and disk drives.
Disks	Accesses the View of Disks screen, which displays all physical disks on the system.
Volumes	Accesses the View of Volumes screen, which displays all volumes (as well as all plexes and associated subdisks) on the system.
World	Accesses the View of World screen, which displays all objects on the system, including disks, volumes, and associated objects.
rootdg	Accesses the View of rootdg screen, which displays all objects in the default disk group, rootdg (including disks, volumes, and associated objects).

13.2 File Menu

The File menu in the root window contains the Exit menu option.

13.2.1 Exit

Access:

File ➤ Exit

Description:

This menu option closes the root window and exits the GUI completely. Although the GUI session is closed down, user preferences set during the session are retained. Since GUI operations are applied to the Volume Manager configuration as they are issued, quitting a GUI session has no effect on the configuration and does not undo any changes made.

13.3 Views Menu

The Views menu in the root window allows you to modify graphical user interface views. It contains the following menu items:

- Create a View
- Rename a View
- Remove Views
- Help

13.3.1 Create a View

Access:

Views ➤ Create a View

Description:

This option creates a new user-specified view (commonly referred to as a *user-created view* or a *user view*). The user specifies a unique name for this new view. Once created, icons can be added to this new view window by copying them over from existing views.

User-created views differ from default views (disk, volume, disk group, and world views) in that they contain copies of icons from the default views. Any operations performed from user-created views are automatically activated in the corresponding default views. Icons that appear in default views are not necessarily duplicated in user-created views.

Form:

View Create Form (described in "View Create Form" on page 13-6).

13.3.2 Rename a View

Access:

Views ➤ Rename a View

Description:

This option renames an existing user-created view. The user specifies a new, unique name for the view.

Requirements:

Only a user-created view can be renamed.

Form:

Rename View Form (described in “Rename View Form” on page 13-6).

13.3.3 Remove Views

Access:

Views ➤ Remove Views

Description:

This option permanently removes a user-created view. The removal of a user-created view results in the removal of its icons, but the objects represented by those icons are unaffected.

Requirements:

Only a user-created view can be removed.

Form:

Remove View Form (described in “Remove View Form” on page 13-7).

13.3.4 Help

Access:

Views ➤ Help

Description:

This option gives helpful information on using the Views menus.

13.3.5 Views Forms

The following forms are accessed through the Views menu (located in the GUI root window only).

13.3.5.1 View Create Form

Access:

Views ➤ Create a View

Description:

This form creates a new view window. Once created, this view is represented by a new view button in the GUI root window. This user-created view is also retained across future GUI sessions.

Fields:

Table 13-2 View Create Form

Field	Description
View Name:	The name of the view to be created. This name must be unique. The maximum length of the name is 14 characters. This field is required.

Requirements:

The name specified for the new view must be unique.

13.3.5.2 Rename View Form

Access:

Views ➤ Rename a View

Description:

This form changes the name of an existing user-created view.

Fields:

Table 13-3 Rename View Form

Field	Description
Old View Name:	The name of the view to be renamed. This name must belong to a user-created view. This field is required.
New View Name:	The name to which the existing name is to be changed. This name must be unique. The maximum length of the name is 14 characters. This field is required.

Requirements:

Only user-created views can be renamed.

The new view name must be unique.

13.3.5.3 Remove View Form

Access:

Views ➤ Remove a View

Description:

This form removes a user-created view. This form contains no fields. Instead, it lists all user-created views that currently exist.

The user selects one or more views for removal by highlighting the desired view name(s) with the LEFT button. If a view is mistakenly selected, the LEFT button can be used again to clear that view name. Removal is activated by clicking LEFT on the Apply button once, then confirming the removal by selecting Apply again (this ensures that a view is not accidentally removed).

13.4 Options Menu

The Options menu allows you to set preferences for how the GUI should operate. Once set, most preferences are saved to the file `.vxva_pref`, located in your home directory. You should not edit the `.vxva_pref` file. Once set, these preferences are maintained across future GUI runs.

The Options menu contains the following menu items:

- Show Command
- When Commands are Ready
- Logging
- Popup the Command Window
- Format of Size
- Help

13.4.1 Show Command

Access:

Options ➤ Show Command

Description:

This selection specifies whether the Command Info Window is to be displayed before each command is executed or only upon command failure.

From the Show Command menu, a cascading menu allows the user to indicate when to display the Command Info Window:

Table 13-4 Show Command Menu

Show on Error	Display the Command Info Window only when a Volume Manager or system command has failed. This is the default behavior.
Show at Start	Display the Command Info Window whenever a Volume Manager or system command is ready to be sent. This enables the user to view the actual command(s) being executed via GUI.

13.4.2 When Commands Are Ready

Access:

Options ➤ When Commands Are Ready

Description:

This selection specifies whether commands should be executed immediately or simply displayed for user review.

From the When Commands Are Ready menu, a cascading menu allows the user to indicate what GUI is to do when a command is issued:

Table 13-5 When Commands Are Ready Menu

Execute Commands	Automatically execute commands as soon as they are issued. This is the default behavior.
Show Commands Only	Display commands in the Command Info Window for user review rather than executing them. Upon approval, the user can execute the displayed command directly from the Command Info Window by highlighting that command and then using the Execute menu.

13.4.3 Logging

Access:

Options ➤ Logging

Description:

This selection starts or stops logging GUI commands. Logging records all commands sent to the Volume Manager or the system by the GUI in a specified file.

From the Logging menu, a cascading menu allows the user to indicate whether logging should be activated or deactivated:

Table 13-6 Logging Menu

Start	Begin recording all commands to a log file. A log file is created if one does not already exist. The file to be used for logging must be specified in the resulting form. If the file exists, the user must have permission to write to that file. The log information will be appended to the end of the specified file.
Stop	Stop recording all commands to the log file. When logging is discontinued, the user is responsible for remembering the name of the log file that was used.

Note – Unlike other user preferences, the logging setting is not saved across GUI sessions.

Form:

Log File Form (described in “Log File Form” on page 13-12).

13.4.4 Popup the Command Window

Access:

Options ➤ Popup the Command Window

Description:

This menu selection accesses and displays the Command Info Window. This window displays current and previous commands, along with the status of each command. Once accessed in this way, the Command Info Window remains visible until it is closed through its File menu.

See “Command Info Window” on page 13-12 for further details.

13.4.5 Format of Size

Access:

Options ➤ Format of Size

Description:

This selection specifies the units (megabytes, kilobytes, or sectors) to be used for size-related output. The unit of size is set to megabytes until the user resets it.

In properties forms, length values are displayed as sectors (s), kilobytes (k), or megabytes (m). If the size cannot be cleanly converted into kilobytes or megabytes, it is displayed in sectors instead (even though another format of size preference may be set).

The preferred format of size applies to output only and does not impact input in any way. Input typically defaults to sectors, unless megabytes or kilobytes are specified.

From the Format of Size menu, a cascading menu allows the user to select the units of size.

13.4.6 Help

Access:

Options ➤ Help

Description:

This selection gives helpful information on using the Options menus.

13.4.7 Options Forms

The following forms are accessed through the Options menu.

13.4.7.1 Log File Form

Access:

Options ➤ Logging ➤ Start

Description:

This form specifies the file to be used for logging purposes.

Fields:

Table 13-7 Log File Form

Field	Description
Log File:	The name of the file (and path name) to be used to store the command log. If no path is specified, the file is created in the directory from which the GUI session was started. The maximum length of the path specified here is 127 characters.

Requirements:

The user must have permissions appropriate to access and write to the named file (and any directories in its path).

If a path name is included, it must be valid.

13.5 Command Info Window

The Command Info Window is accessed through the Options menu. Once accessed, this window displays information on current and previous commands executed through the GUI.

13.5.1 Execute Menu

In addition to viewing or previewing commands, the Command Info Window can be used to execute commands directly through its Execute menu. This is useful for re-executing commands or for executing commands that were previously only shown in this window.

13.5.1.1 Execute

Access:

Execute ➤ Execute

Description:

This selection executes the command highlighted in the Command History section of the window. GUI sends the highlighted command to the Volume Manager or the system for execution.

Requirements:

A single command in the Command History section of the window must be highlighted.

13.5.1.2 Execute With Force Option

Access:

Execute ➤ Execute With Force Option

Description:

This selection forcefully executes the Volume Manager command highlighted in the Command History section of the window (using the `-f` option). This option effectively forces the Volume Manager to complete an operation that is considered unsafe and should therefore be used only when the user is certain that an operation should be performed in this way.

Caution – This is a dangerous operation, which can cause irreparable loss of data and should only be used when absolutely necessary.

Requirements:

The force option works for some Volume Manager commands only (those that take the `-f` option). The `-f` option does not apply to file system operations offered through the GUI.

This option should only be used when the user is sure that an operation should succeed, even though Volume Manager error checking prevents it.

13.6 Help Menu

The Help menu contains the following menu items:

- Help on Views
- Help on Options
- Help on Root Window
- Help on Help

13.6.1 Help on Views

Access:

Help ➤ Help on Views

13.6.2 Help on Options

Access:

Help ➤ Help on Options

13.6.3 Help on Root Window

Access:

Help ➤ Help on Root Window

13.6.4 Help on Help

Access:

Help ➤ Help on Help

Reference for the View of SPARCstorage Array 14

This chapter provides information on menus and forms relating to the View of SPARCstorage Array, shown in Figure 14-1. Note that Figure 14-1 shows the View of a SPARCstorage Array Model 100; while the View of a SPARCstorage Array Model 200 looks slightly different, the menu options are the same.

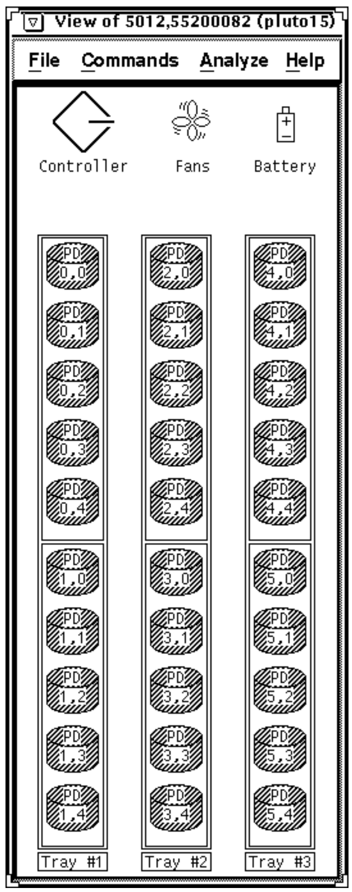

Figure 14-1 View of SPARCstorage Array Model 100

14.1 File Menu

The File menu in the View of SPARCstorage Array contains the following options:

- Close
- Exit

14.1.1 Close

Access:

File ➤ Close

Description:

This option closes the View of SPARCstorage Array only.

14.1.2 Exit

Access:

File ➤ Exit

Description:

This option closes the View of SPARCstorage Array and exits the Visual Administrator completely. Although the Visual Administrator session is closed down, user preferences set during the session are retained. Since the Visual Administrator operations are applied to the Volume Manager configuration as they are issued, quitting a Visual Administrator session has no effect on the configuration and does not undo any changes.

14.2 Commands Menu

The Commands menu in the View of SPARCstorage Array contains the following options:

- Controller
- Tray
- Disk Drive
- Volume Manager
- Help

14.2.1 Controller

The Controller option in the View of SPARCstorage Array contains the following options:

- Sync Cache
- Fast Writes
- Start All Drives
- Stop All Drives
- Release All Drives
- Reserve All Drives
- Purge Cache
- Help

14.2.1.1 Sync Cache

Access:

Commands ➤ Controller ➤ Sync Cache

Description:

This menu option flushes out all outstanding writes for all drives addressed to the controller from the subsystem's NVRAM to the drives.

14.2.1.2 Fast Writes

Access:

Commands ➤ Controller ➤ Fast Writes

Description:

This menu option turns on and off the fast option for all the drives in the SPARCstorage Array. `Enable` turns on fast writes, `Disable` turns off fast writes, and `Synchronous Only` turns on fast writes for synchronous writes only.

14.2.1.3 Start All Drives

Access:

Commands ➤ Controller ➤ Start All Drives

Description:

This option spins up all drives in the SPARCstorage Array.

14.2.1.4 Stop All Drives

Access:

Commands ➤ Controller ➤ Stop All Drives

Description:

This option spins down all drives in the SPARCstorage Array.

14.2.1.5 Release All Drives

Access:

Commands ➤ Controller ➤ Release All Drives

Description:

This option releases all drives in the SPARCstorage Array from the reserved state.

Requirements:

The drives addressed to the controller should have been reserved for exclusive use by the issuing host using the Reserve option before they can be released.

14.2.1.6 Reserve All Drives

Access:

Commands ➤ Controller ➤ Reserve All Drives

Description:

This option reserves all drives in the SPARCstorage Array for exclusive use by the issuing host.

14.2.1.7 Purge Cache

Access:

Commands ➤ Controller ➤ Purge Cache

Description:

This menu option gives the controller permission to throw away any outstanding writes that are currently pending in NVRAM for all drives in the SPARCstorage Array.

14.2.1.8 Help

Access:

Commands ➤ Controller ➤ Help

Description:

This option gives extra information on the Controller options. The Help option is explained in greater detail in Section 14.4, “Help Menu.”

14.2.2 Tray

The Tray option in the View of SPARCstorage Array contains the following options:

- Start All Drives In Tray
- Stop All Drives In Tray
- Help

14.2.2.1 Start All Drives In Tray

Access:

Commands ➤ Tray ➤ Start All Drives In Tray

Description:

This option spins up all drives in a specific drive tray in the SPARCstorage Array.

14.2.2.2 Stop All Drives in Tray

Access:

Commands ➤ Tray ➤ Stop All Drives In Tray

Description:

This option spins down all drives in a specific drive tray in the SPARCstorage Array.

14.2.2.3 Help

Access:

Commands ➤ Tray ➤ Help

Description:

This option gives extra information on the Tray options. The Help option is explained in greater detail in Section 14.4, “Help Menu.”

14.2.3 Disk Drive

The Disk Drive option in the View of SPARCstorage Array contains the following options:

- Sync Cache
- Fast Writes
- Start
- Stop
- Reserve
- Release
- Purge Cache
- Help

14.2.3.1 Sync Cache

Access:

Commands ➤ Disk Drive ➤ Sync Cache

Description:

This menu option flushes out all outstanding writes for selected drive(s) from the subsystem’s NVRAM to the drives

14.2.3.2 Fast Writes

Access:

Commands ➤ Disk Drive ➤ Fast Writes

Description:

This menu option turns on and off the fast option for selected drive(s) in the SPARCstorage Array. `Enable` turns on fast writes, `Disable` turns off fast writes, and `Synchronous Only` turns on fast writes for synchronous writes only.

14.2.3.3 Start

Access:

Commands ➤ Disk Drive ➤ Start

Description:

This option spins up the selected disk drive(s).

14.2.3.4 Stop

Access:

Commands ➤ Disk Drive ➤ Stop

Description:

This option spins down the selected disk drive(s).

14.2.3.5 Reserve

Access:

Commands ➤ Disk Drive ➤ Reserve

Description:

This option reserves the selected disk drive(s) for exclusive use by the issuing host.

14.2.3.6 Release

Access:

Commands ➤ Disk Drive ➤ Release

Description:

This option releases the selected disk drive(s) from the reserved state.

Requirements:

The disk drive(s) should have been reserved for exclusive use by the issuing host using the Reserve option before they can be released.

14.2.3.7 Purge Cache

Access:

Commands ➤ Disk Drive ➤ Purge Cache

Description:

This menu option gives the controller permission to throw away any outstanding writes that are currently pending in NVRAM for the selected disk drive(s).

14.2.3.8 Help

Access:

Commands ➤ Disk Drive ➤ Help

Description:

This option gives extra information on the Disk Drive options. The Help option is explained in greater detail in Section 14.4, "Help Menu."

14.2.4 Volume Manager Menu

The Volume Manager menu provides access to the following operations:

- Initialize Disk
- New Disk Group
- Add Disk
- Advanced Disk Operations
- Help

14.2.4.1 Initialize Disk

Access:

Commands ➤ Volume Manager ➤ Initialize Disk

Description:

This operation defines a new disk with a name specified by you.

Requirements:

At least one physical disk must be selected.

Form:

Initialize Disk Form (described in the “Disk Forms” section).

14.2.4.2 New Disk Group

Access:

Commands ➤ Volume Manager ➤ New Disk Group

Description:

This operation defines a new disk group with a name specified by you. The new disk group contains one or more VM disks corresponding to the partition(s) selected by you.

Requirements:

At least one physical disk must be selected.

Form:

Initialize Disk Group Form (described in the "Disk Group Forms" section).

14.2.4.3 Add Disk

Access:

Commands ➤ Volume Manager ➤ Add Disk

Description:

This operation adds a VM disk corresponding to the selected partition icon to a disk group. This involves creating a disk media record for the disk to be added. Partitions representing disks that already belong to disk groups cannot be added to disk groups.

Requirements:

One physical disk icon must be selected.

The selected physical disk cannot already belong to a disk group.

Only one disk can be added to a disk group at a time.

Form:

Add Disk Form (described in the "Disk Group Forms" section).

14.2.4.4 Advanced Disk Operations

The Advanced Disk Operations menu provides access to the following operations:

- Initialize
- Define
- Remove
- Online
- Offline
- Help

Initialize

Access:

Commands ➤ Volume Manager ➤ Advanced Disk Operations ➤ Initialize

Description:

This operation identifies a disk to the Volume Manager and initializes the disk for Volume Manager use. This involves installing a disk header and writing an empty configuration on the disk. A disk access record is created for the disk, unless such a record already exists.

Requirements

The disk should not already be initialized.

Form:

Disk Init Form (described in “Disk Init Form” on page 14-17).

Define

Access:

Commands ➤ Volume Manager ➤ Advanced Disk Operations ➤ Define

Description:

This operation defines a disk access record, which enables the Volume Manager to scan the disk. This makes the disk accessible, but does not initialize the disk.

Form:

Define Disk Form (described in “Define Disk Form” on page 14-18).

Remove

Access:

Commands ➤ Volume Manager ➤ Advanced Disk Operations ➤ Remove

Description:

This operation removes the VM disk associated with the selected partition(s) from Volume Manager control by removing the associated disk access records. If all partitions on a given disk are selected for removal at once, the disk is effectively removed from Volume Manager control.

Requirements:

At least one partition icon corresponding to a VM disk must be selected.

The VM disks corresponding to the selected partition(s) cannot belong to a disk group at the time of removal.

Online

Access:

Commands ➤ Volume Manager ➤ Advanced Disk Operations ➤ Online

Description:

This operation places the disk access record on a specified partition in an online state. During searches for disk IDs or members of a disk group, online disks are checked.

Form:

Disk Online Form (described in “Disk Online Form” on page 14-19).

Offline

Access:

Commands ➤ Volume Manager ➤ Advanced Disk Operations ➤ Offline

Description:

This operation places the disk access record on the selected partition(s) in an offline state. During searches for disk IDs or members of a disk group, offline disks are ignored.

Requirements:

At least one partition icon must be selected.

The disks corresponding to the selected partitions must be initialized.

The selected partition icon cannot be in use (shaded and associated with a VM disk).

14.2.4.5 Help

Access:

Commands ➤ Volume Manager ➤ Help

Description:

This option gives extra information on the Disk Drive options. The Help option is explained in greater detail in Section 14.4, “Help Menu.”

14.2.5 Volume Manager Forms

Some Volume Manager operations result in the appearance of forms, which must be completed in order for that operation to proceed. Most forms provide a Help button, which contains information relevant to the fields and other aspects of that particular form.

14.2.5.1 Initialize Disk Group Form

Access:

Commands ➤ Volume Manager ➤ Initialize

Description:

This form defines a new disk group consisting of selected disks.

Fields:

Table 14-1 Initialized Disk Group Form

Field	Description
Disk group:	The name of the new disk group. This must be a valid and unique name. This field is required. This is a read / write field.

14.2.5.2 Add Disk Form

Access:

Commands ➤ Disk Group ➤ Add Disk

Description:

This form adds a VM disk to a disk group.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 14-2 Add Disk Group Form

Field	Description
Disk group:	The name of the disk group to which the VM disk is to be added. This must be a valid disk group. This field is required.
Disk media name:	The name of the VM disk to be created. The disk media name must be unique. By default, a unique name is generated. If this field is left blank, then the disk access name is used.

14.2.5.3 Disk Init Form

Access:

Advanced-Ops ➤ Disk ➤ Initialize

Description:

This form initializes a disk for Volume Manager use.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 14-3 Disk Init Form

Field	Description
Public Device:	The pathname of the device node that represents a partition available for use. This name must be a valid entry in /dev/rdsk/. A name of the form c#t#d#s# is appropriate.
Device Type:	The desired disk type. The simple type (default) assumes that the public and private regions are stored on the same disk partition, with the public region following the private region. The sliced type assumes that the public and private regions are stored on different disk partitions. The nopriv type has no private region and log and configuration copies cannot be written to the disk.
Public length (0 for whole disk):	The length of the public section of the disk. If zero is provided as the length, the Volume Manager computes a default value from available partition table information. This length must be valid and cannot exceed the length of the disk.
Private Length:	The length of the private region of the disk. When one is not specified, the Volume Manager chooses a default value. This length must be valid and cannot exceed the length of the disk. This field is optional.
Number of config copies:	The number of configuration copies to be stored in the private section of this disk. The default value is two copies.
Comment:	A comment appropriate for this Volume Manager disk. The maximum length of the comment is 40 characters. This field is optional.

14.2.5.4 Define Disk Form

Access:

Advanced-Ops ➤ Disk ➤ Define

Description:

This form defines a disk.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 14-4 Define Disk Form

Field	Description
Public Device:	The pathname of the device node that represents a partition available for use. This name must be a valid entry in /dev/rdsk/. A name of the form c#t#d#s# is appropriate.
Device Type:	The desired disk type. The simple type (default) assumes that the public and private regions are stored on the same disk partition, with the public region following the private region. The sliced type assumes that the public and private regions are stored on different disk partitions. The nopriv type has no private region and log and configuration copies cannot be written to the disk.
Public Length (0 for whole disk):	The length of the public section of the disk. If zero is provided as the length, the Volume Manager computes a default value from available partition table information. This length must be valid and cannot exceed the length of the disk.
Offline:	Indicates whether to initially place the disk in the offline state. The default is No.
Comment:	A comment appropriate for this Volume Manager disk. The maximum length of the comment is 40 characters. This field is optional.

14.2.5.5 Disk Online Form

Access:

Advanced-Ops ➤ Disk ➤ Online

Description:

This form brings a disk online.

Table 14-5 Disk Online Form

Field	Description
Device name:	The disk access name of the disk to be brought online. This must be a valid disk access name. This field is required.

Access:

Commands ➤ Volume Manager ➤ Advanced Disk Operations

14.3 Analysis Menu

Analysis is the Visual Administrator's way of displaying statistics on the performance of various Volume Manager objects. Statistics are displayed both visually (via color or pattern) and numerically (via pop-up statistics forms).

The Analysis option in the View of SPARCstorage Array contains the following options:

- Start
- Stop
- Parameters
- Help

14.3.1 Start

Access:

Analysis ➤ Start

Description:

This selection begins analysis of the selected icon(s). These icons are added to the list of objects being analyzed. Once analysis is activated, the selected icons begin to display information about their performance characteristics.

Requirements:

At least one icon must be selected.

14.3.2 Stop

Access:

Analysis ➤ Stop

Description:

This selection terminates analysis of the selected icon(s). These icons are removed from the list of objects being analyzed. When analysis stops, the selected icons return to their pre-analysis states. When analysis is stopped for one icon, other icons undergoing analysis are not affected.

Requirements:

At least one icon must be selected.

The selected icon(s) must be undergoing analysis.

14.3.3 Parameters

Access:

Analysis ➤ Parameters

Description:

This selection accesses the Analysis Parameters form, which sets user preferences for how analysis is to be conducted.

Form:

The Analysis Parameters form has the following fields, shown in Figure 14-2:

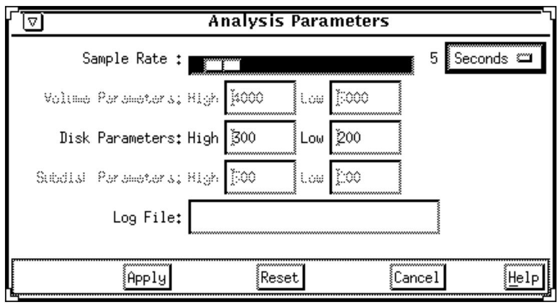

Figure 14-2 Analysis Parameter Form

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 14-6 Analysis Parameters Form

Field	Description
Sample Rate:	Determines the time interval between data samples. This field is divided into two sections: the slider bar selects the interval (1-60) and the menu to the right selects units of time (seconds or minutes). The default is 5 seconds. A shorter interval means the data will be updated more often, but is also a higher load on the system.
Disk Parameters:	Specifies the high and low values that decide the coloring (or pattern) of the VM disk icons.
Log File:	The name of the file to be used for the statistics log. If the file does not already exist, it will be created. The filename is taken to be relative unless a pathname is given. To stop logging to the file, the filename text in this field must be erased. This field is optional. This log file is a binary file. In order to view the log file, `/usr/X/bin/valog2` text *filename* must be run on this file to process it for viewing.

Requirements:

For each set of high / low parameters, the high parameter must be greater than the low parameter.

The user must have access to the specified log file.

14.3.4 Help

Access:

Analysis ➤ Help

Description:

This option gives extra information on the Analysis options. The Help option is explained in greater detail in Section 14.4, “Help Menu.”

14.4 Help Menu

The Visual Administrator provides an extensive Help facility, which is accessible from most windows or operations.

The Help option in the View of SPARCstorage Array contains the following options:

- Help on Commands
- Help on Analysis
- Help on Help

14.4.1 Help on Commands

Access:

Help ➤ Help on Commands

Description:

This option gives extensive help information on the Commands menu. It also tells you if there are other related areas where you can find information.

14.4.2 Help on Analysis

Access:

Help ➤ Help on Analysis

Description:

This option gives extensive help information on the Analysis menu. It also tells you if there are other related areas where you can find information.

14.4.3 Help on Help

Access:

Help ➤ Help on Help

Description:

This option gives extensive help information on the Help menu. It also tells you if there are other related areas where you can find information.

Reference for Volume Manager Views 15

This chapter provides information on menus and forms relating to the Volume Manager views, which include:

- View of Disks
- View of Volumes
- View of World
- View of disk groups

A View of disk groups is shown in Figure 15-1.

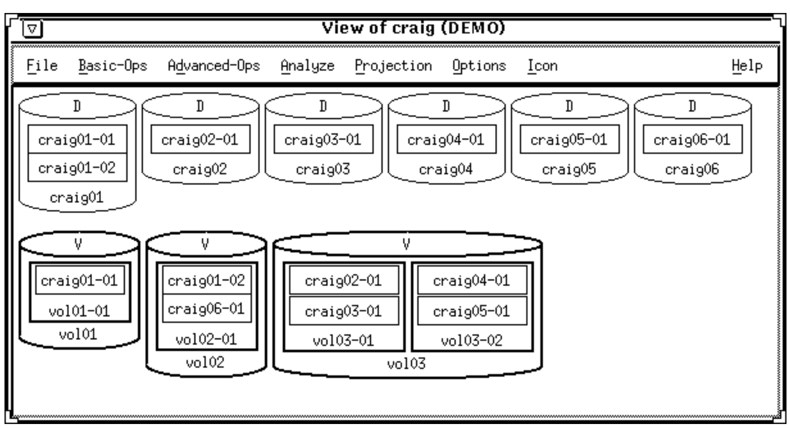

Figure 15-1 View of Disk Group

15

15.1 File Menu

The File menu in the Volume Manager view contains the following options:

- Close
- Exit

15.1.1 Close

Access:

File ➤ Close

Description:

This option closes the Volume Manager view only.

15.1.2 Exit

Access:

File ➤ Exit

Description:

This option closes the Volume Manager view and exits the GUI completely. Although the GUI session is closed down, user preferences set during the session are retained. Since the GUI operations are applied to the Volume Manager configuration as they are issued, quitting a GUI session has no effect on the configuration and does not undo any changes.

15.2 Basic-Ops Menu

The Basic-Ops menu in the Volume Manager view contains the following options:

- File System Operations
- Volume Operations
- Disk Operations
- Help

15.2.1 File System Operations

The File System Operations menu contains the following options:

- Create
- Make File System
- Mount
- Unmount
- Check File System (fsck)
- Resize
- Display Properties
- Help

15.2.1.1 Create

Access:

Basic-Ops ➤ File System Operations ➤ Create

Description:

This operation creates a file system on an underlying volume. This is done by creating a volume on one or more disks and then creating the file system on that volume.

You may select one or more disks on which to create the volume (providing that there is sufficient space on the disks). If no disks are specified, the Volume Manager automatically determines which disks are to be used based on available free space.

From the Create menu, you select the type of volume to be created from a cascading menu listing three of the basic types of volumes:

Table 15-1 Create Menu

Simple	Creates a simple, concatenated volume whose subdisks are arranged both sequentially and contiguously within a plex.
Striped	Creates a volume with data spread fairly evenly across multiple disks by way of striping. *Stripes* are relatively small, equally-sized fragments that are allocated alternately to the subdisks of each plex.
RAID-5	Creates a volume that uses striping to spread data fairly evenly across multiple disks in an configuration and allows independent access. It also stripes parity across all the disks in a configuration. Each stripe contains the data stripe and a parity stripe.

If a mirrored volume is desired, a simple or striped volume must be created and then mirrored using the Add Mirror option from the Volume Operations menu. RAID-5 volumes cannot be mirrored.

Requirements:

Only disks in the same disk group can be selected.

Only VM disks (disks under Volume Manager control) can be selected.

If striping is to be in effect, at least two disks are required in order for the operation to succeed.

Forms:

Simple Volume/FS Create Form (described in "Simple Volume/FS Create Form" on page 15-9).

Striped Volume/FS Create Form (described in “Striped Volume/FS Create Form” on page 15-11).

RAID-5 Volume/FS Create Form (described in “RAID-5 Volume/FS Create Form” on page 15-13).

15.2.1.2 Make File System

Access:

Basic-Ops ➤ File System Operations ➤ Make File System

Description:

This operation makes a file system on an existing volume. You select the volume on which to place the new file system, and specifies the mount point if the file system is to be mounted immediately.

Requirements:

A volume icon must be selected.

The selected volume must be enabled.

Only one mounted file system can exist on each volume.

Form:

Make File System Form (described in “Make File System Form” on page 15-15).

15.2.1.3 Mount

Access:

Basic-Ops ➤ File System Operations ➤ Mount

Description:

This operation mounts the file system that resides on the selected volume. This operation assumes that the selected volume already contains a valid file system. Volume Manager has no way of knowing whether a valid, unmounted file system already exists on a given volume. You are responsible for being aware of the existence of an unmounted file system on a volume, as well as that file system's type.

Requirements:

A volume icon must be selected.

A valid, unmounted file system must already exist on the selected volume.

Form:

Mount File System Form (described in "Mount File System Form" on page 15-16).

15.2.1.4 Unmount

Access:

Basic-Ops ➤ File System Operations ➤ Unmount

Description:

This operation unmounts the file system(s) that resides on the selected volume(s). The file system can be unmounted only if the mount point is not busy.

Requirements:

At least one volume icon must be selected.

The selected volume must contain a mounted file system.

15.2.1.5 Check File System (fsck)

Access:

Basic-Ops ➤ File System Operations ➤ Check File System (fsck)

Description:

This operation checks the file system(s) on the selected volume(s) for consistency (using `fsck`). The file system to be checked must currently be unmounted.

Requirements:

At least one volume icon must be selected.

The selected volume(s) must contain an unmounted file system.

Form:

File System Check Form (described in the "File System Forms" section).

15.2.1.6 Resize

Access:

Basic-Ops ➤ File System Operations ➤ Resize

Description:

This operation resizes the file system that resides on the selected volume. This involves first resizing the volume and then resizing its file system accordingly. Both the volume and file system can be increased to, increased by, reduced to, or reduced by a given length. Disk space is added to or removed from the mirrors associated with the volume. If new disk space is needed during the resize, it is allocated as necessary; if space becomes unused, it is added to the free space pool. A disk cannot be selected for this operation.

Requirements:

A volume icon containing a mounted file system must be selected.

A file system on a striped volume cannot be resized.

Form:

File System Resize Form (described in "File System Resize Form" on page 15-17).

15.2.1.7 Display Properties

Access:

Basic-Ops ➤ File System Operations ➤ Display Properties

Description:

Display information for file systems mounted on the system. You may select the file system for which information is to be displayed from a list of all mounted file systems. If a volume is selected, the properties for the file system that resides on that volume is displayed by default.

15.2.1.8 Help

Access:

Basic-Ops ➤ File System Operations ➤ Help

Description:

This option gives extra information on the File System Operations options. The Help option is explained in greater detail in "Help Menu."

15.2.2 File System Forms

15.2.2.1 Simple Volume/FS Create Form

Access:

Basic-Ops ➤ File System Operations ➤ Create ➤ Simple

Description:

This form creates a concatenated volume and then creates a file system on the new volume. The form is divided into two sections, one for volume creation and the other for file system creation. Most of the form fields are already set to the defaults for the creation of a new volume and file system.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 15-2 Simple Volume / FS Create Form

Field	Description
Volume name:	The name of the volume to be created. The name must be unique within this disk group. The maximum length of this field is 31 characters.
Volume size:	The desired volume size. The size should be entered as a number followed immediately by the letter k, m, or s to indicate kilobytes, megabytes, or sectors, respectively. If no unit is specified, the default is sectors. The volume size should be less than or equal to the available free space of the disks.
Usage Type:	The desired usage type. fsgen is the file system generic usage type, which assumes that the volume is being used by a file system. gen is the generic usage type, which makes no assumptions regarding the data content of the volume. The default is fsgen.
Create file system:	Indicates whether a file system is to be created. When this form is invoked from the File System Operations menu, the default is to create a file system (Yes). All fields below this field are only accessible when Yes is specified here.
Impose inode limit:	Specifies whether to force the file system to have a limit of 65,536 inodes (files). Specifying No in this field instructs the file system to create as many inodes as appropriate for the file system's size. The default is Yes.

Table 15-2 Simple Volume / FS Create Form *(Continued)*

Field	Description
Mount file system:	Indicates whether the file system should be mounted after creation. If the answer is Yes (the default), a mount point must also be specified in the next field. All fields below this field are only accessible when Yes is specified here.
Mount point:	The desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it. This field is required if the file system is to be mounted.
Mount automatically:	Indicates whether this file system should be mounted every time the system comes up (by placing an entry in /etc/vfstab). The default is Yes.

15.2.2.2 Striped Volume/FS Create Form

Access:

Basic-Ops ➤ File System Operations ➤ Create ➤ Striped

Description:

This form creates a striped volume and creates a file system on the new volume. The form is divided into two sections, one for volume creation and the other for file system creation. Most of the form fields are already set to the defaults for the creation of a new volume.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 15-3 Striped Volume / FS Create Form

Field	Description
Volume name:	The name of the volume to be created. The name must be unique within this disk group. The maximum length of this field is 31 characters.
Volume size:	The desired volume size. The size should be entered as a number followed immediately by the letter k, m, or s to indicate kilobytes, megabytes, or sectors, respectively. If no unit is specified, the default is sectors. If the size is not wholly divisible by the stripe width, Volume Manager will adjust the volume size up to the next even multiple in order to create the volume. For a striped volume, the volume size should be calculated as follows: *vol_size* = *stripe_width* * number_of_stripes * *n*, where *n* is an integer greater than zero. The volume size should be less than or equal to the available free space of the disks.
Usage Type:	The desired usage type. fsgen is the file system generic usage type, which assumes that the volume is being used by a file system. gen is the generic usage type, which makes no assumptions regarding the data content of the volume. The default is fsgen.
Number of Stripes:	The number of stripes that the volume's plex is to have. This is effectively the number of disks on which the volume is to be created. If some number of disks have already been selected, that number of stripes appears in this field. This number corresponds to the number of disks across which data will be striped. If no number is specified, Volume Manager selects an appropriate number (usually two).

Table 15-3 Striped Volume / FS Create Form *(Continued)*

Field	**Description**
Stripe width:	The width of the stripes on the mirror that this volume will have. The value specified may be optimized for the particular drive configuration, as best striping performance is achieved when the stripe width corresponds to the track width of the drive. The default value for this field is 128 sectors, chosen as a good stripe width for most systems.
Create file system:	Indicates whether a file system is to be created. When this form is invoked from the File System Operations menu, the default is to create a file system (Yes). All fields below this field are only accessible when Yes is specified here.
Impose inode limit:	Specifies whether to force the file system to have a limit of 65,536 inodes (files). Specifying No in this field instructs the file system to create as many inodes as appropriate for the file system's size. The default is Yes.
Mount file system:	Indicates whether the file system should be mounted after creation. If the answer is Yes (the default), a mount point must also be specified in the next field. All fields below this field are accessible only when Yes is specified here.
Mount point:	The desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it. This field is required if the file system is to be mounted.
Mount automatically:	Indicates whether this file system should be mounted every time the system comes up (by placing an entry in /etc/vfstab). The default is Yes.

15.2.2.3 RAID-5 Volume/FS Create Form

Access:

Basic-Ops ➤ File System Operations ➤ Create ➤ RAID-5

Description:

This form creates a RAID-5 volume and creates a file system on the new volume. The form is divided into two sections, one for volume creation and the other for file system creation. Most of the form fields are already set to the defaults for the creation of a new volume.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 15-4 RAID-5 Volume / FS Create Form

Field	Description
Volume name:	Either keep the default name or enter a different one. The name must be unique within this disk group. The maximum length of this field is 31 characters.
Volume size:	Enter the desired volume size. The size should be entered as a number followed immediately by the letter k, m, or s to indicate kilobytes, megabytes, or sectors, respectively. If no unit is specified, the default is sectors. If the size specified is not wholly divisible by the stripe width, it is rounded up before the volume is created. Since RAID 5 reserves separate space for parity information, the actual space contained in plexes is larger than the addressable sized of the volume. The size specified in this field represents the usable space.
Create log subdisk:	Indicate whether you want to create a log subdisk or not; the default is Yes. At least one log subdisk should be created for a RAID 5 volume.
Number of Columns:	This field represents the number of columns that the volume's plex is to have. Typically, the number needed for RAID 5 should be four columns or more.
Stripe width:	This is the width of the stripes on the volume's plex. The value specified may be optimized for the particular drive configuration. For best striping performance, the stripe width should correspond to the track width of the drive. The default value for this field is 32 sectors, since this is a good stripe width for most systems.
Create file system:	Indicate whether a file system is to be created. Since the default is not to create a file system, No is already selected. Select Yes if you want to create a file system on this volume.

Table 15-4 RAID-5 Volume/FS Create Form *(Continued)*

Field	Description
Mount file system:	Indicate whether the file system should be mounted after creation. If the answer is Yes (the default), a mount point must also be specified in the next field. All fields below this field are only accessible when Yes is specified here.
Mount point:	Enter the desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it.
Mount automatically:	Indicate whether this file system should be mounted every time the system comes up (by placing an entry in /etc/vfstab). Yes is the default.
Volume name:	Either leave the default name or enter a different one. The name must be unique within this disk group. The maximum length of this field is 31 characters.

15.2.2.4 Make File System Form

Access:

Basic-Ops ➤ File System Operations ➤ Make

Description:

This form makes a file system (using `mkfs`) according to your specifications.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless they are listed as read only.

Table 15-5 Make File System Form

Field	Description
Device name:	This field displays the block device on which to make the file system, which corresponds to the name of the selected volume. This field is read only and cannot be changed.
File system size:	The length of the file system to be made. If no units are specified, sectors are assumed. This length should typically correspond to the length of the volume on which the file system is to be made, although it can be altered for special circumstances.
Impose inode limit:	Specifies whether to force the file system to have a limit of 65,536 inodes (files). Specifying No in this field instructs the file system to create as many inodes as appropriate for the file system's size. The default is Yes.
Mount file system:	Indicates whether the file system should be mounted after creation. If the answer is Yes (the default), a mount point must also be specified in the next field. All fields below this field are only accessible when Yes is specified here.
Mount point:	The desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it. This field is required if the file system is to be mounted.
Mount automatically:	Indicates whether this file system should be mounted every time the system comes up (by placing an entry in /etc/vfstab). Yes is the default.

15.2.2.5 Mount File System Form

Access:

Basic-Ops ➤ File System Operations ➤ Mount

Description:

This form mounts a file system that already exists on a selected volume.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 15-6 Mount File System Form

Field	Description
Device name:	This field displays the block device on which to make the file system, which corresponds to the name of the selected volume. This field is read only and cannot be changed.
Mount point:	The desired mount point for the file system. If the specified mount point does not already exist, Volume Manager will automatically create it. Volume Manager attempts to provide a default mount point, which it obtains by scanning /etc/vfstab.
Mount automatically:	Indicates whether this file system should be mounted every time the system comes up (by placing an entry in /etc/vfstab). No is the default.

15.2.2.6 File System Resize Form

Access:

Basic-Ops ➤ File System Operations ➤ Resize

Description:

This form either grows or shrinks a file system and its underlying volume using the Volume Manager free space management resources. If new disk space is needed, it will be allocated as necessary; if space becomes unused, it will be added to the free space pool.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless they are listed as read only.

Table 15-7 File System Resize Form

Field	Description
Mount point:	This field displays the mount point of the file system to be resized. This field is read only and cannot be changed.
Volume:	This field displays the block device of the volume on which the file system resides. This field is read only and cannot be changed.
Current size:	This field displays the current size of the file system to be resized. This field is read only and cannot be changed.
Option:	The type of resize operation to be performed. This will determine whether the volume is grown or shrunk to a certain size, or grown or shrunk by a given amount. The default is Grow To. The alternative is Grow By.
Size / Amount:	Enter either the length to which or the amount by which the file system (and underlying volume) is to be resized. If Grow To is selected, this field should reflect the final size. If Grow By is selected, this field should reflect the amount by which the size should change. The new volume size should be less than or equal to the available free space of the disks.

15.2.2.7 File System Properties Form

Access:

Basic-Ops ➤ File System Operations ➤ Display Properties

Description:

This form provides detailed information on the attributes of a particular file system. This properties form contains a list of mounted file systems, from which you can select the file system whose properties are to be displayed.

Fields:

Note – All fields in this form are read only and cannot be changed.

Table 15-8 File System Properties Form

Field	Description
Mount Point:	The mount point of this file system.
Device:	The block device on which this file system resides.
Block Size:	The block size of this file system.
Default block size:	Fundamental file system block size.
Total disk space:	Number of megabytes of disk storage on this file system available on the disk.
Disk space available:	Number of megabytes of disk storage on this file system that is available for use.

15.2.3 Volume Operations

The Volume Operations menu provides the following selections:

- Create
- Remove Volumes Recursively
- Add Mirror
- Add Log
- Remove Mirror or Log
- Resize
- Snapshot
- Help

15.2.3.1 Create

Access:

Basic-Ops ➤ Volume Operations ➤ Create

Description:

This operation creates a volume from one or more disks. You may select one or more disks on which to create the volume (providing that there is sufficient space on the disks). If no disks are specified, the Volume Manager automatically determines which disks are to be used based on available free space.

From the Create menu, you select the type of volume to be created from a cascading menu listing two of the basic types of volumes:

Table 15-9 Volume Create Menu

Simple	Creates a simple, concatenated volume whose subdisks are arranged both sequentially and contiguously within a plex.
Striped	Creates a volume with data spread fairly evenly across multiple disks by way of striping. *Stripes* are relatively small, equally-sized fragments that are allocated alternately to the subdisks of each plex.
RAID-5	Creates a volume that uses striping to spread data fairly evenly across multiple disks in an configuration and allows independent access. It also stripes parity across all the disks in a configuration. Each stripe contains the data stripe and a parity stripe.

If a mirrored volume is desired, a simple or striped volume must be created and then mirrored using the Add Mirror option. You cannot mirror RAID-5 volumes.

Requirements:

Only disks in the same disk group can be selected.

Only VM disks (disks under Volume Manager control and assigned to a disk group) can be selected.

If striping is to be in effect, at least two disks are required in order for the operation to succeed.

Forms:

Simple Volume/FS Create Form (described in “Simple Volume/FS Create Form” on page 15-25).

Striped Volume/FS Create Form (described in “Striped Volume/FS Create Form” on page 15-27).

RAID-5 Volume/FS Create Form (described in “RAID-5 Volume/FS Create Form” on page 15-30).

15.2.3.2 Remove Volumes Recursively

Access:

Basic-Ops ➤ Volume Operations ➤ Remove Volumes Recursively

Description:

This operation removes the selected volume(s) and de-allocates all of the disk space set aside for that volume. It automatically removes all underlying plexes and subdisks associated with the volume.

Caution – This is a permanent operation and cannot be undone. If completed, it will be difficult or impossible to retrieve the data associated with that volume. For this reason, a confirmation window is presented if the selected volume is not ready for removal (i.e. started or enabled).

Requirements:

At least one volume icon must be selected.

The selected volume(s) cannot contain a mounted file system.

15.2.3.3 Add Mirror

Access:

Basic-Ops ➤ Volume Operations ➤ Add Mirror

Description:

This operation adds a mirror to the selected volume by associating a mirror of the correct length to the volume. The mirror effectively duplicates the information contained in the volume. Although a volume can have a single mirror, at least two are required for true mirroring (redundancy of data) to be in effect.

From the Add Mirror menu, you select the type of mirror to be added from a cascading menu listing two of the basic types of mirrors:

Table 15-10 Add Mirror Menu

Simple	Adds a simple, concatenated mirror whose subdisks are arranged both sequentially and contiguously.
Striped	Adds a mirror whose data is allocated evenly across each of its subdisks in an alternating fashion. This is accomplished with *stripes*, which are relatively small, equally-sized fragments that are allocated alternately to each subdisk.

Disks can be selected for this operation. However, the number of selected disks must be sufficient to accommodate the layout type of both the existing volume and the mirror to be added. If no disks are selected, the free space for the mirror is allocated by the Volume Manager.

Requirements:

A volume icon must be selected.

For a striped mirror, at least two disks other than those already in use by the volume must be available.

15.2.3.4 *Add Log*

Access:

Basic-Ops ➤ Volume Operations ➤ Add Log

Description:

This operation adds a log plex containing a log subdisk to a volume. For striped and concatenated volumes, this adds a Dirty Region Log (DRL). For RAID-5, this operation adds a RAID-5 log.

15.2.3.5 Remove Mirror or Log

Access:

Basic-Ops ➤ Volume Operations ➤ Remove Mirror or Log

Description:

This operation removes the selected mirror or log, along with any associated subdisks.

Requirements:

A mirror or log icon must be selected.

The last valid mirror in a started or enabled volume cannot be removed, unless the volume is stopped before the remove operation.

15.2.3.6 Resize

Access:

Basic-Ops ➤ Volume Operations ➤ Resize

Description:

This operation resizes the selected volume. The volume can be increased to, increased by, reduced to, or reduced by a given length. This involves adding or removing disk space to/from the plexes associated with the volume.

If new disk space is needed during the resize, it is allocated as necessary; if space becomes unused, it is added to the free space pool. A disk cannot be selected for this operation.

Requirements:

A volume icon must be selected.

A volume containing a mounted file system cannot be shrunk.

Form:

Volume Resize Form (described in “Volume Resize Form” on page 15-31).

15.2.3.7 Snapshot

Access:

Basic-Ops ➤ Volume Operations ➤ Snapshot

Description:

This operation backs up a volume by creating a snapshot image of that volume. This is a convenient way of performing online backup with minimal interruption.

This operation invokes the Volume Manager snapshot approach, in which the snapshot operation creates a new volume that is a snapshot of an existing volume. This is done by creating a mirror of the existing volume (creating and associating a plex) using disk space from the pool of free disk space. The mirror is brought up to date (this may take some time) and a separate (snapshot) volume is then created for that mirror. The snapshot volume represents a consistent copy of the original volume at the time the snapshot was begun. The snapshot volume can be used to make a backup of the original volume without stopping it. After the backup is made, the snapshot volume can be removed without losing any data.

From the Snapshot menu, a cascading menu allows you to first create the snapshot mirror and then the snapshot volume:

Table 15-11 Snapshot Menu

Snapstart	Start the snapshot procedure by creating a snapshot mirror within the volume to be backed up. It takes a variable amount of time to update the new mirror, during which time the snapshot mirror icon is grayed out.
Snapshot	At a convenient time (preferably after warning users to reduce activity briefly), create another volume for the snapshot mirror. This portion of the procedure should take only seconds to complete.

Requirements:

A volume icon must be selected.

There must be sufficient free disk space to accommodate the snapshot volume.

Form:

Snapshot Form (described in “Snapshot Form” on page 15-32).

15.2.4 Volume Forms

Some volume operations result in the appearance of forms, which must be completed in order for that operation to proceed. Most forms provide a Help button, which contains information relevant to the fields and other aspects of that particular form.

15.2.4.1 Simple Volume/FS Create Form

Access:

Basic-Ops ➤ Volume Operations ➤ Create ➤ Simple

Description:

This form creates a concatenated volume and optionally creates a file system on the new volume. The form is divided into two sections, one for volume creation and the other for file system creation. Most of the form fields are already set to the defaults for the creation of a new volume; the file system fields are grayed out because the default is not to add a file system to the volume.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. All fields in this form are read / write fields.

Table 15-12 Simple Volume / FS Create Form, Part One

Field	Description
Volume name:	The name of the volume to be created. The name must be unique within this disk group. The maximum length of this field is 31 characters.
Volume size:	The desired volume size. The size should be entered as a number followed immediately by the letter k, m, or s to indicate kilobytes, megabytes, or sectors, respectively. If no unit is specified, the default is sectors. The volume size should be less than or equal to the available free space of the disks.
Usage Type:	The desired usage type. fsgen is the file system generic usage type, which assumes that the volume is being used by a file system. gen is the generic usage type, which makes no assumptions regarding the data content of the volume. The default is fsgen.
Create file system:	Indicates whether a file system is to be created. When this form is invoked from the Volume Operations menu, the default is not to create a file system (No). All fields below this field are only accessible when Yes is specified here.

The following fields only apply if the Create file system: field is set to Yes. Otherwise, these fields are inaccessible.

Table 15-13 Simple Volume / FS Create Form, Part Two

Field	**Description**
Impose inode limit:	Specifies whether to force the file system to have a limit of 65,536 inodes (files). Specifying No in this field instructs the file system to create as many inodes as appropriate for the file system's size. The default is Yes.
Mount file system:	Indicates whether the file system should be mounted after creation. If the answer is Yes (the default), a mount point must also be specified in the next field. All fields below this field are only accessible when Yes is specified here.
Mount point:	The desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it. This field is required if the file system is to be mounted.
Mount automatically:	Indicates whether this file system should be mounted every time the system comes up (by placing an entry in /etc/vfstab). The default is Yes.

15.2.4.2 *Striped Volume/FS Create Form*

Access:

Basic-Ops ➤ Volume Operations ➤ Create ➤ Striped

Description:

This form creates a concatenated volume and optionally creates a file system on the new volume. The form is divided into two sections, one for volume creation and the other for file system creation. Most of the form fields are already set to the defaults for the creation of a new volume; the file system fields are grayed out because the default is not to add a file system to the volume.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. All fields in this form are read / write fields.

Table 15-14 Striped Volume / FS Create Form, Part One

Field	Description
Volume name:	The name of the volume to be created. The name must be unique within this disk group. The maximum length of this field is 31 characters.
Volume size:	The desired volume size. The size should be entered as a number followed immediately by the letter k, m, or s to indicate kilobytes, megabytes, or sectors, respectively. If no unit is specified, the default is sectors. If the size is not wholly divisible by the stripe width, Volume Manager will adjust the volume size up to the next even multiple in order to create the volume. For a striped volume, the volume size should be calculated as follows: *vol_size* = *stripe_width* * number_of_stripes * *n*, where *n* is an integer greater than zero. The volume size should be less than or equal to the available free space of the disks.
Usage Type:	The desired usage type. fsgen is the file system generic usage type, which assumes that the volume is being used by a file system. gen is the generic usage type, which makes no assumptions regarding the data content of the volume. The default is fsgen.

Table 15-14 Striped Volume/FS Create Form, Part One

Field	**Description**
Number of Stripes:	The number of stripes that the volume's plex is to have. This is effectively the number of disks on which the volume is to be created. If some number of disks have already been selected, that number of stripes appears in this field. This number corresponds to the number of disks across which data will be striped. If no number is specified, Volume Manager selects an appropriate number (usually 2).
Stripe width:	The width of the stripes on the plex that this volume will have. The value specified may be optimized for the particular drive configuration. The default value for this field is 128 sectors, chosen as a good stripe width for most systems.
Create file system:	Indicates whether a file system is to be created. When this form is invoked from the Volume Operations menu, the default is not to create a file system (No). All fields below this field are only accessible when Yes is specified here.

The following fields only apply if the Create file system: field is set to Yes. Otherwise, these fields are inaccessible.

Table 15-15 Striped Volume/FS Create Form, Part Two

Field	**Description**
Impose inode limit:	Specifies whether to force the file system to have a limit of 65,536 inodes (files). Specifying No in this field instructs the file system to create as many inodes as appropriate for the file system's size. The default is Yes.
Mount file system:	Indicates whether the file system should be mounted after creation. If the answer is Yes (the default), a mount point must also be specified in the next field. All fields below this field are only accessible when Yes is specified here.
Mount point:	The desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it. This field is required if the file system is to be mounted.
Mount automatically:	Indicates whether this file system should be mounted every time the system comes up (by placing an entry in /etc/vfstab). The default is Yes.

15.2.4.3 RAID-5 Volume/FS Create Form

Access:

Basic-Ops ➤ File System Operations ➤ Create ➤ RAID-5

Description:

This form creates a RAID-5 volume and creates a file system on the new volume. The form is divided into two sections, one for volume creation and the other for file system creation. Most of the form fields are already set to the defaults for the creation of a new volume.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 15-16 RAID-5 Volume / FS Create Form

Field	Description
Volume name:	Either keep the default name or enter a different one. The name must be unique within this disk group. The maximum length of this field is 31 characters.
Volume size:	Enter the desired volume size. The size should be entered as a number followed immediately by the letter k, m, or s to indicate kilobytes, megabytes, or sectors, respectively. If no unit is specified, the default is sectors. If the size specified is not wholly divisible by the stripe width, it is rounded up before the volume is created. Since RAID 5 reserves separate space for parity information, the actual space contained in plexes is larger than the addressable sized of the volume. The size specified in this field represents the usable space.
Create log subdisk:	Indicated whether you want to create a log subdisk or not; the default is Yes. At least one log subdisk should be created for a RAID 5 volume.
Number of Columns:	This field represents the number of columns that the volume's plex is to have. Typically, the number needed for RAID 5 should be four columns or more.

Table 15-16 RAID-5 Volume/FS Create Form

Field	Description
Stripe width:	This is the width of the stripes on the volume's plex. The value specified may be optimized for the particular drive configuration. For best striping performance, the stripe width should correspond to the track width of the drive. The default value for this field is 32 sectors, since this is a good stripe width for most systems.
Create file system:	Indicate whether a file system is to be created. Since the default is not to create a file system, No is already selected. Select Yes if you want to create a file system on this volume.
Mount file system:	Indicate whether the file system should be mounted after creation. If the answer is Yes (the default), a mount point must also be specified in the next field. All fields below this field are only accessible when Yes is specified here.
Mount point:	Enter the desired mount point for the new file system. If the specified mount point does not already exist, Volume Manager will automatically create it.
Mount automatically:	Indicate whether this file system should be mounted every time the system comes up (by placing an entry in /etc/vfstab). Yes is the default.
Volume name:	Either leave the default name or enter a different one. The name must be unique within this disk group. The maximum length of this field is 31 characters.

15.2.4.4 *Volume Resize Form*

Access:

Basic-Ops ➤ Volume Operations ➤ Resize

Description:

This form either grows or shrinks a volume using the Volume Manager free space management resources. If new disk space is needed, it will be allocated as necessary; if space becomes unused, it will be added to the free space pool.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless they are listed as read only.

Table 15-17 Volume Resize Form

Field	Description
Selected Volume:	This field displays the name of the volume to be resized. This field is read only and cannot be changed.
Current size:	This field displays the current size of the volume to be resized. This field is read only and cannot be changed.
Option:	The type of resize operation to be performed. This will determine whether the volume is grown or shrunk to a certain size, or grown or shrunk by a given amount. The default is Grow To.
Size / Amount:	Enter either the length to which or the amount by which the volume is to be resized. If Grow To or Shrink To is selected, this field should reflect the final size. If Grow By or Shrink By is selected, this field should reflect the amount by which the size should change. The new volume size should be less than or equal to the available free space of the disks.

15.2.4.5 Snapshot Form

Access:

Basic-Ops ➤ Volume Operations ➤ Snapshot ➤ Snapshot

Description:

This form creates a snapshot of the selected volume for backup purposes.

Fields:

Note – Fields in this form are required. Fields in this form are read / write fields, unless they are listed as read only.

Table 15-18 Snapshot Form

Field	Description
Selected Volume:	This field displays the name of the volume to be used as the snapshot source. This field is read only and cannot be changed.
Snapshot name:	The name of the snapshot volume to be created as a backup. Although a default name appears in this field, a name that more closely resembles that of the selected volume should be used for easier association. The maximum length is 14 characters. The snapshot name must be unique.

Requirements:

There must be sufficient free space to accommodate the snapshot volume.

15.2.5 Disk Operations

The Disk Operations menu provides the following selections:

- Add Disks
- Replace Disks
- Evacuate Subdisks
- Help

15.2.5.1 Add Disks

Access:

Basic-Ops ➤ Disk Operations ➤ Add Disks

Description:

This option places a physical disk under Volume Manager control. This involves initializing, analyzing, and partitioning the raw disk, initializing the disk for Volume Manager use, and adding the disk to a disk group.

15.2.5.2 Replace Disks

Access:

Basic-Ops ➤ Disk Operations ➤ Replace Disks

Description:

This option replaces a disk. This is normally done when a failed disk needs to be replaced with a new one. This involves initializing, analyzing, and partitioning the raw disk, initializing the disk for Volume Manager use, and replacing the old disk and associated disk media records with the new disk and its information.

15.2.5.3 Evacuate Subdisks

Access:

Basic-Ops ➤ Disk Operations ➤ Evacuate Subdisks

Description:

This option moves all subdisks from one disk to another. This operation can only be performed between two disks in the same disk group.

15.2.5.4 Evacuate Subdisks Form

Access:

Basic-Ops ➤ Disk Operations ➤ Evacuate Subdisks

Description:

This form gives the information for the disks that the subdisks will be moved from and to.

Table 15-19 Evacuate Subdisks Form

Field	Description
Disk group name:	The name of the disk group to which both disks belong. Both disks must share the same disk group.
Evacuate from:	This field displays the name of the disk from which the subdisks are to be evacuated, which can be changed.
To:	The name of the disk to which the subdisks are to be moved. This field is optional. However, the evacuated subdisks will be moved to one or more random disks if no target disk is specified

15.2.6 Help

Access:

Basic-Ops ➤ Help

Description:

This option gives extra information on the Basic-Ops options. The Help option is explained in greater detail in “Help Menu.”

15.3 Advanced-Ops Menus

The following menus, sub-menus, and selections are accessed via the Advanced-Ops menu:

- Volume
- Plex
- Subdisk
- Disk Group
- Disk
- Help

15.3.1 Volume

Access:

Advanced-Ops ➤ Volume

Description:

This menu provides access to assorted volume operations. These volume operations use the manual approach to volume management.

The Volume menu provides the following selections:

- Create
- Remove Volumes
- Initialize Volumes
- Start Volumes
- Stop Volumes
- Resynchronize Volumes
- Set to Maint State
- Help

15.3.1.1 Create

Access:

Advanced-Ops ➤ Volume ➤ Create

Description:

This operation creates a volume. You may select one or more plexes to be associated with the new volume after creation.

Form:

Volume Create Form (described in “Volume Create Form” on page 15-41).

15.3.1.2 Remove Volumes

Access:

Advanced-Ops ➤ Volume ➤ Remove Volumes

Description:

This operation removes the selected volume(s). If the selected volume is started, it must be stopped before it can be removed.

Note – This is a permanent operation and cannot be undone. Any plexes associated with the volume will be dissociated and left behind.

Requirements:

At least one volume icon must be selected.

The volume must be stopped before it can be removed.

15.3.1.3 Initialize Volumes

Access:

Advanced-Ops ➤ Volume ➤ Initialize Volumes

Description:

This operation initializes the selected volume(s).

From the Initialize volumes menu, you select the type of initialization from a cascading menu listing the following choices:

Table 15-20 Initialize Volumes Menu

Active	This enables the selected volume and its associated plexes, and sets the state of all associated plexes to ACTIVE.
Enable	This enables the selected volume and its associated plexes, but leave the plex states as EMPTY.
Clean	This sets the state for all associated plexes of the selected volume to CLEAN. This can be applied only under limited circumstances.
Zero	This enables the selected volume and its associated plexes, then write zeroes over the entire volume. After the operation completes, all associated plexes are set to ACTIVE, assuming that there are no I/O errors.

Requirements:

At least one volume icon must be selected.

The selected volume cannot have been previously initialized.

The selected volume should have at least one associated plex that is complete (or contiguous).

15.3.1.4 Start Volumes

Access:

Advanced-Ops ➤ Volume ➤ Start Volumes

Description:

This operation starts the selected volume(s). A volume must be started before it can be accessed.

From the Start volumes menu, a cascading menu allows you to indicate whether all volumes or just those selected should be started:

Table 15-21 Start Volumes Menu

Start	Start the selected volume, which must be startable.
Start All	Start all volumes in this disk group that can be started.

Requirements:

At least one volume icon must be selected for the Start operation. No volume icons need to be selected for the Start All operation.

A volume should be initialized before it can be started.

15.3.1.5 Stop Volumes

Access:

Advanced-Ops ➤ Volume ➤ Stop Volumes

Description:

This operation stops the selected volume(s). A volume that is stopped is inaccessible.

From the Stop volumes menu, a cascading menu allows you to indicate whether all volumes or just those selected should be stopped:

Table 15-22 Stop Volumes Menu

Stop	Stop the selected volume.
Stop All	Stop all volumes in this disk group.

Requirements:

At least one volume icon must be selected for the Stop operation. No volume icons need to be selected for the Stop All operation.

A volume must be started before it can be stopped.

A volume that is in use or contains a mounted file system cannot be stopped.

15.3.1.6 Resynchronize Volumes

Access:

Advanced-Ops ➤ Volume ➤ Resynchronize Volumes

Description:

This operation brings all plexes within the selected volume(s) up to date. Any plexes that are inconsistent are resynchronized to contain consistent data.

Depending on how current the plexes are, this operation may take some time.

Requirements:

At least one volume icon must be selected.

The selected volume(s) must be started.

15.3.1.7 Set to Maint State

Access:

Advanced-Ops ➤ Volume ➤ Set to Maint State

Description:

This operation sets the state of the selected volume(s) to a maintenance state. Refer to the `volume(1M)` manual page for information on the maintenance state.

Requirements:

At least one volume icon must be selected.

15.3.2 Volume Forms

15.3.2.1 Volume Create Form

Access:

Advanced-Ops ➤ Volume ➤ Create

Description:

This form creates a volume according to your specifications.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 15-23 Volume Create Form

Field	Description
Volume name:	The name of the volume to be created. The name must be unique within this disk group. The maximum length of this field is 31 characters. The name specified for the volume must be unique within this disk group.
Usage Type:	The desired usage type. fsgen is the file system generic usage type, which assumes that the volume is being used by a file system. gen is the generic usage type, which makes no assumptions regarding the data content of the volume. The default is fsgen. This field is optional.
User:	The name of the user who will be the owner of this volume. This must be a valid user name on the system. The maximum length of this field is 14 characters.
Group:	The name of the group that will own this volume. This must be a valid group name on the system. The maximum length of this field is 14 characters.
Mode:	The permissions mode for the new volume. Only numbers of the correct format are valid in this field. The maximum length of this field is four characters.

Table 15-23 Volume Create Form *(Continued)*

Field	Description
Length:	The length of the volume. If no unit is specified, the default is sectors. Only positive numbers greater than zero are valid. This field is optional.
Plexes:	This field displays the number of plexes associated with the volume. If no plexes were selected prior to invoking this form, this field displays 0. This field is read only and cannot be changed.
Read Policy:	The read policy that the volume adopts when deciding which plex to write to. These policies are distinguished as follows: Round Robin – All plexes are read equally, in turn. Preferred Plex – A particular plex is specified as the plex to be read whenever possible. The preferred plex will not be read in situations such as when that plex is detached due to I/O failure. Based on plex layouts – All plexes are read equally and in turn, unless a striped plex is present, in which case the striped plex becomes the preferred plex. This option is the default and it typically gives the best read performance.
Preferred Plex:	The name of the preferred plex if the Preferred Plex read policy has been specified. The string in this field must be the name of a valid plex that is associated with this volume. This field is required if Preferred Plex is specified in the Read Policy: field.
Comment:	An appropriate comment for this volume. The maximum length of the comment is 40 characters. This field is optional.
Startup:	This field may contain an arbitrary string that is reserved for you by usage-type utilities. The intention is that this field be used to store options that apply to the volume, such as for the start volumes operation. This is normally a comma-separated list of flag names and *option=value* pairs. This field is optional.
Logging:	Indicates whether logging is defined and supported on this volume. The default is Don’t Log.
Writeback:	Indicates whether the volume is to write back on read failure. If set to Yes, an attempt will be made to fix a read error from a participating plex. The default is No.

Table 15-23 Volume Create Form *(Continued)*

Field	Description
Putil0:	Permanent utility field 0. This is reserved for Volume Manager use, but may be changed. The maximum length of all Putil fields is 14 characters. This field is optional.
Putil1:	Permanent utility field 1. This field is reserved, but may be changed. This field is optional.
Putil2:	Permanent utility field 2. This field is reserved, but may be changed. This field is optional.

15.3.3 Plex

The Plex option contains the following options:

- Create
- Remove Plexes
- Associate Plexes
- Dissociate Plexes
- Attach Plexes
- Detach Plexes
- Help

15.3.3.1 Create

Access:

Advanced-Ops ➤ Plex ➤ Create

Description:

This operation creates a plex. You may select one or more subdisks to be associated with the new plex after creation.

Form:

Plex Create Form (described in “Plex Create Form” on page 15-46).

15.3.3.2 Remove Plexes

Access:

Advanced-Ops ➤ Plex ➤ Remove plexes

Description:

This operation removes the selected plex(es). This is a permanent operation and cannot be undone. Any subdisks associated with the plex will be dissociated and left behind.

Requirements:

At least one plex icon must be selected.

If the selected plex is associated with a volume, it must be dissociated before it can be removed.

15.3.3.3 Associate Plexes

Access:

Advanced-Ops ➤ Plex ➤ Associate Plexes

Description:

This operation associates one or more selected plexes with the selected volume. If the volume is started, the Volume Manager will begin to bring the plex up to date by copying all necessary data to the plex. This may take a fair amount of time.

Requirements:

A volume icon and at least one plex icon must be selected.

Only non-associated plexes can be associated.

15.3.3.4 Dissociate Plexes

Access:

Advanced-Ops ➤ Plex ➤ Dissociate Plexes

Description:

This operation dissociates one or more selected plexes from their parent volumes. This operation will fail if the plex cannot be dissociated. For example, the last plex in a started volume cannot be dissociated.

Requirements:

At least one plex icon must be selected.

Only associated plexes can be dissociated.

Before the last plex in a volume can be dissociated, that volume must be stopped.

15.3.3.5 Attach Plexes

Access:

Advanced-Ops ➤ Plex ➤ Attach Plexes

Description:

This operation attaches one or more selected plexes to their parent volumes. A plex must be detached but still associated with an enabled volume in order to be attached; the plex is actually being reattached with its parent volume.

Requirements:

At least one plex icon must be selected.

A plex must be detached before it can be attached.

Only a plex associated with an enabled volume can be attached.

15.3.3.6 Detach Plexes

Access:

Advanced-Ops ➤ Plex ➤ Detach Plexes

Description:

This operation detaches one or more selected plexes from their parent volumes. A detached plex is inaccessible for reads and writes, but is still associated with the volume.

Requirements:

At least one plex icon must be selected.

Only associated plexes can be detached.

This operation is not permitted when the specified plex is the last valid plex on the volume.

15.3.4 Plex Forms

Some plex operations result in the appearance of forms, which must be completed in order for that operation to proceed. Most forms provide a Help button, which contains information relevant to the fields and other aspects of that particular form.

15.3.4.1 Plex Create Form

Access:

Advanced-Ops ➤ Plex ➤ Create

Description:

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 15-24 Plex Create Form

Field	Description
Plex name:	The name of the plex to be created. The name must be unique within this disk group. The maximum length of this field is 31 characters.
Plex state:	The plex utility state. This is reserved for use by usage types. This field is optional.
Volume:	The name of the volume that this plex should be associated with. The name must be a valid volume name in this disk group. The maximum length of this field is 14 characters. This field is optional.
Layout:	The desired layout for the plex. A concatenated plex is a plex with associated subdisks that are both sequentially and contiguously arranged. A striped plex is a plex that scatters data evenly across each of its associated subdisks. The default is Concatenated.
Stripe width:	The width of the stripes on the plex. The stripe width must be a number greater than 0. If no units are specified, sectors are assumed. The maximum length of this field is 14 characters. If Striped plex layout has been specified, this field is required. This field must be blank if Concatenated plex layout has been specified.
Subdisks:	The number of subdisks associated with the plex. This field is read only and cannot be changed.
Comment:	An appropriate comment for the plex. The maximum length of the comment is 40 characters. This field is optional.
Errors:	Indicates whether the plex should participate in Volume Manager error policies. The default is Participate.

Table 15-24 Plex Create Form *(Continued)*

Field	Description
Putil0:	Permanent utility field 0. This is reserved for Volume Manager use, but may be changed. The maximum length of all Putil fields is 14 characters. This field is optional.
Putil1:	Permanent utility field 1. This field is reserved, but may be changed. This field is optional.
Putil2:	Permanent utility field 2. This field is reserved, but may be changed. This field is optional.

15.3.5 Subdisk

The Subdisk option contains the following options:

- Create
- Remove Subdisks
- Associate Subdisks
- Associate as Log Sd
- Dissociate Subdisks
- Join Subdisks
- Split the Subdisk
- Help

15.3.5.1 Create

Access:

Advanced-Ops ➤ Subdisk ➤ Create

Description:

This operation creates a subdisk on the selected VM disk.

Requirements:

A VM disk must be selected.

Form:

Subdisk Create Form (described in “Subdisk Create Form” on page 15-52).

15.3.5.2 Remove Subdisks

Access:

Advanced-Ops ➤ Subdisk ➤ Remove Subdisks

Description:

This operation removes the selected subdisk(s). This is a permanent operation and cannot be undone.

Requirements:

At least one subdisk icon must be selected.

If the selected subdisk is associated with a plex, it must be dissociated before it can be removed. Only free subdisks can be removed.

15.3.5.3 Associate Subdisks

Access:

Advanced-Ops ➤ Subdisk ➤ Associate Subdisks

Description:

This operation associates one or more subdisks with the selected plex.

Requirements:

A plex icon and at least one subdisk icon must be selected.

Only non-associated (free) subdisks can be associated.

15.3.5.4 Associate as Log Subdisk

Access:

Advanced-Ops ➤ Subdisk ➤ Associate as Log Sd

Description:

This operation associates the selected subdisk as a log subdisk with the selected plex. Block Change Logging is in effect. The resulting log subdisk icon has double borders to distinguish it from normal subdisks.

Requirements:

A plex icon and a subdisk icon must be selected.

Only non-associated (free) subdisks can be associated.

The selected plex cannot already have a log subdisk.

15.3.5.5 Dissociate Subdisks

Access:

Advanced-Ops ➤ Subdisk ➤ Dissociate Subdisks

Description:

This operation dissociates one or more selected subdisks from their parent plexes. Both log subdisks and normal subdisks can be dissociated.

Requirements:

At least one subdisk icon must be selected.

Only associated subdisks can be dissociated.

The last subdisk associated with a plex that is currently associated with a volume cannot be dissociated. The plex must be dissociated from its volume first.

15.3.5.6 Join Subdisks

Access:

Advanced-Ops ➤ Subdisk ➤ Join Subdisks

Description:

This operation joins the selected subdisks together to create a single subdisk. The resulting subdisk has the offset and name of the first subdisk (as arranged on the disk) and its length is the sum of the subdisk lengths.

Requirements:

At least two subdisk icons must be selected.

The subdisks must be contiguous on the disk.

If the subdisks are associated, they must all be associated with the same plex and be contiguous on that plex.

Logging subdisks and subdisks associated with striped plexes cannot be joined.

15.3.5.7 Split a Subdisk

Access:

Advanced-Ops ➤ Subdisk ➤ Split the Subdisk

Description:

This operation splits the selected subdisk into two or many parts. The resulting subdisks will occupy the same region on the disk that the previous subdisk occupied. If the subdisk is associated with a plex, the resulting subdisks will also be associated with that plex.

From the Split the Subdisk menu, a cascading menu allows you to indicate whether the subdisk is to be split into two or several parts:

Table 15-25 Split the Subdisk Menu

Into 2 Subdisks	Split the selected subdisk into two subdisks.
Into More Than 2 Subdisks	Split the selected subdisk into several subdisks.

Requirements:

Only one subdisk icon can be selected.

Logging subdisks and subdisks associated with striped plexes cannot be split.

Forms:

Subdisk Split Into 2 (described in “Subdisk Split Into 2 Form” on page 15-54)

Subdisk Split Into Many (described in “Subdisk Split Into Many Form” on page 15-55)

15.3.6 Subdisk Forms

The following forms are accessed through subdisk-related selections from the Advanced-Ops menu.

15.3.6.1 Subdisk Create Form

Access:

Advanced-Ops ➤ Subdisk ➤ Create

Description:

This form creates a subdisk according to your specifications.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 15-26 Subdisk Create Form

Field	Description
Disk name:	The name of the Volume Manager disk on which the subdisk is to be created. This field is read only and cannot be changed.
Subdisk name:	The name of the subdisk to be created. The name must be unique within this disk group. The maximum length of this field is 31 characters.
Disk offset:	The length into the disk where this subdisk should be located. If no units are specified, sectors are assumed. This offset should not place this subdisk within the bounds of another subdisk on the disk or past the end of the disk. Only valid positive numbers are allowed in this field.
Subdisk length:	The length of the subdisk to be created. If no units are specified, sectors are assumed. The length should not place this subdisk within the bounds of another subdisk on the disk or past the end of the disk. Only valid positive numbers are allowed in this field.
Plex name:	The name of the plex with which the subdisk is to be associated. This must be a valid plex that already exists in this disk group. The maximum length of this field is 31 characters. This field is optional.
Plex offset:	The offset of this subdisk into its associated plex. Only valid positive numbers are allowed in this field. This field is required only if a plex has been specified for association. If the subdisk is not to be associated with a plex, this field must be left blank.
Comment:	An appropriate comment for the subdisk. The maximum length of the comment is 40 characters. This field is optional.

Table 15-26 Subdisk Create Form *(Continued)*

Field	Description
Putil0:	Permanent utility field 0. This is reserved for Volume Manager use, but may be changed. The maximum length of all Putil fields is 14 characters. This field is optional.
Putil1:	Permanent utility field 1. This field is reserved, but may be changed. The maximum length of this field is 14 characters. This field is optional.
Putil2:	Permanent utility field 2. This field is reserved, but may be changed. The maximum length of this field is 14 characters. This field is optional.

15.3.6.2 Subdisk Split Into 2 Form

Access:

Advanced-Ops ➤ Subdisk ➤ Split the Subdisk ➤ Into 2 Subdisks

Description:

This form splits the selected subdisk into exactly two subdisks. The first subdisk retains the name of the original one; the second subdisk adopts the name and size specified in this form.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 15-27 Subdisk Split Into Two Form

Field	Description
Present size:	The size of the subdisk to be split. This field is read only and cannot be changed.
Name of new subdisk:	The name of the subdisk to be created from the original one. This must be a valid name and must be unique in this disk group.
Size of new subdisk:	The size of the subdisk to be created from the original one. This must be a valid number, greater than zero. The new subdisk size must be at least one sector less than the present subdisk size.

15.3.6.3 Subdisk Split Into Many Form

Access:

Advanced-Ops ➤ Subdisk ➤ Split the Subdisk ➤ Into More Than 2 Subdisks

Description:

This form splits the selected subdisk into several subdisks of equal sizes. The first subdisk retains the name of the original one; the additional subdisks are automatically named by the Volume Manager.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 15-28 Subdisk Split Into Many Form

Field	Description
Original subdisk:	The name of the selected subdisk. This field is read only and cannot be changed.
Present size:	The size of the subdisk to be split. The original subdisk must contain enough sectors to accommodate the desired total number of subdisks for the split. This field is read only and cannot be changed.
Number of new subdisks:	The total number of subdisks to be created by the split. There must be a sufficient number of sectors in the original subdisk to accommodate this number. This number should be at least two.

Requirements:

The number of subdisks is limited by the amount of space left in the configuration database.

15.3.7 Disk Group Menu

The Disk Group menu provides access to the following operations:

- Initialize
- Import Disk Group
- Deport Disk Group
- Add Disk
- Disconnect Disks
- Reconnect Disks
- Remove Disks
- Help

15.3.7.1 Initialize

Access:

Commands ➤ Disk Group ➤ Initialize

Description:

This operation defines a new disk group with a name specified by you. The new disk group contains one or more VM disks corresponding to the partition(s) selected by you.

Requirements:

At least physical disk must be selected.

Form:

Initialize Disk Group Form (described in "Initialize Disk Group Form" on page 15-61).

15.3.7.2 Import Disk Group

Access:

Commands ➤ Disk Group ➤ Import Disk Group

Description:

This operation imports a disk group to make that disk group available on the local machine. If the name of a deported disk group is known, this operation can be used to make that disk group accessible again.

Form:

Import Disk Group Form (described in "Import Disk Group Form" on page 15-61).

15.3.7.3 Deport Disk Group

Access:

Commands ➤ Disk Group ➤ Deport Disk Group

Description:

This operation disables access to a disk group. A deported disk group is no longer accessible and its view window disappears. Once deported, a disk group can be reimported.

Requirements:

A disk group cannot be deported if any volumes in that disk group are currently open.

Form:

Deport Disk Group Form (described in "Deport Disk Group Form" on page 15-62).

15.3.7.4 Add Disk

Access:

Commands ➤ Disk Group ➤ Add Disk

Description:

This operation adds a VM disk corresponding to the selected partition icon to a disk group. This involves creating a disk media record for the disk to be added. Partitions representing disks that already belong to disk groups cannot be added to disk groups.

Requirements:

One physical disk icon must be selected.

The selected physical disk cannot already belong to a disk group.

Only one disk can be added to a disk group at a time.

Form:

Add Disk Form (described in “Add Disk Form” on page 15-63).

15.3.7.5 Disconnect Disks

Access:

Commands ➤ Disk Group ➤ Disconnect Disks

Description:

This operation disables the selected VM disk, making it unavailable for use within its disk group. This involves dissociating the disk media record from its disk access record.

Requirements:

At least one VM disk icon must be selected.

The VM disk icon(s) must contain a disk media record at the time of selection.

15.3.7.6 Reconnect Disks

Access:

Commands ➤ Disk Group ➤ Reconnect Disks

Description:

This operation enables a VM disk that has previously been disconnected. This involves connecting the selected VM disk’s disk media record with the selected disk access record. Although the VM disk must be disconnected, it does not necessarily have to be reconnected to its former partition (disk access record).

Requirements:

One VM disk icon and one physical disk icon must be selected.

Neither the VM disk icon nor the physical disk icon can already be connected.

15.3.7.7 Remove Disks

Access:

Commands ➤ Disk Group ➤ Remove Disks

Description:

This operation removes the selected VM disk(s) from a disk group. Disks are removed from the disk group in which they reside. Any subdisks that exist on the selected disk(s) must be removed before the disk can be removed.

Requirements:

At least one VM disk icon must be selected.

Only disks associated with the specified disk group can be removed.

Disks containing any subdisks cannot be removed.

Only disks in the same disk group can be selected for removal in a single operation.

The last disk in a disk group cannot be removed. The disk group itself must be deported in order for its last disk to be removed.

15.3.7.8 Help

Access:

Commands ➤ Disk Group ➤ Help

Description:

This option gives extra information on the Disk Drive options. The Help option is explained in greater detail in “Help Option.”

15.3.8 Disk Group Forms

Some disk group operations result in the appearance of forms, which must be completed in order for that operation to proceed. Most forms provide a Help button, which contains information relevant to the fields and other aspects of that particular form.

15.3.8.1 Initialize Disk Group Form

Access:

Commands ➤ Disk Group ➤ Initialize

Description:

This form defines a new disk group consisting of selected disks.

Fields:

Table 15-29 Initialized Disk Group Form

Field	Description
Disk group:	The name of the new disk group. This must be a valid and unique name. This field is required. This is a read / write field.

15.3.8.2 Import Disk Group Form

Access:

Commands ➤ Disk Group ➤ Import Disk Group

Description:

This form makes the specified disk group available to the system.

Fields:

Table 15-30 Import Disk Group Form

Field	Description
Disk group:	The name of the disk group to be imported and made available to the system. This must be a valid and unique disk group name. This field is required. This is a read / write field.
New name:	This disk group will be renamed to the contents of this field when imported. This field does not contain a default value. If you do not fill in this field, the disk group will not be renamed.
Clear host ID:	Use this field with caution to clear the existing host ID that is recorded on all disks in the disk group. The default is `No`.

15.3.8.3 Deport Disk Group Form

Access:

Commands ➤ Disk Group ➤ Deport Disk Group

Description:

This form makes the specified disk group inaccessible to the system.

Fields:

Table 15-31 Deport Disk Group Form

Field	Description
Disk group:	The name of the disk group to be deported and made inaccessible to the system. This must be a valid disk group.
New name:	This will become the disk group's new name after it is deported. This field does not contain a default value. If you do not fill in this field, the existing name will remain the same.
New host:	This will set the disk group's host as part of the deport operation. This field does not contain a default value. If you do not fill in this field, the host name will remain unchanged. This field, along with the `New name` field, are useful for fixing a `rootdg` on another machine that shares disks with the current host.

Requirements:

The root disk group (`rootdg`) cannot be deported.

15.3.8.4 Add Disk Form

Access:

Commands ➤ Disk Group ➤ Add Disk

Description:

This form adds a VM disk to a disk group.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 15-32 Add Disk Group Form

Field	Description
Disk group:	The name of the disk group to which the VM disk is to be added. This must be a valid disk group. This field is required.
Disk media name:	The name of the VM disk to be created. The disk media name must be unique. By default, a unique name is generated. If this field is left blank, then the disk access name is used.

15.3.9 Disk

The Disk menu provides the following selections:

- Initialize
- Define
- Remove
- Online
- Offline
- Help

15.3.9.1 Initialize

Access:

Advanced-Ops ➤ Disk ➤ Initialize

Description:

This operation identifies a disk to the Volume Manager and initializes the disk for Volume Manager use. This involves installing a disk header and writing an empty configuration on the disk. A disk access record is created for the disk, unless such a record already exists.

Requirements

The disk should not already be initialized.

Form:

Disk Init Form (described in "Disk Init Form" on page 15-67).

15.3.9.2 Define

Access:

Advanced-Ops ➤ Disk ➤ Define

Description:

This operation defines a disk access record, which enables the Volume Manager to scan the disk. This makes the disk accessible, but does not initialize the disk.

Form:

Define Disk Form (described in "Define Disk Form" on page 15-68).

15.3.9.3 Remove

Access:

Advanced-Ops ➤ Disk ➤ Remove

Description:

This operation removes the VM disk associated with the selected partition(s) from Volume Manager control by removing the associated disk access records. If all partitions on a given disk are selected for removal at once, the disk is effectively removed from Volume Manager control.

Requirements:

At least one partition icon corresponding to a VM disk must be selected.

The VM disks corresponding to the selected partition(s) cannot belong to a disk group at the time of removal.

15.3.9.4 Online

Access:

Advanced-Ops ➤ Disk ➤ Online

Description:

This operation places the disk access record on a specified partition in an online state. During searches for disk IDs or members of a disk group, online disks are checked.

Form:

Disk Online Form (described in “Disk Online Form” on page 15-69).

15.3.9.5 Offline

Access:

Advanced-Ops ➤ Disk ➤ Offline

Description:

This operation places the disk access record on the selected partition(s) in an offline state. During searches for disk IDs or members of a disk group, offline disks are ignored.

Requirements:

At least one partition icon must be selected.

The disks corresponding to the selected partitions must be initialized.

The selected partition icon cannot be in use (shaded and associated with a VM disk).

15.3.10 Disk Forms

Some disk operations result in the appearance of forms, which must be completed in order for that operation to proceed. Most forms provide a Help button, which contains information relevant to the fields and other aspects of that form.

15.3.10.1 Disk Init Form

Access:

Advanced-Ops ➤ Disk ➤ Initialize

Description:

This form initializes a disk for Volume Manager use.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read/write fields, unless listed as read only.

Table 15-33 Disk Init Form

Field	**Description**
Public Device:	The pathname of the device node that represents a partition available for use. This name must be a valid entry in /dev/rdsk/. A name of the form c#t#d#s# is appropriate.
Device Type:	The desired disk type. The simple type (default) assumes that the public and private regions are stored on the same disk partition, with the public region following the private region. The sliced type assumes that the public and private regions are stored on different disk partitions. The nopriv type has no private region and log and configuration copies cannot be written to the disk.
Public length (0 for whole disk):	The length of the public section of the disk. If zero is provided as the length, the Volume Manager computes a default value from available partition table information. This length must be valid and cannot exceed the length of the disk.

Table 15-33 Disk Init Form *(Continued)*

Field	Description
Private Length:	The length of the private region of the disk. When one is not specified, the Volume Manager chooses a default value. This length must be valid and cannot exceed the length of the disk. This field is optional.
Number of config copies:	The number of configuration copies to be stored in the private section of this disk. The default value is two copies.
Comment:	A comment appropriate for this Volume Manager disk. The maximum length of the comment is 40 characters. This field is optional.

15.3.10.2 Define Disk Form

Access:

Advanced-Ops ➤ Disk ➤ Define

Description:

This form defines a disk.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 15-34 Define Disk Form

Field	**Description**
Public Device:	The pathname of the device node that represents a partition available for use. This name must be a valid entry in /dev/rdsk/. A name of the form c#t#d#s# is appropriate.
Device Type:	The desired disk type. The simple type (default) assumes that the public and private regions are stored on the same disk partition, with the public region following the private region. The sliced type assumes that the public and private regions are stored on different disk partitions. The nopriv type has no private region and log and configuration copies cannot be written to the disk.
Public Length (0 for whole disk):	The length of the public section of the disk. If zero is provided as the length, the Volume Manager computes a default value from available partition table information. This length must be valid and cannot exceed the length of the disk.
Offline:	Indicates whether to initially place the disk in the offline state. The default is No.
Comment:	A comment appropriate for this Volume Manager disk. The maximum length of the comment is 40 characters. This field is optional.

15.3.10.3 Disk Online Form

Access:

Advanced-Ops ➤ Disk ➤ Online

Description:

This form brings a disk online.

Table 15-35 Disk Online Form

Field	Description
Device name:	The disk access name of the disk to be brought online. This must be a valid disk access name. This field is required.

15.3.10.4 *Free Space Form*

Access:

Click RIGHT mouse button on a gap between subdisk icons in a VM disk icon.

Description:

This form provides information about a specific region of a Volume Manager disk that contains free space.

Free space results when subdisks are removed for some reason, making the space that they occupied available for use. Free space is visually represented as a gap or “hole” between subdisks that reside on a VM disk icon.

Fields:

Note – All fields in this form are read only and cannot be changed.

Table 15-36 Free Space Form

Field	Description
Device:	The name of the VM disk where this free space resides.
Hole offset:	The offset into the Volume Manager disk where this free space extent begins.
Hole size:	The size of this free space extent. The units used are specified by you under the Options pull down menu.

15.4 Analyze

The following selections are accessed via the Analyze menu:

- Start
- Stop
- Analyze All
- Stop All
- Parameters
- Help

15.4.1 Start

Access:

Analyze ➤ Start

Description:

This selection begins analysis of the selected icon(s). These icons are added to the list of objects being analyzed. Only volume and VM disk icons can be analyzed. Once analysis is activated, the selected icons begin to display information about their performance characteristics.

Requirements:

At least one volume or VM disk icon must be selected.

15.4.2 Stop

Access:

Analyze ➤ Stop

Description:

This selection terminates analysis of the selected icon(s). These icons are removed from the list of objects being analyzed. When analysis stops, the selected icons return to their pre-analysis states. When analysis is stopped for one icon, other icons undergoing analysis are not affected.

Requirements:

At least one volume or VM disk icon must be selected.

The selected icon(s) must be undergoing analysis.

15.4.3 Analyze All

Access:

Analyze ➤ Analyze All

Description:

This selection begins analysis on all volumes and VM disk icons in all view windows. Once analysis is activated, all icons begin to display information about their performance characteristics using color or bitmap patterns.

15.4.4 Stop All

Access:

Analyze ➤ Stop All

Description:

This selection automatically terminates analysis of all icon(s) in all views. All icons return to their pre-analysis states.

Requirements:

Analysis must be in effect.

15.4.5 Parameters

Access:

Analyze ➤ Parameters

Description:

This selection accesses the Analysis Parameters form, which sets user preferences for how analysis is to be conducted.

Form:

Analysis Parameters Form (described in “Analysis Parameters Form” on page 15-73).

15.4.6 Analysis Forms

The following forms are accessed through the Analyze menu.

15.4.6.1 Analysis Parameters Form

Access:

Analyze ➤ Parameters

Description:

This form sets user preferences for conducting analysis.

Fields:

Note – Most fields in this form are required; those that are optional are listed here as such. Fields in this form are read / write fields, unless listed as read only.

Table 15-37 Analysis Parameters Form

Field	Description
Sample Rate:	Determines the time interval between data samples. This field is divided into two sections: the slider bar selects the interval (1-60) and the menu to the right selects units of time (seconds or minutes). The default is 5 seconds. A shorter interval means the data will be updated more often, but is also a higher load on the system.
Volume Parameters:	Specifies the high and low values that decide the coloring (or pattern) of the volume icons.
Disk Parameters:	Specifies the high and low values that decide the coloring (or pattern) of the VM disk icons.
Subdisk Parameters:	Specifies the high and low values that decide the coloring (or pattern) of the subdisk icons.
Log File:	The name of the file to be used for the statistics log. If the file does not already exist, it will be created. The filename is taken to be relative unless a pathname is given. To stop logging to the file, the filename text in this field must be erased. This field is optional. This log file is a binary file. In order to view the log file, /opt/vxva/bin/vxvalog2text *filename* must be run on this file to process it for viewing.

Requirements:

For each set of high / low parameters, the high parameter must be greater than the low parameter.

You must have access to the specified log file.

15.4.6.2 Analysis Statistics Form

Access:

Click RIGHT mouse button on desired icon that is being analyzed.

Description:

This form displays analysis statistics relevant to the selected volume or VM disk icon. This form applies only to volume or disk icons that are undergoing analysis.

Fields:

Note – All fields in this form are read only and cannot be changed.

Table 15-38 Analysis Statistics Form

Field	**Description**
Reads:	The number of times the object was read from during the last interval.
Writes:	The number of times the object was written to during the last interval.
Total R/W:	The total number of reads and writes during the last interval.
Blocks Read:	The number of disk blocks read from the object during the last interval.
Blocks Written:	The number of disk blocks written to the object during the last interval.
Total Blocks:	The total number of blocks read from or written to the object during the last interval.

Table 15-38 Analysis Statistics Form

Field	Description
Avg Read Time:	The average time, in milliseconds, that it took for a read operation to complete. This is equal to the number of number of reads during the last interval divided by the total time spent on reads.
Avg Write Time:	The average time, in milliseconds, that it took for a write operation to complete. This is equal to the number of writes during the last interval divided by the total time spent on writes.
Interval:	The actual time, in seconds, since the last data was sampled. This may vary slightly from the specified interval time due to uncontrollable variances from system to system.

Requirements:

The icon selected with the RIGHT button must be undergoing analysis.

15.4.7 Analysis Table

The following table summarizes the default colors and patterns associated with the various levels of analysis.

Table 15-39 Analysis Table

Analysis Level	Color	Bitmap Pattern
low	green	cross_weave
medium	yellow	root_weave
high	red	wide_weave

These default colors and patterns can be changed through the GUI X resources. See Appendix B, "GUI-Related X Resources," for information on these resources.

15.5 Projection

The following menus, sub-menus, and selections are accessed via the Projection menu:

- Icon Projection
- Show Free Subdisks
- Help

15.5.1 Icon Projection

Access:

Projection ➤ Icon Projection

Description:

This menu provides access to projection options used to start or stop projection for icons.

15.5.1.1 Start

Access:

Projection ➤ Icon Projection ➤ Start

Description:

This option starts projection for the selected icon(s). When projection is started, all icons related to the selected icon(s) are highlighted. Highlighting occurs for related icons in any view windows. If the selected icon has no associated objects, Volume Manager issues a warning to this effect.

Requirements:

At least one icon must be selected.

Physical disk and partition icons cannot be selected for projection.

The selected icon(s) must be associated with at least one other icon in order for projection to take effect.

15.5.1.2 Stop

Access:

Projection ➤ Icon Projection ➤ Stop

Description:

This option stops projection for the selected icon(s). When projection is stopped, all icons related to the selected icon(s) lose their projection highlighting.

Requirements:

At least one icon must be selected. If the selected icon is not undergoing projection, Volume Manager ignores the stop request.

15.5.1.3 Stop All

Access:

Projection ➤ Icon Projection ➤ Stop All

Description:

This option stops projection for all icons that are currently undergoing selection.

15.5.2 Show Free Subdisks

Access:

Projection ➤ Show Free Subdisks

Description:

This selection determines whether free subdisks should be highlighted or not. When Show Free Subdisks is turned on, Volume Manager highlights all unassociated subdisks (representing unallocated disk space). Once turned on, any future free subdisks are automatically highlighted. Free subdisk icons can be used by designating them to objects, but Volume Manager cannot automatically use free subdisks as free space. Free subdisk projection is either started or stopped across all Volume Manager views. The start or stop preference is also retained for a particular user in future Volume Manager sessions.

From the Show Free Subdisks menu, a cascading menu allows you to indicate whether or not to highlight free subdisks:

Table 15-40 Show Free Subdisks Menu

Start	Start highlighting free subdisks immediately and continue to do so until instructed to stop.
Stop	Stop highlighting free subdisks.

15.5.3 Projection Table

The following table summarizes the projection relationships that are highlighted for particular icon types. If no icons of the correct type are associated with the selected icon, then nothing is highlighted.

Table 15-41 Projection Table

Icon Selected	**Icons Highlighted**
Volume	All subdisks associated with any plex associated with the volume
Plex	All subdisks associated with the plex
Subdisk	Associated plex and volume, and all other subdisks associated with the plex
VM Disk	All plexes associated with the subdisks that reside on the disk

15.6 Options Menu

The Options menu allows you to set preferences for how the GUI should operate. Once set, most preferences are saved to the file `.vxva_pref`, located in your home directory. You should not edit the `.vxva_pref` file. Once set, these preferences are maintained across future GUI runs.

The Options menu contains the following menu items:

- Show Command
- When Commands are Ready
- Logging
- Popup the Command Window
- Format of Size
- Help

15.6.1 Show Command

Access:

Options ➤ Show Command

Description:

This selection specifies whether the Command Info Window is to be displayed before each command is executed or only upon command failure.

From the Show Command menu, a cascading menu allows you to indicate when to display the Command Info Window:

Table 15-42 Show Command Menu

Show on Error	Display the Command Info Window only when a Volume Manager or system command has failed. This is the default behavior.
Show at Start	Display the Command Info Window whenever a Volume Manager or system command is ready to be sent. This enables you to view the actual command(s) being executed via GUI.

15.6.2 When Commands Are Ready

Access:

Options ➤ When Commands Are Ready

Description:

This selection specifies whether commands should be executed immediately or simply displayed for user review.

From the When Commands Are Ready menu, a cascading menu allows you to indicate what the GUI is to do when a command is issued:

Table 15-43 When Commands Are Ready Menu

Execute Commands	Automatically execute commands as soon as they are issued. This is the default.
Show Commands Only	Display commands in the Command Info Window for user review rather than executing them. Upon approval, you can execute the displayed command directly from the Command Info Window by highlighting that command and then using the Execute menu.

15.6.3 Logging

Access:

Options ➤ Logging

Description:

This selection starts or stops logging of GUI commands. Logging records all commands sent to the Volume Manager or the system by the GUI in a specified file.

From the Logging menu, a cascading menu allows you to indicate whether logging should be activated or deactivated:

Table 15-44 Logging Menu

Start	Begin recording all commands to a log file. A log file is created if one does not already exist. The file to be used for logging must be specified in the resulting form. If the file exists, you must have permission to write to that file. The log information will be appended to the end of the specified file.
Stop	Stop recording all commands to the log file. When logging is discontinued, you are responsible for remembering the name of the log file that was used.

Note – Unlike other user preferences, the logging setting is not saved across GUI sessions.

Form:

Log File Form (described in "Log File Form" on page 15-83).

15.6.4 Popup the Command Window

Access:

Options ➤ Popup the Command Window

Description:

This selection accesses and displays the Command Info Window. This window displays current and previous commands, along with the status of each command. Once accessed in this way, the Command Info Window remains visible until it is closed via its File menu.

See the "Command Info Window" section for further details.

15.6.5 Format of Size

Access:

Options ➤ Format of Size

Description:

This selection specifies the units (megabytes, kilobytes, or sectors) to be used for size-related output. The unit of size is set to megabytes until you reset it.

In properties forms, length values are displayed as sectors (s), kilobytes (k), or megabytes (m). If the size cannot be cleanly converted into kilobytes or megabytes, it is displayed in sectors instead (even though another format of size preference may be set).

The preferred format of size applies to output only and does not impact input in any way. Input typically defaults to sectors, unless megabytes or kilobytes are specified.

From the Format of Size menu, a cascading menu allows you to select the units of size.

15.6.6 Help

Access:

Options ➤ Help

Description:

This option gives helpful information on using the Options menus.

15.6.7 Options Forms

The following forms are accessed via the Options menu.

15.6.7.1 Log File Form

Access:

Options ➤ Logging ➤ Start

Description:

This form specifies the file to be used for logging purposes.

Fields:

Table 15-45 Log File Form

Field	Description
Log File:	The name of the file (and path name) to be used to store the command log. If no path is specified, the file is created in the directory from which the GUI session was started. The maximum length of the path specified here is 127 characters.

Requirements:

You must have appropriate permissions to access and write to the named file (and any directories in its path).

If a path name is included, it must be valid.

15.7 Icon Menu

A set of icon-related options can be accessed via the Icon menu located in views.

15.7.1 Maximize Icons

Access:

Icon ➤ Maximize Icons

Description:

This operation restores the selected minimized icons to their full size, making them show all of their sub-icons.

Requirements:

At least one minimized icon must be selected.

15.7.2 Minimize Icons

Access:

Icon ➤ Minimize Icons

Description:

This operation reduces the size of the selected icons, making them shrink down in size and hide all of their sub-icons.

Minimized icons occupy less space and are displayed with their names in reverse type.

Requirements:

At least one maximized icon must be selected.

15.7.3 Maximize All Icons

Access:

Icon ➤ Maximize All Icons

Description:

This operation restores all icons in the current view window to their full size, making them show all of their sub-icons. No icons need to be selected.

15.7.4 Minimize All Icons

Access:

Icon ➤ Minimize All Icons

Description:

This operation reduces the size of all icons in the current view window, making them hide all of their sub-icons. No icons need to be selected.

15.7.5 Create Icons

Access:

Icon ➤ Create Icons

Description:

This operation creates a copy of the icon(s) selected from another view and places the new copy in the current user-created view. Icons that already exist in this user-created view will not be duplicated.

Requirements:

This option is only available in user-created views.

15.7.6 Remove Icons

Access:

Icon ➤ Remove Icons

Description:

This operation removes the selected icon(s) from the current user-created view.

Requirements:

This option is only available in user-created views.

Part 5— Tutorial

Tutorial for the Graphical User Interface 16

This chapter is the tutorial for using the graphical user interface for the SPARCstorage Array. By following the procedures given in this chapter, you will learn how to:

1. Bring up the graphical user interface (GUI) in the tutorial mode.
2. Bring a non-SPARCstorage Array disk under Volume Manager control by creating a disk group and adding the disk to that disk group.
3. Bring two physical disks from your SPARCstorage Array under Volume Manager control.
4. Set up the data structure so that data is striped across all three disks.

The tutorial mode is useful if you want to get used to the GUI and understand the components of the Volume Manager without making any changes to your system's setup.

16.1 Bringing Up the GUI

In order to run the GUI from the command line, your PATH environment variable needs to contain /opt/vxva/bin. If necessary, update PATH to include /opt/vxva/bin before attempting to run the GUI.

- For a Bourne Shell (sh or ksh), update PATH to reflect /opt/vxva/bin:

```
PATH=${PATH}:/opt/vxva/bin
```

- For a C Shell (csh or tcsh), update PATH to reflect /opt/vxva/bin:

```
setenv PATH ${PATH}:/opt/vxva/bin
```

1. Enter the following command to run the GUI from the tutorial mode:

```
vxva -t &
```

If this is the first time you've brought up the tutorial, you may see this error message:

Figure 16-1 Tutorial Error Message

If you see this error message, enter the following command at the prompt:

```
/opt/vxva/bin/vxva_setup
```

Then enter vxva -t & once more to run the GUI from the tutorial mode.

When you run the GUI from the tutorial mode, you will be able to go through the process of setting up your system for certain disk management tasks, but none of these changes will actually be applied to your system.

You should see the root window pop up after you enter the command (see Figure 16-2).

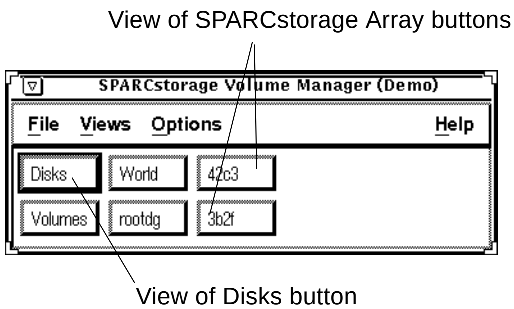

Figure 16-2 Root Window

16.2 Bringing a Non-SPARCstorage Array Disk Under Volume Manager Control

1. **Click LEFT on the Disks screen button.**
   This will bring up the View of Disks window (see Figure 16-3). In normal mode, this view will show you all the non-SPARCstorage Array disks that you have connected to your system.

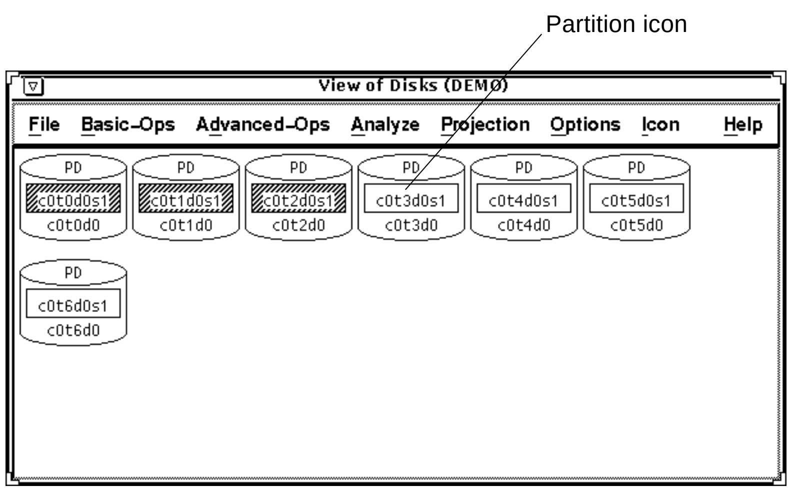

Figure 16-3 View of Disks

2. **Click LEFT on a partition icon in one of the physical disk icons in the View of Disks window (see Figure 16-3).**
   The partition icon should change its color or pattern, indicating that it has been selected.

3. **From the Advanced-Ops menu, select Disk Group, then Initialize (see Figure 16-4).**

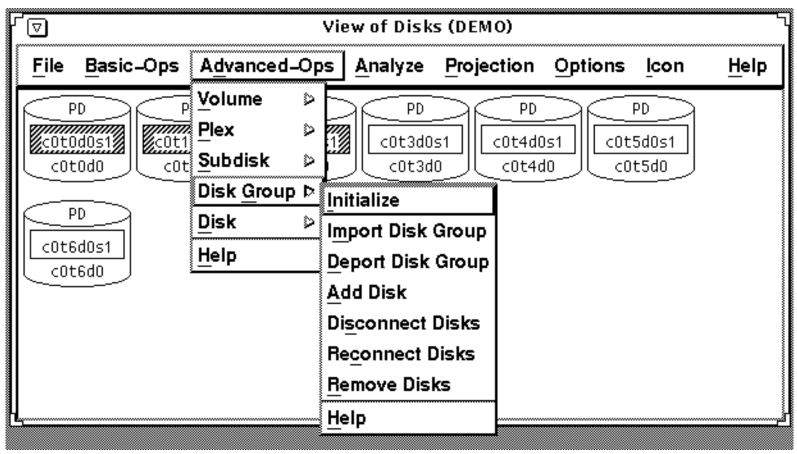

Figure 16-4 Initializing a Disk Group

This will bring up the Disk Group Initialize Form (see Figure 16-5).

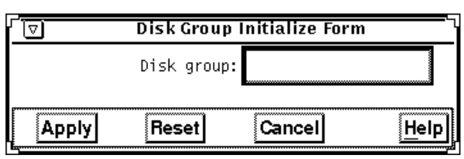

Figure 16-5 Disk Group Initialize Form

4. **Enter a name for the disk group that you want to create and click LEFT over the Apply screen button.**
   This will create the new disk group and add the physical disk to that disk group, thereby bringing it under Volume Manager control. If you look at the root window, you will see the new disk group displayed as a screen button.

5. **Go to the root window and click LEFT over the screen button for the disk group you just created (see Figure 16-2).**
   This will bring up the view of the disk group. You should see a single VM disk in your disk group. This VM disk is assigned to the partition you selected from the View of Disks window.

16.3 Bringing SPARCstorage Array Disks Under Volume Manager Control

1. **Locate the screen button in the root window for the SPARCstorage Array that you want to access (see Figure 16-2).**
   Every SPARCstorage Array has a unique world-wide name; the four characters shown in the SPARCstorage Array screen buttons are the last four characters of that name.

2. **Click LEFT over the four-character screen button in the root window for the SPARCstorage Array you want to access.**
   This will bring up the View of SPARCstorage Array (see Figure 16-6).

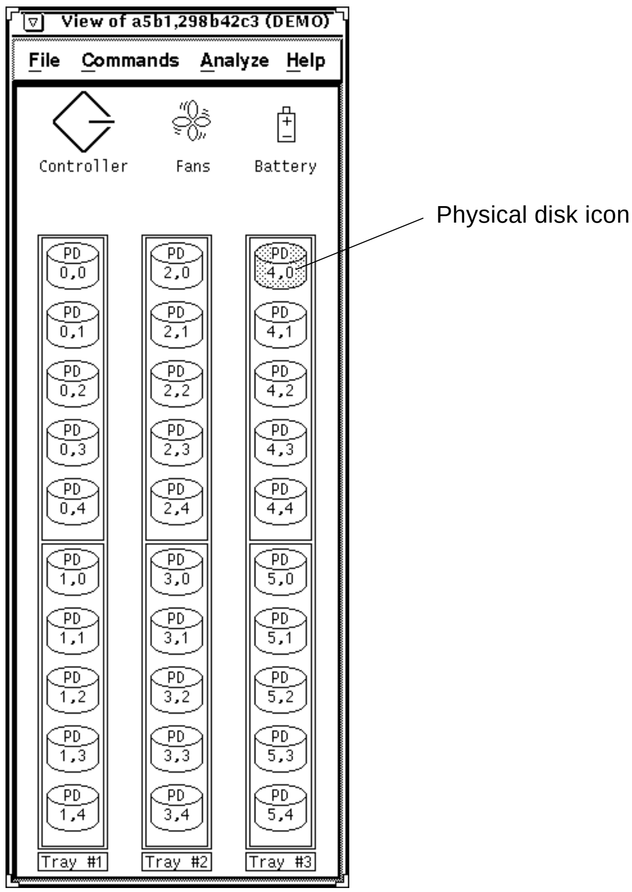

Figure 16-6 View of SPARCstorage Array

3. **From the View of SPARCstorage Array window, click LEFT over a physical disk icon (see Figure 16-6).**
   The icon will change its color or pattern, indicating that it has been selected.

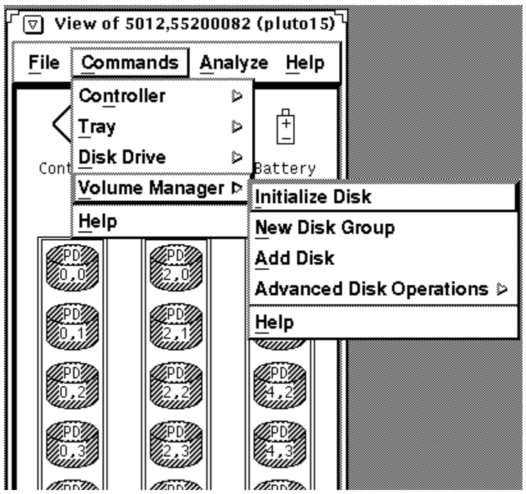

Figure 16-7 Accessing the Volume Manager Menu

4. **From the Commands menu, select Volume Manager, and Initialize Disk (see Figure 16-7).**
   This will bring up the Initialize Disk warning box:

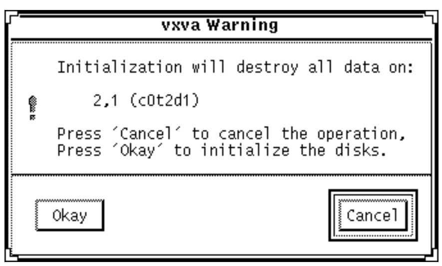

Figure 16-8 Initialize Disk Warning Box

   Click LEFT over the Okay button.

5. **Drag the physical disk icon from the View of SPARCstorage Array window into the view of your disk group and drop the physical disk icon anywhere within that window.**
   A second VM disk icon will now appear in the view of your disk group. This VM disk is assigned to the physical disk you selected from the SPARCstorage Array window.

6. **Repeat Step 3 through Step 5 for another disk in the View of SPARCstorage Array window.**
   You should now see three VM disks in the view of your disk group.

16.4 Setting Up the Disks for Striping

1. **Go to the view of your disk group and click MIDDLE over all three VM disk icons.**

2. **From the Basic-Ops menu, select Volume Operations, then Create (see Figure 16-9).**
   A submenu listing basic volume types appears.

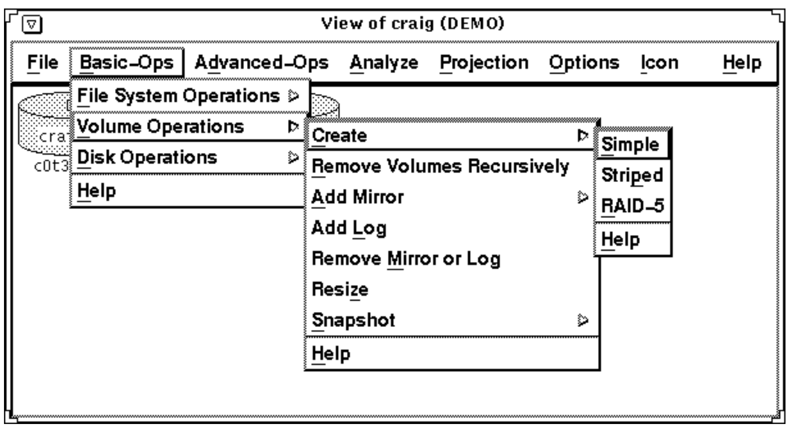

Figure 16-9 Accessing the Volume Operations Create Menu

3. **Select Striped.**
   The Striped Volume/FS Create form appears (see Figure 16-10). This form creates a striped volume. The form is divided into two sections, one for volume creation and the other for file system creation. Most of the form fields are already set to the defaults for the creation of a new volume; the file system fields are grayed out because the default is not to add a file system to the volume. Since you are running the graphical user interface from the demo mode, you will not be able to access the file system portion of the form.

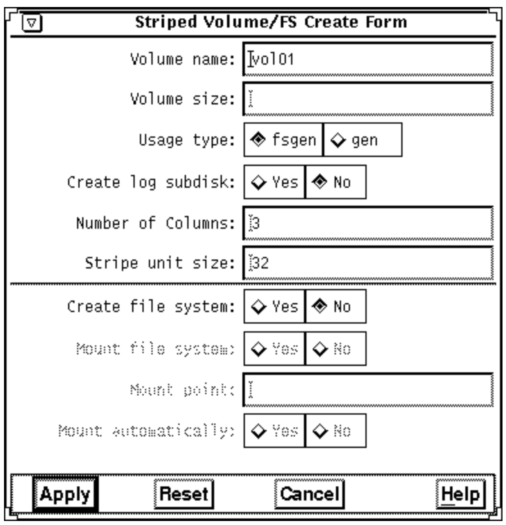

Figure 16-10 Striped Volume / FS Create Form

4. Complete the Striped Volume/FS Create form:

a. At the `Volume name:` **field, either keep the default name or enter a different one.**

The name must be unique within this disk group. The maximum length of this field is 14 characters.

b. **At the** `Volume size:` **field, enter the desired volume size.**
The size should be entered as a number followed immediately by the letter `k, m,` or `s` to indicate kilobytes, megabytes, or sectors, respectively. If no unit is specified, the default is sectors. For a striped volume, the volume size should be calculated as follows: *vol_size* = *stripe_width* * *number_of_stripes* * *n*, where *n* is an integer greater than zero.

For the purpose of this exercise, let's assume that you want to set aside 100 megabytes from each disk for striping. Using the formula given above, you would enter *300m* in the `Volume size:` field (100 * 3 * 1).

c. **At the** `Usage type:` **field, select a usage type.**
The default, `fsgen`, is already selected. `fsgen` is the file system generic usage type, which assumes that the volume is being used by a file system. `gen` is the generic usage type, which makes no assumptions regarding the data content of the volume.

d. **At the** `Create log subdisk:` **field, select No.**
If you want to learn about log subdisks, see Section 1.2.5, "Dirty Region Logging," on page 1-31.

e. **At the** `Number of Columns:` **field, enter the number of columns that the volume's plex is to have.**
This is the number of disks that you want to have data striped across. Since you already selected three VM disks to have data striped across, the number 3 should already appear in this field.

f. **At the** `Stripe unit size:` **field, enter the unit size of the stripe.**
This is the size of the stripes on the volume's plex. The value specified may be optimized for the particular drive configuration, as best striping performance is achieved when the stripe width corresponds to the track width of the drive. The default value for this field is 32 sectors, chosen as a good stripe width for most systems.

5. **When the form is properly completed, select Apply to activate the volume and file system creation.**
A new volume icon appears. Since this is a striped volume, it contains a single plex and three subdisks. Figure 16-11 shows how the view of your disk group should look with all three VM disks (now with one subdisk apiece) and a striped volume.

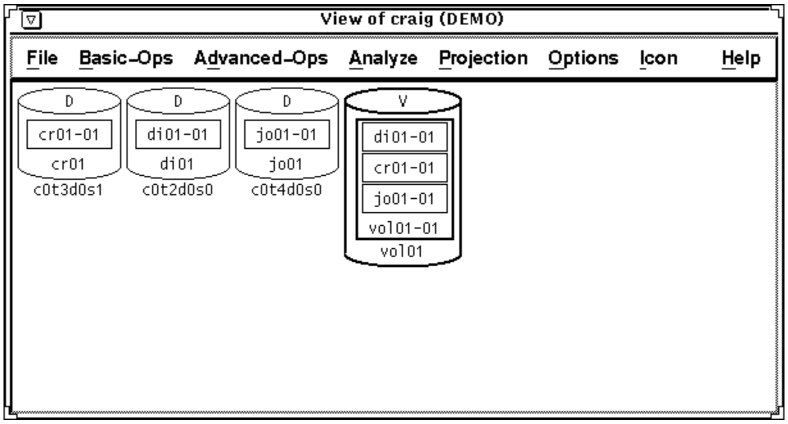

Figure 16-11 Sample Disk Group View with Striped Volume

Using the Volume Manager Software on Non-SPARCstorage Array Disks A

There are several software tasks you can perform on non-SPARCstorage Array disks using the Volume Manager. On non-SPARCstorage Array disks, you can create any type of volumes except striped. This appendix explains the tasks that are different for non-SPARCstorage Array disks.

A.1 Identifying a Physical Disk to the Volume Manager

Before a disk can be used within the Volume Manager, it must be identified to the Volume Manager. To be used as a VM disk, it must also be added to a disk group.

The GUI allows you to identify a disk to the Volume Manager in one of two ways:

- Using the basic approach, which also partitions the disk and adds it to a disk group.
- Using the advanced approach, which does not automatically partition the disk or add it to a disk group.

Basic Approach

The following underlying steps are handled automatically when a disk is placed under Volume Manager control via the GUI's Basic-Ops menu:

- initializing and partitioning the raw disk
- initializing the new disk for Volume Manager use (installing header and configuration information on the disk)
- adding the disk to a disk group

Note – This operation alters the partitioning of a disk and should therefore be used with caution.

To make a physical disk available to the Volume Manager, follow these steps:

1. **Go to the View of Disks window.**
2. **From the Basic-Ops menu, select Disk Operations, then Add Disks.**
   The Add Disks form appears.
3. **Complete the Add Disks Form.**
4. **When the form is properly completed, select Apply to activate the disk add operation.**

After a confirmation window pops up, a new physical disk icon (with the name supplied as the new disk name) appears. The new physical disk icon contains a partition icon. The partition icons of physical disks under Volume

Manager control are patterned or colored to differentiate them from others. If a disk group was specified, a new VM disk icon appears in the view window corresponding to that disk group.

Advanced Approach

The Advanced-Ops menu provides an option to initialize a new disk for Volume Manager use (installing header and configuration information on the disk). If the disk is to be partitioned and/or added to a disk group, these tasks must be performed separately afterwards.

To make a physical disk available to the Volume Manager, follow these steps:

1. **Go to the View of Disks window.**

2. **From the Advanced-Ops menu, select Disk, then Initialize.**
   The Disk Init form appears. This form is used to initialize regions of a disk used by the Volume Manager.

3. **Complete the Disk Init Form.**

4. **When the form is properly completed, select Apply to activate the disk initialization.**
   A new physical disk icon containing a partition icon appears.

To assign the disk to a disk group, follow the instructions in Section A.3, "Adding a Disk Drive to a Disk Group."

A.2 Creating a Disk Group

1. **From the root window, click LEFT over the Disks screen button (see Figure A-1).**

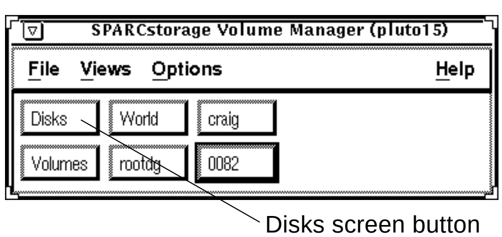

Figure A-1 Root Window

2. **In the View of Disks, from the Advanced-Ops menu, select Disk Group, then Initialize (see Figure A-2).**

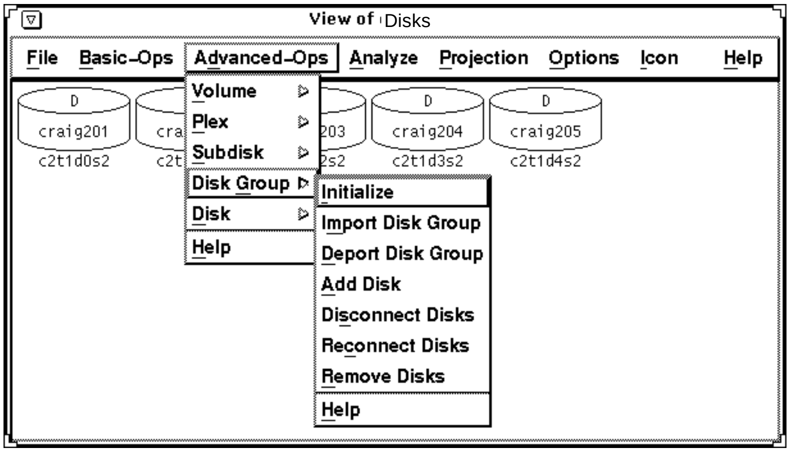

Figure A-2 Accessing the Disk Group Menu

This will bring up the Disk Group Initialize Form (see Figure A-3).

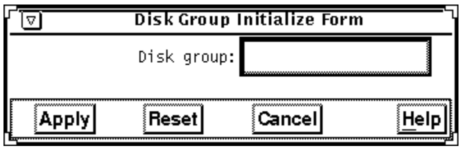

Figure A-3 Disk Group Initialize Form

3. **Enter a name for the disk group that you want to create and click LEFT over the Apply screen button.**
   The new disk group will now appear in your root window.

A.3 Adding a Disk Drive to a Disk Group

1. **Access the View of Disks from the root window and click LEFT over the physical disk icon.**

2. **In the View of Disks, from the Advanced-Ops menu, select Disk Group, then Add Disk (see Figure A-2).**
   This will bring up the Add Disk Form (see Figure A-4).

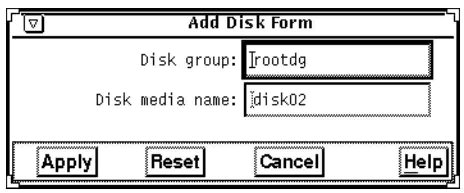

Figure A-4 Add Disk Form

3. **Complete the Add Disk Form and click LEFT over the Apply screen button.**
   The VM disk will now appear in the view of the disk group.

A.4 Reconnecting a VM Disk to a Disk Group

This operation will reconnect a VM disk that was previously disconnected from a physical disk's partition. Note that the VM disk does not have to be reconnected to the original physical disk partition.

1. **Click LEFT over the screen button in the root window for the view you want to access.**

2. **Click LEFT on the physical disk icon that you want to bring reconnect to the disk group.**

3. **From the Advanced-Ops menu, select Disk Group, then Reconnect Disks (see Figure A-5).**

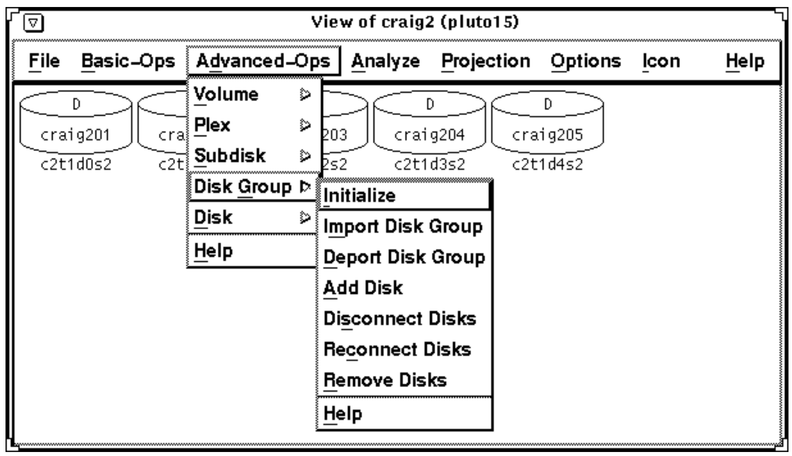

Figure A-5 Accessing the Disk Group Menu

The disk will now reconnect to the disk group.

A.5 Removing a VM Disk From a Disk Group

You can remove a VM disk from a disk group when it is no longer necessary to access the physical disk that the VM disk is assigned to or when that physical disk is about to be removed from Volume Manager control altogether. A physical disk cannot be removed until its VM disk has been removed from its disk group. Any subdisk residing on the VM disk that is assigned to that physical disk must be removed or moved to another VM disk before the VM disk can be removed from its disk group.

Note – The last VM disk in a disk group cannot be removed. In order for the last VM disk in a disk group to be removed, the disk group itself must be deported using the procedures in Section A.9, "Deporting a Disk Group."

---

1. **Go to the view of the VM disk to be removed.**
2. **Select that VM disk by clicking the LEFT button on its icon.**
3. **From the Advanced-Ops menu, select Disk Group, then Remove Disks.**
   The VM disk icon disappears from the view window, indicating that the VM disk no longer belongs to this disk group.

A.6 Removing a Physical Disk from Volume Manager Control

1. **Go to the view window corresponding to the disk group in which you want to perform this operation.**
2. **Click LEFT over the VM disk icon that you want to remove from Volume Manager control.**
3. **From the Advanced-Ops menu, select Disk, then Remove.**

A.7 Disconnecting a VM Disk from a Disk Group

This operation will disconnect the selected VM disk from its associated partition, making it unavailable for use within its disk group. This is slightly different from removing a VM disk from a disk group in that the VM disk icon remains in the disk group instead of removing it from the group altogether, so you can reconnect it the same partition or a different one at a later date.

1. **Click LEFT over the screen button in the root window for the view you want to access (see Figure A-1).**
2. **Click LEFT over the VM disk that you want to disconnect from the disk group.**
3. **From the Advanced-Ops menu, select Disk Group, then Disconnect Disks (see Figure A-2).**
   The VM disk will disconnect from the disk group.

A.8 Importing a Disk Group

Importing a disk group enables access to a disk group that has been deported. To import a deported disk group, its former name must be known and at least one partition (containing a disk access record) formerly assigned to the deported disk group must remain unused. If all disks associated with a deported disk group have since been reused, that disk group cannot be imported.

1. **Go to any view window.**
2. **From the Advanced-Ops menu, select Disk Group, then Import Disk Group.**
   The Import Disk Group form appears.

3. **Complete the Import Disk Group form, entering the name of the disk group to be imported.**
   The named disk group must have been deported at one time. At least one partition formerly assigned to the deported disk group must still exist (unused) in order for the disk group to be importable.

4. **When the form is properly completed, select Apply to activate the disk import.**
   The view button corresponding to the disk group that has been imported reappears in the Visual Administrator root window. The partition icon(s) corresponding to the VM disk icon(s) that formerly belonged to that disk group and has not been reused becomes shaded. When the view of the newly-imported disk group is accessed, its VM disk icon(s) is visible.

A.9 Deporting a Disk Group

Deporting a disk group disables access to that disk group. A disk group may be deported so that its last VM disk can be removed, thus allowing the physical disk assigned to that VM disk to be reused. Once deported, the disk group is inaccessible and any VM disk icons that belonged to that disk group disappear along with the disk group view. The partition icon corresponding to a VM disk that disappeared when its disk group was deported reverts to an unshaded state. However, the partition retains its disk access record and knowledge of the deported disk group until it is reused (assigned to another disk group) or removed.

1. **Go to the view of the disk group to be deported.**

2. **From the Advanced-Ops menu, select Disk Group, then Deport Disk Group.**
   The Deport Disk Group form appears.

3. **The Deport Disk Group form should already display the current disk group name as the disk group to be deported.**
   If this is correct, select Apply to deport the disk group.

The disk group view disappears, along with any VM disks that it contained. The view button corresponding to the deported disk group also disappears from the GUI root window.

GUI-Related X Resources B

This appendix lists X resources that can be used to configure the GUI according to personal preferences and system requirements. The default values specified here correspond to those defaults that have been compiled into the GUI. Preferences specified in the system's `app-defaults` file may change these defaults.

The files in `/opt/vxva/app-defaults` contain a subset of resources that can be used to customize some aspects of the GUI, such as fonts and colors.

The X resources and associated preferences can be specified in the user's `.Xdefaults` file. Refer to the X Window System documentation on X resources for further information.

GUI entries in the `.Xdefaults` file should take the following form:

```
vxva*resource: value
```

For example, the color used to represent a disabled icon can be altered from the default color (grey) to orange by editing the `.Xdefaults` file to include the following line:

```
vxva*disabledPixel:orange
```

Resources for the GUI can also be specified for a single session by invoking the GUI as follows:

```
vxva -xrm "vxva*resource: value"
```

The following are the resources and their default values, which can be changed according to user preferences. The resources are listed to the left with their default values to the right. Each resource-value pair is prefaced by a brief description.

B.1 Color Resources

The following resources apply only when the GUI is run on a color monitor. The current default settings are listed below.

The basic background colors will affect the entire background, unless you specify an override value. The default background and foreground colors are as follows:

```
background                              white
foreground                              black
```

The colors used for all menus and menu bars, except the command window:

```
menubar*background                       SteelBlue
menubar*foreground                       white
```

The background color used under the main window buttons (both values must be set to the same color):

```
swindow*XmDrawingArea.background        #33648B
swindow*row_column.background           SteelBlue4
```

The background color used for the main window menu bar:

RootWin.menubar*background	SteelBlue
RootWin.menubar*foreground	White

The colors used for the command window menu bar:

cmd_main.cmd_menubar*background	SteelBlue
cmd_main.cmd_menubar*foreground	White

The basic outline colors used for the command window:

cmd_main*background	#4F94CD
cmd_main*cmd_pane*foreground	White

The colors used for the output text area in the command window:

cmd_main*cmd_pane*XmText.background	#63B8FF
cmd_main*cmd_pane*XmText.foreground	Black

The colors used for the history and the running command lists in the command window:

cmd_main*cmd_pane*XmScrolledWindow.XmList.background	#63B8FF
cmd_main*cmd_pane*XmScrolledWindow.XmList.foreground	Black

The color used for the menu bar in the view windows:

```
TopLevelShell.XmForm.menubar*foreground                        White

TopLevelShell.XmForm.menubar*background                        #63B8FF
```

The color used for the background in the view windows and scroll bars:

```
TopLevelShell.XmForm.ViewScroll.XmDrawingArea.background       #d9f0ff

TopLevelShell.XmForm.ViewScroll.XmScrollBar.background         White

TopLevelShell.XmForm.ViewScroll.background                     White
```

The background color used for forms:

```
TopLevelShell.propertiesForm*background                        #4F94CD
```

The color used for the foreground for text forms:

```
TopLevelShell.propertiesForm*foreground                        White
```

The highlight color used behind form text fields and toggle buttons. For consistency, use the same color:

```
TopLevelShell.propertiesForm.XmForm*XmText.background          #7EC0EE

TopLevelShell.propertiesForm.XmForm.Toggle.XmToggleButton*background    #7EC0EE
```

The highlight color used behind the file system information property sheet text fields and toggle buttons. For consistency, use the same color:

```
TopLevelShell.propertiesForm.XmForm.FsMenuForm*XmText.background         #7EC0EE

TopLevelShell.propertiesForm.XmForm.XmForm.Toggle.XmToggleButton.background    #7EC0EE
```

The default color used for a volume in a view:

```
TopLevelShell.XmForm.ViewScroll.XmDrawingArea*VolWidgetClass.background       #99cdeb
```

The default color used for a VM disk in a view:

```
TopLevelShell.XmForm.ViewScroll.XmDrawingArea*VmdiskWidgetClass.background    #99cdeb
```

The default color used for a physical disk in a view:

```
TopLevelShell.XmForm.ViewScroll.XmDrawingArea*DiskWidgetClass.background      #99cdeb
```

The default color used for a plex in a view:

```
TopLevelShell.XmForm.ViewScroll.XmDrawingArea*PlexWidgetClass.background      #a3dbfa
```

The default color used for a subdisk in a view:

```
TopLevelShell.XmForm.ViewScroll.XmDrawingArea*SubdiskWidgetClass.background   #c2efff
```

The default color used for a physical disk partition which is not associated with a disk group:

```
TopLevelShell.XmForm.ViewScroll.XmDrawingArea*PartWidgetClass.background      #c2efff
```

The colors used for the notice popup windows:

```
Notice_popup*background                                  SteelBlue

Notice_popup*foreground                                  White
```

The color used for icons that have been selected:

`selectedPixel`	`green`

The color used for icons that are disabled and cannot be used by the GUI. For example, the color to be used for detached plexes:

`disabledPixel`	`gray`

The color used for icons that have been selected when a GUI error occurs (incorrectly selected icons, for example):

`alarmPixel`	`red`

The color used for subdisk icons that are free (unassociated) when “Show Free Subdisks” has been turned on:

`freesdPixel`	`yellow`

The color used for icons that are projecting (displaying object relationships) when “Icon Projection” has been turned on for that icon or a related icon:

`projectPixel`	`yellow`

The color used for icons that have a *low* usage level (as defined in the Analysis Parameters Form) when analysis has been turned on for that icon or a related icon:

`lowPixel`	`green`

The color used for icons that have a *medium* usage level (as defined in the Analysis Parameters Form) when analysis has been turned on for that icon or a related icon:

`midPixel`	`yellow`

The color used for icons that have a *high* usage level (as defined in the Analysis Properties Form) when analysis has been turned on for that icon or a related icon:

`highPixel`	`red`

The color used to indicate whether the GUI should run in monochrome mode. When `True` is specified, the GUI is forced to operate in monochrome (black and white) mode, whether or not a color monitor is being used:

`mono`	`False`

B.2 Monochrome Resources

The following resources only apply when the GUI is run on a monochrome monitor.

The color in which all foreground items are displayed. This typically applies to icon outlines and text:

`foreground`	`black`

The color that serves as the background for all windows in the GUI:

`background`	`white`

The bitmap pattern for icons that have been selected:

selectedPixmap	gray3

The bitmap pattern for icons that are disabled and cannot be used by the GUI (detached plexes, for example):

disabledPixmap	stripe4

The bitmap pattern for icons that have been selected when a GUI error occurs (incorrectly selected icons, for example):

alarmPixmap	gray1

The bitmap pattern for subdisk icons that are free (unassociated) when "Show Free Subdisks" has been turned on:

freesdPixmap	root_weave

The bitmap pattern for icons that are projecting (displaying object relationships) when "Icon Projection" has been turned on for that icon or a related icon:

projectPixmap	root_weave

The bitmap pattern for icons that have a *low* usage level (as defined in the Analysis Parameters Form) when analysis has been turned on for that icon or a related icon:

lowPixmap	cross_weave

The bitmap pattern for icons that have a *medium* usage level (as defined in the Analysis Parameters Form) when analysis has been turned on for that icon or a related icon:

midPixmap	root_weave

The bitmap pattern for icons that have a *high* usage level (as defined in the Analysis Parameters Form) when analysis has been turned on for that icon or a related icon:

highPixmap	wide_weave

B.3 Icon Resources

The following resources relate to the GUI icons.

When `True` is specified, volume icons will be minimized when created, by default:

volumeMinimizeIcons	False

When `True` is specified, plex icons will be minimized when created, by default. This feature is useful to display structures within volumes, but to hide details about the subdisk structure that makes up the plex:

plexMinimizeIcons	False

When `True` is specified, disk icons will be minimized when created, by default:

diskMinimizeIcons	False

When `True` is specified, physical disk icons will be minimized when created, by default:

```
phyDiskMinimizeIcons                              False
```

When `True` is specified, icons selected for an operation are automatically deselected when the operation completes. If set to `False`, icons remain selected until the user decides to deselect them, making it possible to perform multiple operations on the same set of selected icons:

```
autoDeselect                                      True
```

B.4 Miscellaneous Resources

The following are miscellaneous resources.

When `True` is specified, the GUI will be run in tutorial mode whether or not the `-t` command line option has been specified. This function is useful for users who are learning to use the GUI:

```
tutorial                                          False
```

The title of the GUI's main window:

```
title                                             "GUI"
```

Specify the number of command silos supported. A *command silo* is a set of sequentially dependent commands (like file system create, followed by file system mount). A larger number of silos supports a larger number of concurrent operations that can be run, but also requires the GUI to use more memory:

```
commandSilos                                      50
```

Specify the number of commands that the GUI should remember and display in the history portion of the Command Info Window:

```
commandHistorySize          20
```

Specify the name of the disk group to be popped up by default when the GUI is run:

```
defaultViewWindow           NONE
```

Specify how often, in seconds, the GUI should check the system mount table to accurately display information about mounted file systems:

```
chkMntptInterval            5
```

When `True` is specified, the GUI will force the mouse to be remapped so that it can be used with a two button mouse:

```
twoButtonMouse              False
```

Changes the height of the GUI forms and the menu bars. For small resolution displays, this value can be set to `0`:

```
marginHeight                1
```

Changes the width of the GUI forms and menu bars. For small resolution displays, this value can be set to `0`:

```
marginWidth                 1
```

The following example shows you how to change the number of columns from its default value of `6` to `12` using a resource:

```
vxva*viewname*numColumns    12
```

Substitute *viewname* with World, `rootdg`, volumes, or user created view names.

B.5 Font Settings

This section lists the existing default font settings.

The following affects the font used in views of plexes, volumes, and subdisks:

```
vxva.fontList                          -*-lucidatypewriter-medium-r-*-*-*-120-*-*-*-*-*-*
```

The default font used for all menu bars:

```
menubar*fontList                       -*-helvetica-bold-r-normal-*-*-140-*-*-*-*-*-*
```

The font used for the command window menu bar:

```
cmd_main.cmd_menubar*fontList          -*-helvetica-bold-r-normal-*-*-140-*-*-*-*-*-*
```

The font used for the labels on the main window buttons:

```
swindow*row_column*fontList             -*-helvetica-medium-r-*-*-*-140-*-*-*-*-*-*
```

The font used for the form buttons:

```
TopLevelShell.XmForm.XmFrame*fontList   -*-helvetica-bold-r-normal-*-*-140-*-*-*-*-*-*
```

Volume Manager Error Messages C

C.1 Introduction

This appendix provides information on error messages associated with the Volume Manager configuration daemon (`vxconfigd`) and the kernel.

The appendix 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. Clarifications are included to elaborate on the situation or problem that may have generated a particular message. Wherever possible, a recovery procedure (action) is provided to locate and correct the problem.

The following sections are included in this appendix:

C.2 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/vxvm/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`.

For Solaris, 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
```

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
```

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 higest debug
# level is 9.

#debug=1                               # enable debugging console ouptut
```

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.

C.3 Volume Manager Configuration Daemon Error Messages

The Volume Manager is fault-tolerant and resolves most problems without system administrator intervention. If the volume 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.

C.3.1 `vxconfigd` Usage Messages

The following are usage messages associated with `vxconfigd`.

C.3.1.1 `-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`.

♦ **User Action**

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

C.3.1.2 `-x argument: invalid debug string`

```
-x argument: invalid debug string
```

♦ **Clarification**

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

♦ **User Action**

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

C.3.1.3 -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.

♦ **User Action**

Use a shorter prefix.

C.3.1.4 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`.

C.3.1.5 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.

♦ **User Action**

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

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

C.3.2 `vxconfigd` *Error Messages*

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

C.3.2.1 /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:

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

♦ **User 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 CD insert for information on how to add the package using `pkgadd`.

C.3.2.2 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.3 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.4 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.5 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.6 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.7 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.8 `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 send 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.

♦ **User 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.

C.3.2.9 `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.

♦ **User 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.

C.3.2.10 `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:

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

♦ **User 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 CD insert for information on how to add the package using `pkgadd`. If you cannot re-add the package, then contact Customer Support for more information.

C.3.2.11 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.

♦ **User 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” appendix.

C.3.2.12 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.13 `Cannot reset Volume Manager kernel`

```
vxvm:vxconfigd: ERROR: Cannot reset Volume Manager kernel: reason
```

♦ **Clarification**

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

♦ **User 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 Volume Manager 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.

C.3.2.14 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 user actions that caused all plexes to become unusable (for example, forcing the dissociation of subdisks or detaching, dissociation, or offlining of plexes).

♦ **User 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" appendix for information on how to fix problems with root or `/usr` file system volumes.

C.3.2.15 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 user actions that caused all plexes to become unusable (for example, forcing the dissociation of subdisks or detaching, dissociating, or offlining plexes).

♦ **User 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" appendix for information on how to fix problems with root or `/usr` file system volumes.

C.3.2.16 `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.

♦ **User 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" appendix for further information.

C.3.2.17 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.18 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 Volume Manager 1.3 to Volume Manager 2.0 and run `vxconfigd` without a reboot.

♦ **User Action**

To fix, reboot the system.

C.3.2.19 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 `vxdctl 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.

♦ **User 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 `vxdctl hostid` again. If it still results in an error, contact Customer Support.

C.3.2.20 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.

♦ **User 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.

C.3.2.21 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.

♦ **User 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.

C.3.2.22 `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:

1. 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).

2. 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.

♦ **User Action**

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

C.3.2.23 Disk group: Cannot recover temp database

```
vxvm:vxconfigd: ERROR: Disk group group: Cannot recover temp
database: reason
Consider use of "vxconfigd -x cleartempdir" [see vxconfigd(1M)].
```

♦ **Clarification**

This can happen if you kill and restart `vxconfigd` or you if you disable and enable it with `vxdctl disable` and `vxdctl 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.

♦ **User 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.

C.3.2.24 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`.

♦ **User 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.

C.3.2.25 `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.

♦ **User 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.

C.3.2.26 `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.

♦ **User 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.

C.3.2.27 `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
```

♦ **User 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.

C.3.2.28 `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 `vxdctl enable`. This error can 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.

♦ **User 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.

C.3.2.29 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.30 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.31 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.32 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.

♦ **User 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.

C.3.2.33 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.

♦ **User 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
```

C.3.2.34 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.

♦ **User Action**

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

C.3.2.35 `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.

♦ **User Action**

If the error was "`No such file or directory`," then create the directory (using `mkdir`). Then run the command `vxdctl 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.

C.3.2.36 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/rdsk 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.

♦ **User 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.

C.3.2.37 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.

♦ **User Action**

Contact Customer Support.

C.3.2.38 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.

♦ **User Action**

Boot from the one of the disks named in the error message. Under Solaris, you may be able to boot using a device alias for one of the named disks. For example, try:

`boot vx-`*diskname*

C.3.2.39 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.

♦ **User Action**

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

C.3.2.40 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:

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.

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.

♦ **User 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 `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.

C.3.2.41 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.42 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.43 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.44 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.2.45 `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`.

♦ **User Action**

Supply a correct option argument.

C.3.2.46 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.

♦ **User Action**

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

C.3.2.47 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 few possible causes of this error:

1. The /etc/vfstab file was erroneously updated to indicate the device for the /usr file system is a volume, but the volume named is not in the rootdg disk group. This should happen only as a result of direct manipulation by the administrator.

2. 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.

♦ **User 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” appendix. 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.

C.3.2.48 `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.

♦ **User 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.

C.3.3 `vxconfigd` *Fatal Error Messages*

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

C.3.3.1 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.3.2 `Group 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.3.3 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.3.4 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.3.5 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.

♦ **User 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.

C.3.3.6 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.3.7 `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 Solaris multi-threading libraries.

♦ **User Action**

Contact Customer Support for more information.

C.3.4 `vxconfigd` *Notice Messages*

The following are notice messages associated with `vxconfigd`.

C.3.4.1 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.

♦ **User Action**

If there is an active hot-spare disk for the disk group, then the disk failure may be taken care of automatically. Mail will be sent to the `root` user indicating what actions were taken by the Volume Manager, and what further actions the administrator should take.

C.3.4.2 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 `vxdg -k rmdisk`. A failing disk should be indicated by a `Detached disk` *disk* message.

♦ **User Action**

Remove the failing logs using either `vxplex dis` or `vxsd dis.` Then, use `vxassist addlog` [see `vxassist`(1M)] to add a new log to the volume.

C.3.4.3 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 `vxdg -k rmdisk`. A failing disk should be indicated by a `Detached disk` *disk* message.

♦ **User Action**

If there is an active hot-spare disk for the disk group, then the disk failure may be taken care of automatically. Mail will be sent to the `root` user indicating what actions were taken by the Volume Manager, and what further actions the administrator should take.

C.3.4.4 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 `vxdg -k rmdisk`. A failing disk should be indicated by a `Detached disk` *disk* message.

♦ **User Action**

If there is an active hot-spare disk for the disk group, then the disk failure may be taken care of automatically. Mail will be sent to the `root` user indicating what actions were taken by the Volume Manager, and what further actions the administrator should take.

C.3.4.5 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 `vxdg -k rmdisk`. A failing disk should be 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.

♦ **User Action**

If there is an active hot-spare disk for the disk group, then the disk failure may be taken care of automatically, though the volume contents cannot be recovered automatically. Mail will be sent to the `root` user indicating what actions were taken by the Volume Manager, and what further actions the administrator should take.

C.3.4.6 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.

♦ **User 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.

C.3.4.7 `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.

♦ **User Action**

If there is an active hot-spare disk for the disk group, then the disk failure may be taken care of automatically. Mail will be sent to the `root` user indicating what actions were taken by the Volume Manager, and what further actions the administrator should take.

C.3.5 `vxconfigd` *Warning Messages*

The following are warning messages associated with `vxconfigd`.

C.3.5.1 `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.

♦ **User 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.

C.3.5.2 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.5.3 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/rdsk`. This should happen only if the root file system has run out of inodes.

♦ **User Action**

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

```
vxdctl enable
```

C.3.5.4 `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 will continue to use space in the root file system. However, this does imply that the `/usr` file system is not mounted, or that the `rm` utility is missing or is not in its usual spot. These may be serious problems for the general running of your system.

♦ **User Action**

If the `/usr` file system is not mounted, then you will need to determine how to get it mounted. If the `rm` utility is missing or is not in the `/usr/bin` directory, then you should restore it from somewhere.

C.3.5.5 `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.

♦ **User 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.

C.3.5.6 `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.

- **User Action**
  Contact Customer Support for more information.

C.3.5.7 `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.

- **User Action**

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

C.3.5.8 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.5.9 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.5.10 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.5.11 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.

♦ **User 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`.

C.3.5.12 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.

♦ **User 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.

C.3.5.13 `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.

♦ **User 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.

C.3.5.14 `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 subdisks on this disk will be unusable; any RAID-5 plexes or regular 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.

♦ **User Action**

If there is an active hot-spare disk for the disk group, then the disk failure may be taken care of automatically. Mail will be sent to the `root` user indicating what actions were taken by the Volume Manager, and what further actions the administrator should take.

C.3.5.15 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.5.16 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.

♦ **User 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.

C.3.5.17 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 `vxdctl hostid` operation.

♦ **User 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 `vxdg import` with the `-C` option.

C.3.5.18 `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.

♦ User 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.

C.3.5.19 `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.

♦ **User 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.

C.3.5.20 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).

♦ **User Action**

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

```
vxdctl rm disk device
```

C.3.5.21 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.5.22 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.5.23 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).

♦ **User Action**

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

```
vxdctl rm disk device
```

C.3.5.24 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.

♦ **User Action**

Contact Customer Support for more information.

C.3.5.25 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.

♦ **User Action**

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

C.3.5.26 `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.

♦ **User Action**
Contact Customer Support for more information.

C.3.5.27 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.5.28 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.5.29 `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.

♦ **User Action**

Contact Customer Support for more information.

C.3.5.30 `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.

♦ **User Action**

Contact Customer Support for more information.

C.4 Kernel Error Messages

The following sections cover the kernel level error messages.

C.4.1 Kernel Notice Messages

The following are notice messages associated with the kernel.

C.4.1.1 `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.

C.4.1.2 `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.

C.4.1.3 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.

C.4.1.4 *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.

C.4.2 *Kernel Warning Messages*

The following are warning messages associated with the kernel.

C.4.2.1 `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.

C.4.2.2 `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.

C.4.2.3 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.

C.4.2.4 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.

C.4.2.5 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. See Appendix E, “Recovery” for further information.

C.4.2.6 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.

C.4.2.7 `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.

C.4.2.8 `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.

C.4.2.9 `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.

C.4.2.10 `mod_install returned` *errno*

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

♦ **Clarification**

A call made to the Solaris `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.

C.4.2.11 `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.

C.4.2.12 `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.

C.4.2.13 `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.

♦ **User Action**

Contact Customer Support for more information.

C.4.2.14 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 user 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.

C.4.2.15 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.

C.4.2.16 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.

C.4.2.17 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.

C.4.2.18 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.

C.4.2.19 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.

♦ **User Action**

Contact Customer Support for more information.

C.4.2.20 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.

C.4.2.21 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.

C.4.2.22 `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.

C.4.3 *Kernel Panic Messages*

The following are panic messages associated with the kernel.

C.4.3.1 `Can't clear group commit log record for group`

```
WARNING: vxlog_dgfree: Can't clear group commit log record for
   group disk_group
```

♦ **Clarification**

This can occur if a log flush to disk could not be performed because no valid log copies remained. This is likely to compromise the ability of the Volume Manager to recover from any further I/O errors.

♦ **User Action**

Disks should be added to the system such that new viable logging areas can be generated. Alternatively, failed disks should be removed and replaced with working devices.

C.4.3.2 Can't close disk in group

```
NOTICE: vxvm: Can't close disk disk_name 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 error_number/error_number
```

♦ **Clarification**

This is unlikely to happen; closes cannot fail.

♦ **User Action**

None.

C.4.3.3 Can't free kernel logging area for vx_reset_kernel of group

```
WARNING: volklog_dgfree: Can't free kernel logging area for
   vol_reset_kernel of group disk_group
```

♦ **Clarification**

A free of the logs for a disk group failed because either no valid log areas remained for flushing or some log records remained in the log before the clear operation was requested.

♦ **User Action**

No user action can be taken here; this is a Volume Manager internal error. Contact Customer Support.

C.4.3.4 Can't open disk in group

```
NOTICE: vxvm: Can't open disk disk_name 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 error_number
```

♦ **Clarification**

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

♦ **User Action**

Turn on the drive.

C.4.3.5 Correctable error on volume, plex, block

```
NOTICE: io/vol.c(volerror): Correctable type error on volume
   volume_name, plex plex_name, block block_number
```

♦ **Clarification**

A correctable I/O error was detected and corrected. A correctable I/O error is one where a read error from an underlying device driver could be corrected by reading the data from an alternate mirror copy and then writing it back to the failed mirror.

♦ **User Action**

If the I/O could have been completed by reading from an alternate mirror but the writeback to the failed mirror still failed, the mirror will be detached. This failure will cause the exception handling code to be entered, which will result in the volume's error recovery policy being followed. This usually results in either a mirror or the volume becoming detached. The user must intervene to reattach the mirror (`vxplex att`), in order to bring back the failed mirror copy. If the volume was detached, then the data contained on it is unrecoverable and will have to be restored from backups.

C.4.3.6 Failed to set up the root disk

```
WARNING: volroot.c: Failed to set up the root disk, error
error_number
```

♦ **Clarification**

This can occur if the disk cannot be opened. This might be because of a device error that should have already caused a message to be output.

♦ **User Action**

The only recovery possible here is to correct the hardware problem and reboot.

C.4.3.7 Failed to setup root volume

```
WARNING: volroot.c: Failed to setup root volume, error
error_number
```

♦ **Clarification**

This can happen if the open of the root device passed in an invalid device number for the volume or an invalid open type. This is not very likely.

♦ **User Action**

This situation should not occur. If it does, the only possible recovery might be to reboot from an alternate device or to reload the system.

C.4.3.8 Failed to set up the swap disk

```
WARNING: volroot.c: Failed to set up the swap disk, error
error_number
```

♦ **Clarification**

This can occur if the disk cannot be opened. This might be because of a device error that should have already caused a message to be output.

♦ **User Action**

The only recovery possible here is to correct the hardware problem and reboot.

C.4.3.9 `Failed to setup swap volume`

```
WARNING: volroot.c: Failed to setup swap volume, error
error_number
```

♦ **Clarification**

This can happen if the open of the swap device passed in an invalid device number for the volume or an invalid open type. This is not very likely.

♦ **User Action**

This situation should not occur. If it does, the only possible recovery might be to reboot from an alternate device or to reload the system.

C.4.3.10 `NOTICE: message on volume device hex_device_number (volume_name) in diskgroup group_name`

```
NOTICE: message on volume device hex_device_number (volume_name)
in diskgroup group_name
```

♦ **Clarification**

This is caused by a driver above the Volume Manager level calling the Volume Manager `vxprint()` function. This usually happens when a driver detects some error condition in Volume Manager and wishes to display the error.

♦ **User Action**

No action necessary, unless specified in a supplied string.

C.4.3.11 No volume error daemon - Cannot Log plex detach, detaching volume

```
WARNING: io/vol.c(volexcept): No volume error daemon - Cannot Log
plex detach, detaching volume
```

♦ **Clarification**

No vxiod process was running and able to log a detach record for a mirror that is being detached due to an I/O error. This is a fatal error that causes future access to the volume to be rejected, since any system failure coming after additional I/O would not be able to detect the failure of the mirror and mirror inconsistencies might then occur.

♦ **User Action**

Although it is too late to rescue this volume, at least one vxiod process should be started as soon as possible (using `vxiod set 2`). The failed volume will have to be stopped and restarted, then reloaded from backups. Mirrors will have become inconsistent and so any attempt at using the data on the volume could prove disastrous.

C.4.3.12 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.

♦ **User Action**

Contact Customer Support for more information.

C.4.3.13 Uncorrectable error on volume, plex, block

```
NOTICE: io/vol.c(volerror): Uncorrectable type
error on volume volume_name, plex plex_name, block block_number
```

♦ **Clarification**

Following an I/O error from one mirror, an attempt to reread the data from an alternate mirror failed. This could be because no other mirrors exist or could be because the other mirrors also had I/O failures.

♦ **User Action**

This failure will cause the exception handling code to be entered, which will result in the volume's error recovery policy being followed. This can have effects ranging from detaching a mirror to disabling the volume. The user must intervene to reattach the mirror (`vxplex att`), in order to bring back the failed mirror copy. If the volume was detached, then the data contained on it is unrecoverable and will have to be restored from backups.

SPARCstorage Array Firmware and Device Driver Error Messages

D

This appendix gives error messages specific to the SPARCstorage Array. Since these error messages are SPARCstorage Array-specific, you may see them even if you are not running the Volume Manager software.

D.1 Message Formats

Error indications from the SPARCstorage Array drivers (pln and soc) are always sent to syslog (/var/adm/messages). Additionally, depending on the type of event that generated the message, it may be sent to the console. These messages are limited to significant events like cable disconnections. Messages sent to the console are of the form:

```
[WARNING:]  instance:  <message>
```

The syslog messages may contain additional text. This message ID identifies the message, its producer, and its severity.:

```
ID[SUNWssa.soc.messageid.####] instance: <message>
```

Some examples

```
soc3: Transport error:  Fibre Channel Online Timeout
ID[SUNWssa.soc.link.6010] soc1: port: 0 Fibre Channel is ONLINE
```

In the following discussion, messages will be presented with the message ID and the message text, even though the message ID is not displayed on the console. The character "#" implies a numeric quantity and "..." implies a string of characters or numbers. The prefix "ID[SUNWssa]" is implied, and not shown.

```
soc.link.6010     soc#: port: # Fibre Channel is ONLINE
```

Note that most disk drive and media-related errors will result in messages from the sd or ssd drivers. See the manual pages for `sd(7)` for information on these messages.

D.2 System Configuration Errors

This class of errors may occur because of insufficient system resources (for example, not enough memory to complete installation of the driver), or because of hardware restrictions of the machine into which the SPARCstorage Array host adapter is installed.

This class of errors may also occur when your host system encounters a hardware error on the host's system board, such as a failed SIMM.

soc driver

```
soc.attach.4004 soc#: attach failed: bad soft state
soc.attach.4010 soc#: attach failed: unable to map eeprom
soc.attach.4020 soc#: attach failed: unable to map XRAM
soc.attach.4030 soc#: attach failed: unable to map registers
soc.attach.4040 soc#: attach failed: unable to access status register
soc.attach.4050 soc#: attach failed: unable to access hostadapter XRAM
soc.attach.4060 soc#: attach failed: unable to install interrupt handl
soc.attach.4003 soc#: attach failed: alloc soft state
soc.attach.4070 soc#: attach failed: offline packet structure allocat
```

These messages indicate that the initialization of the soc driver was unable to complete due to insufficient system virtual address mapping resources or kernel memory space for some of its internal structures. The host adapter(s) associated with these messages will not be functional.

```
soc.driver.4020   soc#: alloc of request queue failed
soc.driver.4040   soc#: DVMA request queue alloc failed
soc.driver.4050   soc#: alloc of response queue failed
soc.driver.4060   soc#: DVMA response queue alloc failed
soc.driver.4070   soc#: alloc failed
soc.driver.4090   soc#: alloc failed
soc.driver.4100   soc#: DMA address setup failed
soc.driver.4110   soc#: DVMA alloc failed
```

These messages indicate there are not enough system DVMA or kernel heap resources available to complete driver initialization. The associated host adapter(s) will be inoperable if any of these conditions occurs.

```
soc.attach.4001   soc#: attach failed: device in slave-only slot
soc.attach.4002   soc#: attach failed: hilevel interrupt unsupported
soc.driver.4001 soc#: Not self-identifying
```

The SBus slot into which the host adapter is installed cannot support the features required to operate the SPARCstorage Array. The host adapter should be relocated to a different SBus slot. If you see this error message, it's possible that you are running an unsupported configuration (for example, you may have the SPARCstorage Array connected to a server that is not supported).

pln driver

```
pln_ctlr_attach: controller struct alloc failed
pln_ctlr_attach: scsi_device alloc failed
pln_ctlr_attach: pln_address alloc failed
pln_ctlr_attach: controller struct alloc failed
pln_ctlr_attach: scsi_device alloc failed
pln_ctlr_attach: pln_address alloc failed
```

The pln driver was unable to obtain enough kernel memory space for some of its internal structures if one of these messages is displayed. The SPARCstorage Array(s) associated with these messages will not be functional.

```
pln_init: mod_install failed error=%d
```

Module installation of the pln driver failed. None of the SPARCstorage Arrays connected to the machine will be operable.

D.3 Hardware Errors

Errors under this classification are generally due to hardware failures (transient or permanent), or improper configuration of some subsystem components.

soc driver

```
soc.wwn.3010      soc#: No SSA World Wide Name, using defaults
```

The associated SPARCstorage Array has an invalid World Wide Name. A default World Wide Name is being assumed by the software – the system will still function with a default World Wide Name if only one SSA gives this message (they all would all be using the same default WWN). A valid World Wide Name should be programmed into the SPARCstorage Array (refer to the `ssacli.1m` man pages for more information).

```
soc.wwn.3020     soc#: Could not get port world wide name
```

If there is a failure on the SSA and the driver software is unable to obtain the devices WWN, this message is displayed.

```
soc.wwn.5020     soc#: INCORRECT WWN: Found: ... Expected: ...
```

This message is usually the result of plugging the wrong fibre channel cable into a host adapter. It indicates that the World Wide Name of the device connected to the host adapter does not match the World Wide Name of the device connected when the system was booted.

```
soc.driver.3010   soc#: host adapter fw date code: <not available>
```

This may appear if no date code is present in the host adapter microcode. This situation should not occur under normal circumstances and possibly indicates the use of invalid SPARCstorage Array drivers or a failed host adaptor.

For reference, the expected message is:

```
soc.driver.1010   soc#: host adapter fw date code: ...
```

This is printed at boot time to indicate the revision of the microcode loaded into the host adapter.

```
soc.link.4060     soc#: invalid FC packet; ...
```

The soc driver has detected some invalid fields in a packet received from the host adapter. The cause of this is most likely incorrectly functioning hardware (either the host adapter itself or some other SBus hardware).

```
soc.link.4020     soc#: Unsupported Link Service command: ...
soc.link.4030     soc#: Unknown FC-4 command: ...
soc.link.4040     soc#: unsupported FC frame R_CTL: ...
soc.link.4010     soc#: incomplete continuation entry
soc.link.3010     soc#: unknown LS_Command
```

pln driver

```
Transport error:  Received P_RJT status, but no header
Transport error:  Fibre Channel P_RJT
Transport error:  Fibre Channel P_BSY
```

These messages indicate the presence of invalid fields in the fibre channel frames received by the host adapter. This may indicate a fibre channel device other than Sun's fibre channel device for the SPARCstorage Array. The messages may also be caused by a failed host adaptor, Fibre Channel Optical Module, fibre optic cable or array controller.

```
soc.link.4080 soc#: Connections via Fibre Channel Fabric are unsupported
```

The current SPARCstorage Array software does not support fibre channel fabric (switch) operation. This message indicates that the software has detected the presence of a fabric.

```
soc.login.5010    soc#: Fibre Channel login failed
soc.login.5020    soc#: fabric login failed
soc.login.5030    soc#: N-PORT login not successful
soc.login.5040    soc#: N-PORT login failure
```

These messages may occur if part of the fibre channel link initialization or login procedures fail. Retries of the login procedure will be performed.

```
soc.login.6010    soc#: Fibre Channel login succeeded
```

The soc driver will display this message following a successful fibre channel login procedure (part of link initialization) if the link had previously gone from an operable to an inoperable state. The "login succeeded" message indicates the link has again become fully functional.

```
soc.login.4020    soc#: login retry count exceeded for port: #
soc.login.4040    soc#: login retry count exceeded
```

These errors indicate that the login retry procedure is not working and the port/card associated with the message is terminating the login attempt. The associated SPARCstorage Array will be inaccessible by the system.

Note that the fibre channel specification requires each device to attempt a login to a fibre channel fabric, even though one may not be present. A failure of the fabric login procedure due to link errors (even in a point-to-point topology) may result in the printing of fabric login failure messages even with no fabric present.

```
Link errors detected
```

A number of retryable errors may have occurred on the fibre channel link. This message may be displayed if the number of link errors exceeds the allowable link bit error rate (1 bit/10^{12} bits). If you see this message, clean the fibre optic cable according to the instructions given in the *SPARCstorage Array Service Manual*. If the problem still exists, replace either the fibre optic cable or the Fibre Channel Optical Module.

pln driver

```
Transport error:  FCP_RSP_CMD_INCOMPLETE
Transport error:  FCP_RSP_CMD_DMA_ERR
Transport error:  FCP_RSP_CMD_TRAN_ERR
Transport error:  FCP_RSP_CMD_RESET
Transport error:  FCP_RSP_CMD_ABORTED
```

An error internal to the SPARCstorage Array controller has occurred during an I/O operation. This may be due to a hardware failure in a SCSI interface of the SPARCstorage Array controller, a failure of the associated SCSI bus (drive tray) in the SPARCstorage Array package, or a faulty disk drive.

```
Transport error:  FCP_RSP_CMD_TIMEOUT
```

The SCSI interface logic on the SPARCstorage Array controller board has timed out on a command issued to a disk drive. This may be caused by a faulty drive, drive tray, or array controller.

```
Transport error:  FCP_RSP_CMD_OVERRUN
```

This error (on an individual I/O operation) may indicate either a hardware failure of a disk drive in the SPARCstorage Array, a failure of the associated drive tray, or a fault in the SCSI interface on the SPARCstorage Array controller. The system will try to access the failed hardware again after you see this message.

```
Transport error:  FCP_RSP_SCSI_PORT_ERR
```

The firmware on the SPARCstorage Array controller has detected the failure of the associated SCSI interface chip. Any I/O operations to drives connected to this particular SCSI bus will fail. If you see this message, you may have to replace the array controller.

```
Transport error:  Fibre Channel Offline
soc.link.6010     soc#: port: # Fibre Channel is ONLINE
```

If you see these messages together, the system was able to recover from the error, so no action is necessary.

```
Transport error:  Fibre Channel Offline
Transport error:  Fibre Channel Online Timeout
```

If you see these messages together, an I/O operation to a SPARCstorage Array drive has failed because the fibre channel link has become inoperable. The driver will detect the transition of the link to an inoperable state and will then initiate a timeout period. Within the timeout period, if the link should become usable again, any waiting I/O operations will be resumed. However, if the timeout should expire before the link becomes operational, any I/O operations will fail.

The latter message (timeout) means that the host adapter microcode has detected a timeout on a particular I/O operation. This message will be printed (and the associated I/O operation will fail) only if the retry count of the driver for this class of link errors has been exhausted.

```
Transport error:  CMD_DATA_OVR
Transport error:  Unknown CQ type
Transport error:  Bad SEG CNT
Transport error:  Fibre Channel Invalid X_ID
Transport error:  Fibre Channel Exchange Busy
Transport error:  Insufficient CQEs
Transport error:  ALLOC FAIL
Transport error:  Fibre Channel Invalid S_ID
Transport error:  Fibre Channel Seq Init Error
Transport error:  Unknown FC Status
```

These errors indicate the driver or host adapter microcode has detected a condition from which it cannot recover. The associated I/O operation will fail. This message should be followed or preceded by other error messages; refer to these other error messages to determine what action you should take to fix the problem.

```
Timeout recovery failed, resetting
```

This message may be displayed by the pln driver if the normal I/O timeout error recovery procedures were unsuccessful. In this case, the software will perform a hardware reset of the host adapter and attempt to continue system operation.

```
reset recovery failed
```

This message will be printed only if the hardware reset error recovery has failed, following the failure of normal fibre channel link error recovery. The associated SPARCstorage Array(s) will be inaccessible by the system. This situation should only occur due to failed host adapter hardware.

D.4 Informational Messages:

Messages in this category will be used to convey some information about the configuration or state of various SPARCstorage Array subsystem components.

soc driver

```
soc.driver.1010   soc#: host adapter fw date code: ...
```

This string will be printed at boot time to indicate the revision of the microcode loaded into the host adapter.

```
soc.link.6010     soc#: port: # Fibre Channel is ONLINE
soc.link.5010     soc#: port: # Fibre Channel is OFFLINE
```

Under a variety of circumstances, the fibre channel link may appear to the host adapter to have entered an inoperable state. Frequently, such a condition is temporary.

The following are possible causes for the fibre channel link to appear to go “offline”:

- A temporary burst of errors on the fibre cable. In this case, the “OFFLINE” message should be followed by an “ONLINE” message shortly afterwards.
- Unplugging of the fibre channel cable from either the host adapter or the SPARCstorage Array

- Powering off a connected SPARCstorage Array
- Failure of an Fibre Channel Optical Module in either the host adapter or the SPARCstorage Array
- Failure of an optical cable
- Failure of a SPARCstorage Array controller
- Failure of a host adapter card

Note that any pending I/O operations to the SPARCstorage Array will be held by the driver for a period of time (one to two minutes) following a link “offline” in case the link should return to an operable state so that pending operations can be completed. However, if a sufficient time elapses following the transition of the link to “offline” without a corresponding “online” transition, the driver will fail the I/O operations associated with the formerly connected SPARCstorage Array.

It is normal to see the ONLINE message for each connected SPARCstorage Array when the system is booting.

```
soc.link.1010  soc#: message: ...
```

Peripheral devices on the Fibre Channel (like the SPARCstorage Array) can cause messages to be printed on the system console/syslog under certain circumstances.

Under normal operation at boot time, the SSA will display the revision date of its firmware following a fibre channel login. This message will be of the form:

```
soc.link.1010 soc#: message:SSA EEprom date: Fri May 27 12:35:46 1994
```

Other messages from the controller may indicate the presence of warning or failure conditions detected by the controller firmware.

D.5 Internal Software Errors

These messages may be printed by the driver in a situation where it has detected some inconsistency in the state of the machine. These may sometimes be the result of failed hardware, usually either the SPARCstorage Array host adapter or SBus hardware.

These are not expected to occur under normal operation.

soc driver

```
soc.driver.4010   soc#: Illegal state: SOC_COMPLETE == 0
soc.driver.4030   soc#: too many continuation entries
soc.driver.4080   soc#: no unsolicited commands to get
soc.link.3020     soc#: unknown status: ...
soc.link.4050     soc#: unsolicited: Illegal state: flags: ...
soc.link.4070     soc#: invalid fc_ioclass
soc.login.1010    soc#: reset with resets disabled
```

pln driver

```
ddi_dma_sync failed (rsp)
Invalid transport status
Unknown state change
Grouped disks not supported
pln_scsi_pktfree: freeing free packet
```

Recovery

E

Disk failures can cause two types of problems: loss of data on the failed disk and loss of access to your system due to the failures of a key disk (a disk involved with system operation). The Volume Manager provides the ability to protect your system from either type of problem. Volume Manager allows you to use *mirroring* to protect the data. By mirroring your data, you prevent data loss from a disk failure. By mirroring drives critical to system operation, you ensure that no single disk failure will leave your system unusable.

E.1 Plex States

Plex states reflect whether or not plexes are complete and consistent copies (mirrors) of the volume contents. Volume Manager utilities automatically maintain the plex state. However, a system administrator can modify the state of a plex if changes to the volume with which the plex is associated should not be written to the plex. For example, if a disk with a particular plex located on it begins to fail, that plex can be temporarily disabled.

Note – A plex does not have to be associated with a volume. A plex can be created from the command line interface and later attached to a volume if required.

Volume Manager utilities use plex states to:

- Indicate whether volume contents have been initialized to a known state
- Determine if a plex contains a valid copy (mirror) of the volume contents
- Track whether a plex was in active use at the time of a system failure
- Monitor operations on plexes

This section explains plex states in detail. This section is designed for users who wish to have a detailed knowledge of plex states.

Plexes that are associated with a volume always have one of the following states:

- EMPTY
- CLEAN
- ACTIVE
- STALE
- OFFLINE
- TEMP
- TEMPRM
- IOFAIL

E.1.1 EMPTY Plex State

Volume creation sets all plexes associated with the volume to the EMPTY state to indicate to the usage type utilities.

E.1.2 CLEAN Plex State

A plex is in a CLEAN state when it is known to contain a consistent copy (mirror) of the volume contents and an operation has disabled the volume. As a result, when all plexes of a volume are clean, no action is required to guarantee that the plexes are identical when that volume is started.

E.1.3 ACTIVE Plex State

A plex can be in the ACTIVE state in two situations:

- When the volume is started and the plex fully participates in normal volume I/O (meaning that the plex contents change as the contents of the volume change)
- When the volume was stopped as a result of a system crash and the plex was ACTIVE at the moment of the crash

In the latter case, a system failure may leave plex contents in an inconsistent state. When a volume is started, Volume Manager performs a recovery action to guarantee that the contents of the plexes that are marked as ACTIVE are made identical.

Note – On a well-running system, ACTIVE should be the most common state you see for any volume's plexes.

E.1.4 STALE Plex State

If there is a possibility that a plex does not have the complete and current volume contents, that plex is placed in the STALE state. Also, if an I/O error occurs on a plex, the kernel stops using and updating the contents of that plex, and an operation sets the state of the plex to STALE.

A `vxplex attach` operation revives STALE plexes from an ACTIVE plex. Atomic copy operations copy the contents of the volume to the STALE plexes. The system administrator can force a plex to the STALE state with an utility operation.

E.1.5 OFFLINE Plex State

The `vxmend off` operation indefinitely detaches a plex from a volume by setting the plex state to OFFLINE. Although the detached plex maintains its association with the volume, changes to the volume do not update the OFFLINE plex until the plex is put online and reattached with the `vxplex att` operation. When this occurs, the plex is placed in the STALE state, which causes its contents to be recovered at the next `volume start` operation.

E.1.6 TEMP Plex State

Setting a plex to the TEMP state facilitates some plex operations that cannot occur in a truly atomic fashion. For example, attaching a plex to an enabled volume requires copying volume contents to the plex before it can be considered fully attached.

A utility will set the plex state to TEMP at the start of such an operation and to an appropriate state at the end of the operation. If the system goes down for any reason, a TEMP plex state indicates that the operation is incomplete; a subsequent `volume start` will dissociate plexes in the TEMP state.

E.1.7 TEMPRM Plex State

A TEMPRM plex state resembles a TEMP state except that at the completion of the operation, TEMPRM plex is removed. Some subdisk operations require a temporary plex. Associating a subdisk with a plex, for example, requires updating the subdisk with the volume contents before actually associating the subdisk. This update requires associating the subdisk with a temporary plex, marked TEMPRM, until the operation completes and removes the TEMPRM plex.

If the system goes down for any reason, the TEMPRM state indicates that the operation did not complete successfully. A subsequent operation will dissociate and remove TEMPRM plexes.

E.1.8 IOFAIL Plex State

The IOFAIL plex state is associated with persistent state logging. On the detection of a failure of an ACTIVE plex, `vxconfigd` places that plex in the IOFAIL state so that it is disqualified from the recovery selection process at volume start time.

E.1.9 The Plex State Cycle

The changing of plex states accompanies normal operations. Deviations in plex state indicate abnormalities that Volume Manager must normalize. At startup, the `volume start` operation makes all CLEAN plexes ACTIVE. If all goes well until shutdown, the volume-stopping operation marks all ACTIVE plexes

CLEAN and the cycle continues. Having all plexes CLEAN at startup (before `volume start` makes them ACTIVE) indicates a normal shutdown and optimizes startup.

If a crash occurred, the volume-starting operation finds no CLEAN plexes, only ACTIVE ones. The operation then establishes one plex as an up-to-date and suitable source for reviving the other plexes, and marks that source plex ACTIVE and the others STALE. The volume usage type determines which plex is selected as the source plex.

If an I/O error occurred and caused a plex to become disabled, the volume-stopping operation marks the plex in which the error occurred as STALE. Any STALE plexes require recovery. When the system restarts, a utility copies data from an ACTIVE to a STALE plex and makes the STALE plex ACTIVE.

E.1.10 Plex Kernel State

This state indicates the accessibility of the plex. The plex kernel state is monitored in the volume driver and allows a plex to have an off-line (DISABLED), maintenance (DETACHED), and on-line (ENABLED) mode of operation.

- DISABLED — The plex may not be accessed.
- DETACHED — A write to the volume is not reflected to the plex. A read request from the volume will never be satisfied from the plex device. Plex operations and ioctl functions are accepted.
- ENABLED — A write request to the volume will be reflected to the plex, if the plex is set to ENABLED for write mode. A read request from the volume is satisfied from the plex if the plex is set to ENABLED.

Note – No user intervention is required to set these states, they are maintained internally. On a system that is operating properly, all mirrors are enabled.

E.2 Volume States

There are four volume states, some of which are similar to plex states. Following is a list of volume states for non-RAID-5 volumes:

- CLEAN—The volume is not started (kstate is DISABLED) and its plexes are synchronized.
- ACTIVE—The volume has been started (kstate is currently ENABLED) or was in use (kstate was ENABLED) when the machine was rebooted. If the volume is currently ENABLED, the state of its plexes at any moment is not certain (since the volume is in use). If the volume is currently DISABLED, this means that the plexes cannot be guaranteed to be consistent.
- EMPTY—The volume contents are not initialized. The kernel state (kstate) is always DISABLED when the volume is EMPTY.
- SYNC—The volume is either in read-writeback mode (kstate is currently ENABLED) or was in read-writeback mode when the machine was rebooted (kstate is DISABLED). If the volume is ENABLED, this means that the plexes are being resynchronized via the read-writeback recovery. If the volume is DISABLED, it means that the plexes were being resynchronized via read-writeback when the machine rebooted and therefore still need to be synchronized.

The interpretation of these flags during volume startup is modified by the persistent state log for the volume (for example, the dirty/clean flag). If the clean flag is set, this means that an ACTIVE volume was not written to by any processes or was not even open at the time of the reboot; therefore, it can be considered CLEAN. The clean flag will always be set in any case where the volume is marked CLEAN.

Following is a listing of volume states for RAID-5 volumes:

- CLEAN—The volume is not started (kernel state is DISABLED) and its parity is good. The RAID-5 plex stripes are consistent.
- ACTIVE—The volume has been started (kernel state is currently ENABLED) or was in use (kernel state was ENABLED) when the machine was rebooted. If the volume is currently ENABLED, the state of its RAID-5 plex at any moment is not certain (since the volume is in use). If the volume is currently DISABLED, this means that the parity cannot be guaranteed to be synchronized.

- EMPTY—The volume contents are not initialized. The kernel state is always DISABLED when the volume is EMPTY.
- SYNC—The volume is either undergoing a parity resynchronization (kernel state is currently ENABLED) or was having its parity resynchronized when the machine was rebooted (kernel state is DISABLED).
- NEEDSYNC—The volume will require a parity resynchronization operation the next time it is started.

E.2.1 Volume Kernel State

The *volume kernel state* indicates the accessibility of the volume. The volume kernel state allows a volume to have an off-line (DISABLED), maintenance (DETACHED), and on-line (ENABLED) mode of operation.

- DISABLED—The volume cannot be accessed.
- DETACHED—The volume cannot be read or written, but plex device operations and ioctl functions are accepted.
- ENABLED—The volumes can be read and written.

E.3 Protecting Your System

In order to maintain system availability in the event of disk failure, the data important to running and booting your system must be mirrored. Furthermore, it must be preserved in such a way that it can be used in case of failure. The most difficult part of this is the ability to boot the system after a failure of a disk that is critical to the boot process.

To preserve data, you should create and use volumes that have at least two mirrors. The mirrors must be on different disks. The `vxassist` utility locates the mirrors such that the loss of one disk will not result in a loss of data. You can edit the file `/etc/default/vxassist` to set the default number of mirrors for newly created volumes to two.

You must also do regular backups (of all data except the root file system) to protect your data. Backups are necessary if all copies of a volume are lost or corrupted in some way.

E.4 Possible root, swap and usr Configurations

While installing the Volume Manager, different configurations are possible for root, swap and usr. The following cases are possible:

1. usr is a directory under the root and no separate partition is allocated for it. In this case, usr becomes part of the rootvol when the root disk is encapsulated and put under Volume Manager control.

2. usr is on a separate partition which is on the root disk. In this case, a separate volume will be created for usr partition. vxrootmir will mirror the usr volume on the destination disk.

3. usr is on a separate partition which is not on the root disk. In this case, a volume will be created for the usr partition only if that disk is encapsulated by Volume Manager. Note that in such cases, encapsulating the root disk and having mirrors of the root volume will not help if the usr partition becomes inaccessible for any reason. Therefore it is suggested that you encapsulate the disk containing the usr partition in addition to the root disk and have mirrors for the usr volume along with mirrors for the root and swap volumes for maximum availability of the system.

The Solaris 2.x environment allows you to put swap partitions on any disk; it does not need an initial swap area during early phases of the boot process. By default the Solaris 2.x installation chooses the partition 0 on the selected root disk as the root partition and partition 1 as the swap partition. However, it is possible to have the swap partition on a partition not located on the root disk. In such cases you are advised to encapsulate that disk and create mirrors for the swap volume. If you do not do this, in the event of damage to the swap partition, the system will eventually crash. It may be possible to boot the system but having mirrors for the swap volume will prevent system failures.

E.5 Failures and Recovery Procedures

While there are many types of failures that can prevent a system from booting, the same basic procedure can be taken to bring the system up. When a system fails to boot, the administrator should first try to identify the failure by the evidence left behind on the screen, and repair it if possible (for example, if a drive was accidentally powered off). If the problem cannot be repaired (such as data errors on the boot disk), boot the system from a backup boot disk so that the damage can be repaired or the failing disk replaced.

Some of the possible failures and instructions on the corrective actions are outlined in this section.

E.5.1 Failures In Unix Partitioning

Once the boot program has loaded, it will attempt to access the boot disk through the normal UNIX partition information. If this information is damaged, the boot program will fail with the following error:

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

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

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

This indicates that the failure was due to an invalid disk partition. The administrator can attempt to re-add the disk as described in the section “Re-adding a Failed Boot Disk” on page E-17. However, if the reattach fails, then the disk will need to be replaced as described in the section “Replacing a Failed Boot Disk” on page E-18.

E.5.2 Failures Accessing the Boot Device

Early in the boot process, immediately following system initialization, the screen will look similar to:

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

This means that the system PROM was unable to read the boot program from the boot drive.

If no hardware problems are found, the error is probably due to data errors on the boot disk. In order to repair this problem, attempt to boot the system from a backup boot disk. If you are unable to boot from a backup boot disk, there is still some type of hardware problem. Similarly, if swapping the failed boot disk with a backup boot disk fails to allow the system to boot, this also indicates hardware problems.

E.5.3 Failures Due to Incorrect Entries in `/etc/vfstab`

When the `root` disk is encapsulated and put under Volume Manager control, as part of the normal encapsulation process, volumes are created for all the partitions on the disk. Volume Manager modifies the `/etc/vfstab` to use the corresponding volumes instead of the disk partitions. Care should be taken while editing the `/etc/vfstab` file manually. The most important entries are those corresponding to the `/` and `/usr`. The `vfstab` prior to Volume Manager installation is saved in `/etc/vfstab.prevm`.

E.5.3.1 Damaged `/` Entry in `/etc/vfstab`

In case the entry in `/etc/vfstab` for the `/` is lost or is incorrect, the system will boot in single user mode. The following messages will appear on the console:

```
Hostname: Leo
/sbin/rcS: test: argument expected
/etc/vx/bin/vx-reconfig: /tmp/vx1.33: cannot create
/tmp/vx.133: No such file or directory.
INIT: Cannot create /var/adm/utmp.ar /var/adm/utmpx
INIT: failed write of utmpx entry:” “
INIT: failed write of utmpx entry:” “
INIT: SINGLE USER MODE
Type Ctrl-d to proceed with normal startup
(or give root password for system maintenance):
Entering System Maintenance Mode
```

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

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

After mounting the / read/write, exit the shell. The system will ask for the run level.

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

For multi-user mode, enter the run level as 3. Restore the entry in /etc/vfstab for / after the system boots.

E.5.3.2 Damaged /usr Entry in /etc/vfstab

/etc/vfstab will have entry for /usr only if /usr is mounted on a disk partition. After encapsulation of the disk containing the /usr partition, Volume Manager will change the entry in /etc/vfstab to use the corresponding volume.

In the event of loss of the entry for /usr from /etc/vfstab, the system can not be booted (even if you have mirrors of /usr volume).

In such cases, boot the system from a Solaris CD and restore the /etc/vfstab.

E.5.4 Failures due to Missing or Damaged /etc/system

Do not edit any entry in /etc/system that are added by Volume Manager. All Volume Manager entries are enclosed in * Volume Manager_START and * Volume Manager_END. It is advisable to make a copy of /etc/system in the root file system before making any changes to it. The saved system file can be specified to the boot program if changes to the new /etc/system file are incorrect. To specify the saved system file to the boot program, boot the system with the command boot -a. When the system asks for the name of the system file enter the path of the saved system file.

In case the /etc/system file is damaged and the saved copy of the system file is not available, the system cannot be booted with the Volume Manager rootability feature on. The system may be booted without the Volume Manager rootability (for example, without the rootvol being the /) if /usr is not a volume. After booting the system make the following entries in the /etc/system.

```
* Volume Manager_START
rootdev:/pseudo/vxio@0:0
set vxio:vol_rootdev_is_volume=1
set vxio:vol_swapdev_is_volume=1
* Volume Manager_END
```

Also force load all the drivers required for the root mirror disks. The driver names for these disks could be found in the /devices directory.

To boot the system without the Volume Manager rootability, use the following command and responses:

```
ok boot -a
   .
   .
Rebooting with command: -a
Boot device: /iommu/sbus/espdma/esp/sd@5,0   File and args: -a
Enter filename [/kernel/unix]: <return>
Name of system file [/dev/null]: <return>
Name of default directory for modules [/kernel
/usr/kernel]:<return>
Enter name of device instance number file
[/etc/path_to_inst]:<return>
root file system type [ufs]:<return>
Enter physical name of root device
[/iommu.....................................]:<return>
```

Now the system will boot in single user mode. The following error messages will appear on the console:

```
The / file system (/dev/vx/rdsk/rootvol) is being checked.
Can't open /dev/vx/rdsk/rootvol
/dev/vx/rdsk/rootvol: CAN'T CHECK FILE SYSTEM
/dev/vx/rdsk/rootvol: UNEXPECTED INCONSISTENCY; RUN fsck
MANUALLY.
WARNING - Unable to repair the / filesystem.
Run fsck manually (fsck -F ufs /dev/vx/rdsk/rootvol).
Exit the shell when done to continue the boot process.
```

At this point in the boot process the actual device mounted on / is the physical root device and not the rootvol. Run fsck on the physical root device and remount it read/write.

```
fsck -F ufs /dev/rdsk/c0t3d0s0
mount -o rw,remount /dev/dsk/c0t3d0s0 /
mount: warning: cannot lock temp file </etc/.mnt.lock>
```

Exiting the shell will now resume the boot for multi-user level.

The mount command will still show the / to be mounted on /dev/vx/dsk/rootvol. To correct this, remount the / again.

In the case where the /etc/system file is damaged, but a saved copy of the system file is available, the system can be booted with Volume Manager rootability. To boot the system with the Volume Manager rootability, boot the system with the following command and responses:

```
 ok boot -a
  .
  .
 Rebooting with command: -a
 Boot device: /iommu/sbus/espdma/esp/sd@5,0   File and args: -a
 Enter filename [/kernel/unix]: <return>
 Name of system file [/etc/system.sav]: <return>
 Name of default directory for modules [/kernel
/usr/kernel]:<return>
 Enter name of device instance number file
[/etc/path_to_inst]:<return>
 root file system type [ufs]:<return>
 Enter physical name of root device
 [/iommu....................................]:/pseudo/vxio@0:0
```

With this, the system will boot with Volume Manager rootability.

If the system file is damaged or lost, and a backup copy of the system file is not available, and /usr is a volume, the system must be booted from a Solaris CD. Mount one of the root partitions and edit the system file on it. Enter the following lines in the system file:

```
* Volume Manager_START
rootdev:/pseudo/vxio@0:0
set vxio:vol_rootdev_is_volume=1
set vxio:vol_swapdev_is_volume=1
set vxio:usrport_is_volume=1
* Volume Manager_END
```

Also force load all the drivers required for the root mirror disks. After these changes, reboot the system from the same root partition on which the system file was restored.

E.5.5 Failures Due To Booting From Unusable or Stale Plexes

If a disk is unavailable when the system is running, any mirrors of volumes that reside on that disk will become stale, meaning the data on that disk is out of date relative to the other mirrors of that volume. During the boot process, the system accesses only one copy of the `root` volume (the copy on the boot disk) until a complete configuration for this volume can be obtained. If it turns out that the mirror of this volume that was used for booting is stale, the system must be rebooted from another boot disk that contains non-stale mirrors. This problem can occur, for example, if the system was booted from one of the disks made bootable by Volume Manager with the original boot drive turned off. The system will boot normally, but the mirrors that reside on the unpowered disk will be stale. If the system reboots from the original boot drive with the drive turned back on, the system will boot using that stale plex.

Another possible problem can occur if errors in the Volume Manager headers on the boot disk prevent Volume Manager from properly identifying the disk. In this case, Volume Manager will not know the name of that disk. This is a problem because mirrors are associated with disk names, therefore any mirrors on the unidentified disk are unusable. If either of these situations occurs, the Volume Manager utility `vxconfigd` will notice it when it is configuring system as part of the init processing of the boot sequence. It will display a message describing the error, describe what can be done about it, and halt the system. For example, if the mirror `rootvol-01` of the root volume `rootvol` on disk `disk01` of the system was stale, `vxconfigd` would print the following message:

```
Volume Manager:vxconfigd: Warning Plex rootvol-01 for root volume
is stale or unusable.
Volume Manager:vxconfigd: Error: System boot disk does not have
a valid root plex
Please boot from one of the following disks:
Disk: disk01              Device: c0t1d0s0
Volume Manager:vxconfigd:    Error: System startup failed
The system is down.
```

This informs the administrator that the disk `disk01` contains a usable copy of the `root` mirror and should be used for booting. This is the name of the system backup disk. When this message appears, the administrator should reboot the system from the system backup disk.

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

If the mirrors on the boot disk were unavailable, the administrator should get mail from the Volume Manager utilities describing the problem. Another way to discover the problem is by listing the disks with the `vxdisk` utility. In the above example, if the problem is a failure in the private area of `disk01` (such as due to media failures or accidentally overwriting the Volume Manager private region on the disk), the command:

```
vxdisk list
```

would show the following output:

```
DEVICE      TYPE        DISK        GROUP     STATUS
-           -          disk01       rootdg   failed was: c0t0d0s0
```

E.5.6 Re-adding and Replacing Boot Disks

Normally, replacing a failed disk is as simple as physically replacing the failed drive and running the Volume Manager replace disk commands. It's even possible to move the data areas from that disk to available space on other disks, or to use a "hot spare" disk to replace the failure. For data that is not critical for booting the system, it doesn't matter where the data is located. All data that is not boot critical is only accessed by the Volume Manager after the system is fully operational. The Volume Manager can find this data for you.

On the other hand, boot-critical data must be placed in specific areas on the bootable disks in order for the boot process to find it. The system OpenBoot PROM constrains the location of this data. Therefore, the process of replacing a boot disk is slightly more complex.

When a disk fails, there are two possible routes that can be taken to correct the action. If the error(s) are transient or correctable, then the same disk can be re-used. This is known as *re-adding* a disk. In some cases, actions such as reformatting a failed disk or simply doing a complete surface analysis to rebuild the alternate-sector mappings will be sufficient to make a disk re-usable and thus a candidate for re-addition. On the other hand, if the disk has truly failed, then it should be completely replaced.

E.5.6.1 Re-adding a Failed Boot Disk

Re-adding a disk is actually the same procedure as replacing the disk, except that the same physical disk is used. Normally, a disk that needs to be re-added has been *detached*, meaning that Volume Manager has noticed that the disk has failed and has ceased to access it. For example, take a system that has two disks, `disk01` and `disk02` which are normally mapped into the system configuration during boot as disks `c0t0d0s2` and `c0t1d0s2`, respectively. A failure has occurred to `disk01` that has caused the disk to become detached. This can be confirmed by listing the disks with the `vxdisk` utility, as in:

```
vxdisk list
```

This would result in the following output:

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

Notice that the disk `disk01` has no device associated with it, and has a status of `failed` with an indication of the device that it was detached from. It is also possible that the device `c0t0d0s2` would not be listed at all; this would occur if the disk failed totally and could not communicate with the system.

This is not necessarily always the case. For example, if the boot disk has uncorrectable failures associated with the Solaris partition table (such as a missing root partition that cannot be corrected), the output of the `vxdisk list` command resembles the following:

```
DEVICE      TYPE      DISK      GROUP      STATUS
c0t0d0s2    sliced    disk01    rootdg     online
c0t1d0s2    sliced    disk02    rootdg     online
```

However, because the error was not correctable by the described procedures, the disk is still deemed to have failed. In this case, it is necessary to manually detach the failing disk from its device. You can do this using the "Remove Disk for Replacement" function of the `vxdiskadm` utility (see the `vxdiskadm`(1M) man page for more information).

Once the disk is detached from the device, any special procedures for correcting the problem can be taken, such as reformatting the device.

To re-add the disk, use the "Replace a failed or removed disk" function of the `vxdiskadm` utility to replace the disk, and select the same device as the replacement. In the above example, this would mean replacing `disk01` with the device `c0t0d0s0`.

If a re-add of the disk fails, the physical disk should be replaced.

E.5.6.2 Replacing a Failed Boot Disk

When a boot disk needs to be replaced, the system should first be booted off a backup boot disk and if the failing disk is not detached from its device, it should be manually detached using `vxdiskadm`. Once the disk is detached, the system should be shut down and the hardware replaced.

The replacement disk should have at least as much storage capacity as was in use on the disk being replaced. If the new disk has as much or more space than the old one, it can just be replaced. To determine the minimum size of a replacement disk you need to determine how much space was in use on the disk that failed.

The replacement disk should be large enough such that the region of the disk for storing subdisks will be sufficient to hold all subdisks of the original disk at their current disk offsets.

To approximate the size of the replacement disk, use the command:

```
vxprint -st -e ’sd_disk=”diskname”’
```

Add the values under the `DISKOFFS` and `LENGTH` columns for the last subdisk listed. The total is in 512-byte sectors. Divide the sum by 2 for the total in kilobytes.

Once a replacement disk has been found, shut down the machine cleanly and replace the necessary hardware. The replacement disk should be added at the same location on the bus (for example, for SCSI controllers, the disk should have the same SCSI ID) as the disk being removed.

When the hardware replacement is complete, boot the system and replace the failing disk with the new device that was just added using `vxdiskadm`.

E.5.7 Booting After Failures

To boot the system after failure of the primary boot disk, do the following:

- Check for aliased VM disks using the `devalias` command at the OpenBoot command prompt.
- Disks that are suitable mirrors of the root disk will be listed with the name `vx-`*`medianame`*, where *`medianame`* represents the disk media name for the disk containing the candidate root file system.
- Enter the following:

```
ok boot alias_name
```

  where *`alias_name`* represents the aliased name of the selected disk.

- If a selected disk contains a root mirror that is stale, then `vxconfigd` will display an error stating that the mirror is unusable and giving the list of non-stale alternative disks to boot from.

E.5.8 Repairing root or /usr File Systems on Volumes

If, for some reason, the root or `/usr` file systems become unusable, you should boot from a network-mounted root file system or from the Solaris installation CD-ROM, mount the file system, and repair it. This is necessary, for example, if certain key files involved in booting have been corrupted.

This task is made more difficult when either file system is defined on a mirrored volume, because changes to the partition that underlies one of the mirrors can result in corruption when the volume manager later boots and presumes that the mirrors are reasonably synchronized.

There are two workarounds for this: the simplest workaround, in many situations, is to mount one mirror of the root or `/usr` file system, repair it, and use `dd` to copy the fixed mirror to all other mirrors. However, this could result in errors. Another way, which can also be used if one of the mirrors is striped, non-contiguous, or has no exactly-mapped underlying partition, is to use a set of scripts and files that are provided on the SPARCstorage Volume Manager CD. If you can mount the SPARCstorage Volume Manager CD or make its contents available over the network, then you can run the Volume Manager.

The first step is to make the contents of the `scripts` directory of the SPARCstorage Volume Manager CD available:

1. **Become superuser.**
2. **Insert the SPARCstorage Volume Manager CD into a CD-ROM connected to your system.**
3. **At the prompt, enter the following commands:**

```
# mount -F hsfs -r /dev/dsk/c0t6d0s2 /mnt
# cd /mnt
# ls
```

   You should see a listing of the contents on the SPARCstorage Volume Manager CD.

4. **Setup a special environment for running the Volume Manager using the command:**

```
# . path_to_CD-ROM/name_of_CD/op_environ/scripts/fixsetup
```

   For example:

```
# . /cdrom/ssa_2_0_sparc/2.3_SSA_PKGS/scripts/fixsetup
```

You must run this command from the bourne-shell or K-shell. This script sets environment variables, so if you want to use C-shell you must run `csh` as a subshell. The `fixsetup` command will prompt for the location of the CD-ROM image. In the above example, you would enter **`/cdrom/ssa_2_0_sparc/2.3_SSA_PKGS`**.

5. **After running the `fixsetup` command, you can try to mount the root file system read-only and get the Volume Manager startup files by running the command:**

```
# fixmountroot
```

This command will prompt for a disk containing a mirror of the root file system. For example, if you normally boot from disk 3 on your SCSI controller, you would respond `c0t3d0`. If you were successful in mounting the root file system mirror, this will ask if you want to start up the Volume Manager. You should normally respond with `yes`. It will next ask if you wish to start all volumes in the `rootdg` and other disk groups. You should normally start volumes in the volumes in the `rootdg` disk group, but not in other disk groups.

The root file system mirror will be mounted read-only under the mount point `/tmp/root_mirror`. You can change to that directory and examine the root file system for errors, but you cannot fix the root file system directly under that mount point because it is a read-only mount. If you want to mount and repair the root file system (or the `/usr` file system), then you must mount the volume. For example:

```
# mkdir /tmp/rootvol
# mount /dev/vx/dsk/rootvol /tmp/rootvol
```

When you are done with any repairs, unmount the volume and reboot.

If the `fixmountroot` command fails (for example, if the file system is too corrupted to be mounted), or some of the volume manager startup files are missing from the mounted root file system, then you can start the volume manager directly using the command:

```
# fixstartup
```

This requires that you enter the host identifier that Volume Manager uses for your host (normally the hostname, not the host ID), and that you enter any license keys that are needed for correct operation.

You can use most Volume Manager commands, including the `vxdiskadm` menus while booted in this environment.

E.6 Failures and RAID-5 Volumes

Failures, when taken in the context of volume management, come in two types: *system failures* and *disk failures*. A system failure means that the system has abruptly ceased to operate, such as due to an operating system panic or power failure. Disk failures imply that the data on some number of disks has become unavailable due to some sort of failure in the system (for example, a head crash, electronics failure on disk, disk controller failures).

E.6.1 System Failures

RAID-5 volumes are designed to remain available in the face of disk failures with a minimum of disk space overhead. However, many implementations of RAID-5 can become vulnerable to data loss following a system failure. This occurs because a system failure causes the data and parity in the RAID-5 volume to become out of synchronization because the disposition of writes that were outstanding at the time of the failure cannot be determined. If this occurs while a RAID-5 volume is being accessed, the volume is described as having *stale parity*. When this occurs, the parity must be reconstructed by reading all the non-parity columns within each stripe, recalculating the parity and writing out the parity stripe unit in the stripe. This must be done for every stripe in the volume; thus it can take a long time to complete.

Caution – While this resynchronization is going on, any failure of a disk within the array will cause the data in the volume to be lost. This applies to RAID-5 without log plexes.

Having the array vulnerable in this way is undesirable. Besides the vulnerability to failure, the resynchronization process can tax the system resources and slow down system operation.

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

E.6.2 Disk Failures

Disk failures can cause the data on a disk to become unavailable. In terms of a RAID-5 volume, this would mean that a subdisk becomes unavailable. This can occur due to an uncorrectable I/O error while writing to the disk detected by the kernel, which causes the subdisk to be detached from the array, or could occur if a disk is unavailable when the system is booted (such as from a cabling problem or having a drive powered down). When this occurs, the subdisk cannot be used to hold data and is considered *stale* and *detached*. If the underlying disk becomes available or is replaced, the subdisk will still be considered stale and will not be used.

If an attempt is made to read data contained on a stale subdisk, the data is reconstructed from data from all other stripe elements in the stripe; this operation is called a *reconstructing-read*. This is a significantly more expensive operation than simply reading the data, resulting in degraded read performance. Thus, when a RAID-5 volume has stale subdisks, it is considered to be in *degraded mode*.

A RAID-5 volume in degraded mode can be recognized as such from the output of `vxprint`.

Table E-1 shows the output from `vxprint`:

Table E-1 vxprint Output of Degraded RAID-5 Volumes

V	NAME	USETYPE	KSTATE	STATE	LENGTH	READPOL	PREFPLEX	
PL	NAME	VOLUME	KSTATE	STATE	LENGTH	LAYOUT	NCOL/WDT H	MODE
SD	NAME	PLEX	DISK	DISKOFFS		LENGTH	[COL/]OF FFLAG	S
v	r5vol	RAID-5	ENABLED	DEGRADED	20480	RAID	-	
pl	r5vol-01	r5vol	ENABLED	ACTIVE	20480	RAID	3/16	RW
sd	disk00-00	r5vol-01	disk00	0		10240	0/0	
sd	disk01-00	r5vol-01	disk01	0		10240	1/0	dS
sd	disk02-00	r5vol-01	disk02	0		10240	2/0	-
pl	r5vol-l1	r5vol	ENABLED	LOG	1024	CONCAT	-	RW
sd	disk03-01	r5vol-l1	disk00	10240		1024	0	-
pl	r5vol-l2	r5vol	ENABLED	LOG	1024	CONCAT	-	RW
sd	disk04-01	r5vol-l2	disk02	10240		1024	0	-

The volume `r5vol` is in degraded mode as shown by the state (column 5) being `DEGRADED`. The failed subdisk is `disk01-00` as shown by the flags in the last column: the `d` indicates the subdisk is detached, and the `S` indicates the subdisk contents are stale.

It is also possible that a disk containing a RAID-5 log could experience a failure. This has no direct effect on the operation of the volume; however, the loss of all RAID-5 logs on a volume makes the volume vulnerable to a complete failure as described earlier. A failure within a RAID-5 log plex is indicated by the plex state being `BADLOG` as shown in the following example, where the RAID-5 log plex `r5vol-l1` has failed.

Table E-2 vxprint Output of a RAID-5 Volume with a Failed Log Plex

V	NAME	USETYPE	KSTATE	STATE	LENGTH	READPOL	PREFPLEX	
PL	NAME	VOLUME	KSTATE	STATE	LENGTH	LAYOUT	NCOL/WDTH	MODE
SD	NAME	PLEX	DISK	DISKOFFS		LENGTH	[COL/]OFF	FLAGS
v	r5vol	RAID-5	ENABLED	ACTIVE	20480	RAID	-	
pl	r5vol-01	r5vol	ENABLED	ACTIVE	20480	RAID	3/16	RW
sd	disk00-00	r5vol-01	disk00	0		10240	0/0	-
sd	disk01-00	r5vol-01	disk01	0		10240	1/0	dS
sd	disk02-00	r5vol-01	disk02	0		10240	2/0	-
pl	r5vol-l1	r5vol	ENABLED	BADLOG	1024	CONCAT	-	RW
sd	disk03-01	r5vol-l1	disk00	10240		1024	0	-
pl	r5vol-l2	r5vol	ENABLED	LOG	1024	CONCAT	-	RW
sd	disk04-01	r5vol-l2	disk02	10240		1024	0	-

E.7 Recovery

There are two basic types of recovery needed for RAID-5 volumes: parity resynchronization and stale subdisk recovery. In most cases, these recoveries are performed when the RAID-5 volume is started, shortly after the system boots, or by calling the `vxrecover`(1M) command. For more information on starting RAID-5 volumes, see Section E.8.3, "Starting RAID-5 Volumes," on page E-30.

E.7.1 Parity Recovery

In most circumstances, a RAID-5 array will not have stale parity; it should only occur after all RAID-5 log plexes for the RAID-5 volume have failed, and then only if there is a system failure. Furthermore, even if a RAID-5 volume has stale parity, it is usually taken care of as part of the `volume start` process. However, if a volume without valid RAID-5 logs is started and the process is killed before the volume is resynchronized, the result will be an active volume with stale parity. This can be determined by checking the volume state printed in column 5 of the output of the `vxprint` command, as shown in the following example.

Table E-3 vxprint Output for a Stale RAID-5 Volume

V	NAME	USETYPE	KSTATE	STATE	LENGTH	READPOL	PREFPLEX	
PL	NAME	VOLUME	KSTATE	STATE	LENGTH	LAYOUT	NCOL/WDTH	MODE
SD	NAME	PLEX	DISK	DISKOFFS		LENGTH	[COL/]OFF	FLAGS
v	r5vol	RAID-5	ENABLED	NEEDSYNC	20480	RAID	-	
pl	r5vol-01	r5vol	ENABLED	ACTIVE	20480	RAID	3/16	RW
sd	disk00-00	r5vol-01	disk00	0		10240	0/0	-
sd	disk01-00	r5vol-01	disk01	0		10240	1/0	-
sd	disk02-00	r5vol-01	disk02	0		10240	2/0	-

Note that the state is `NEEDSYNC`, indicating that the parity needs to be resynchronized. The state could also have been `SYNC`, indicating that a synchronization was attempted at start time and that a process should be doing the synchronization. If no such process exists or if the volume is in the `NEEDSYNC` state, a synchronization can be started by hand using the `resync` keyword for the `vxvol` command. For example, to resynchronize the RAID-5 volume in Table E-3, the following command would be used:

```
% vxvol resync r5vol
```

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

For larger volumes, parity regeneration can take a significant amount of time and it is possible that the system could be shut down or crash before the operation is completed. Unless the progress of parity regeneration is somehow kept across reboots, the process would have to start all over again. To avoid this situation, parity regeneration is *checkpointed*, meaning that the offset up to which the parity has been regenerated is saved in the configuration database. The `-o checkpt=size` option controls how often the checkpoint is saved; if not specified, it defaults to the default checkpoint size. Since saving the checkpoint offset requires a transaction, making the checkpoint size too small can significantly extend the time required to regenerate parity. After a system reboot, a RAID-5 volume that has a checkpoint offset smaller than the volume length will start a parity resynchronization at the checkpoint offset.

E.7.2 Recovering Logs After Failures

It is possible that RAID-5 log plexes can become detached due to disk failures, as shown in Table E-1. These RAID-5 logs can be reattached by using the `att` keyword for the `vxplex` command. To reattach the failed RAID-5 log plex shown in Table E-1, the following command should be issued:

```
% vxplex att r5vol r5vol-l1
```

Like parity resynchronization, stale subdisk recovery is usually done at volume start time. However, it is possible that the process doing the recovery may get killed, or that the volume was started with an option to prevent subdisk recovery. It's also possible that the disk on which the subdisk resides was replaced without any recovery operations being performed. In any case, if a subdisk is in need of recovery (as shown in Table E-1) it can be achieved by using the `recover` keyword of the `vxvol` command. For example, to recover the stale subdisk in the RAID-5 volume shown in Table E-1, the following command would be used:

```
% vxvol recover r5vol disk01-00
```

If a RAID-5 volume has multiple stale subdisks to be caught up all at once, calling `vxvol recover` with only the name of the volume will achieve this, for example:

```
% vxvol recover r5vol
```

E.8 Miscellaneous RAID-5 Operations

Many operations exist for manipulating RAID-5 volumes and associated objects. These operations are usually performed by other commands such as `vxassist` and `vxrecover` as part of larger operations, such as evacuating disks, etc. These command line operations should not be necessary for casual usage of the volume manager.

E.8.1 Manipulating RAID-5 logs

RAID-5 logs are represented as plexes of RAID volumes and, as such are manipulated using the `vxplex` command. A RAID-5 log can be added using `vxplex att`:

```
vxplex att r5vol r5log
```

The attach operation can only proceed if the new log's size is large enough to hold an entire stripe's worth of data. If the RAID-5 volume already has logs, the new log must be at least as large as the volume's log length; otherwise, the volume log length is set to the `contig len` of the new log plex.

If the RAID-5 volume is not enabled, the new log is marked as BADLOG and will be enabled when the volume is started though it's contents will be ignored. If the RAID-5 volume is enabled and has other enabled RAID-5 logs, the new log will have it's contents synchronized with the other logs via `ATOMIC_COPY` ioctls. If the RAID-5 volume currently has no enabled logs, the new log is zeroed before it is enabled.

Log plexes can be removed from a volume using the `vxplex dis` command:

```
vxplex dis r5log3
```

If removing the log would leave the volume with less than two valid logs, a warning will be printed and the operation will not be allowed to continue. The operation must be forced by using the `-o force` usage type option.

E.8.2 Manipulating Subdisks

Like other subdisks, subdisks of the RAID-5 plex of a RAID-5 volume are manipulated using the `vxsd` command. Association is done using the `assoc` keyword in the same manner as for striped plexes. For example, to add subdisks at the end of each column of a RAID-5 volume, the following command would be used:

```
vxsd assoc r5vol-01 disk10-01:0 disk11-01:1 disk12-01:2
```

If a subdisk is filling a "hole" in the plex (i.e., some portion of the volume's logical address space is mapped by the subdisk) the subdisk will be considered stale. If the RAID-5 volume is enabled, the association operation will regenerate the data that belongs on the subdisk using `VOL_R5_RECOVER` ioctls; otherwise it is simply marked as stale and will be recovered when the volume is started.

Subdisks can be removed from the RAID-5 plex by using `vxsd dis`:

```
vxsd dis disk10-01
```

Warning – If the subdisk maps a portion of the RAID-5 volume's address space, this places the volume in DEGRADED mode. If this is the case, the `dis` operation will print a warning and must be forced using the `-o force` usage type option. Additionally, if another subdisk in any of the stripes that the subdisk occupies are stale or missing or are being recovered, this operation would render the RAID-5 volume unusable. This also will print a warning and can be forced via `-o force`.

Subdisks can be moved to change the disks which a RAID-5 volume occupies by using `vxsd mv`. For example, if `disk03` needed to be evacuated and `disk22` had enough room by using two portions of it's space, the following command could be used:

```
vxsd mv disk03-01 disk22-01 disk22-02
```

While the command is similar to doing the same for striped plexes, the actual mechanics of the operation are not. In order to do RAID-5 subdisk moves, the old subdisk is removed from the RAID-5 plex and replaced by the new subdisks, which are marked as stale and then caught up using `VOL_R5_RECOVER` operations either by `vxsd` or (if the volume is not active) when the volume is started. *This means that the RAID-5 volume is degraded for the duration of the operation.* Another failure in the stripes involved in the move will make the volume unusable. The RAID-5 volume could also become invalid if the parity of the volume were to become stale.

To avoid these situations, the `vxsd` utility will not allow a subdisk move if:

1. a stale subdisk occupies any of the same stripes as the subdisk being moved;
2. the RAID-5 volume is stopped but was not shut down cleanly (parity is considered stale); or
3. the RAID-5 volume is active and has no valid log areas.

The third case can be overridden using the `-o force` usage type option.

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

E.8.3 Starting RAID-5 Volumes

When a RAID-5 volume is started it can be in one of many states. After a normal system shutdown, the volume should be clean and require no recovery. However, if the volume was not stopped cleanly (such as via `vxvol stop`), it may require some type of recovery when it is started before it can be made available. The following sections outline what actions would be taken under which circumstances.

Under normal circumstances, volumes are started automatically after a reboot and any recovery takes place automatically or is done via the `vxrecover`(1) command.

E.8.4 Unstartable RAID-5 Volumes

A RAID-5 volume is unusable if some part of the RAID-5 plex does not map the volume length. In other words, the RAID-5 plex cannot be sparse in relation to the RAID-5 volume length. The RAID-5 plex does not map a region where two subdisks have failed within a stripe, either because they are stale or because they are built on a failed disk. When this occurs, the `vxvol start` command will return with the following error message:

```
vxvm:vxvol: ERROR: Volume r5vol is not startable; RAID-5 plex
does
not map entire volume length.
```

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

Another possible way that a RAID-5 volume can become unstartable is if the parity is stale and a subdisk becomes detached or stale. This occurs because within the stripes that contain the bad subdisk, the parity stripe unit is invalid (because the parity is stale) *and* the stripe unit on the bad subdisk is also invalid. This situation is shown in Figure E-1.

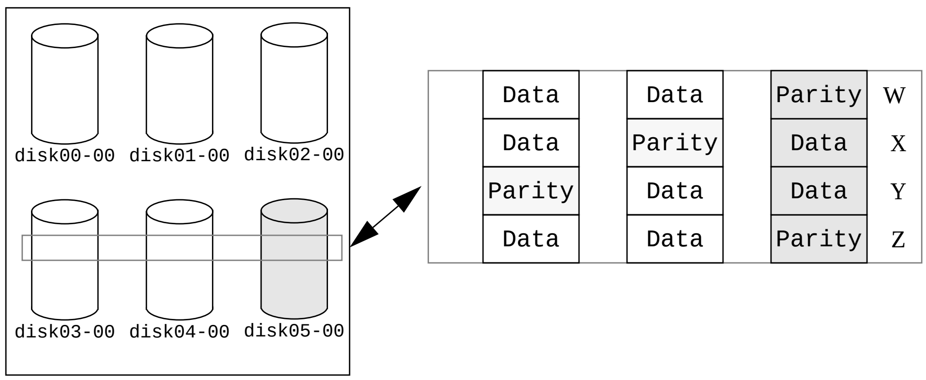

Figure E-1 Invalid RAID-5 Due to Stale Parity and A Failed Subdisk

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

This qualifies as two failures within a stripe and prevents the use of the volume. In this case, the output from `volume start` would be:

```
vxvm:vxvol: ERROR: Volume r5vol is not startable; some subdisks
are unusable and the parity is stale.
```

This situation can be avoided by *always* using RAID-5 log plexes in RAID-5 volumes, preferably two or more. RAID-5 logging plexes prevent the parity within the volume from becoming stale, thus preventing this situation (see “System Failures” on page E-22 for details).

E.8.5 Forcibly Starting RAID-5 Volumes

It is possible that the user may want to start a volume despite the fact that subdisks are marked as stale. For example, if a stopped volume has stale parity and no RAID-5 logs and a disk becomes detached and then reattached, the subdisk is considered stale even though the data is not out of date (because the volume was in use when the subdisk was unavailable) and the RAID-5 volume is considered invalid. The best way to prevent this case is to always have multiple valid RAID-5 logs associated with the array. However, this may not always be possible.

To start a RAID-5 volume despite the presence of stale subdisks, the `-f` option can be used with the `vxvol start` command. This will cause all stale subdisks to be marked as non-stale before the `start` operation begins its evaluation of the validity of the RAID-5 volume and what is needed to start it. Alternately, individual subdisks can be marked as non-stale by using the command `vxmend fix unstale` *`subdisk`*.

E.8.5.1 Recovery When Starting RAID-5 Volumes

Several operations may be necessary to fully restore the contents of a RAID-5 volume and make it usable. Whenever a volume is started, any RAID-5 logging plexes are zeroed before the volume is started. This is done to prevent random data being interpreted as a log entry and corrupting the volume contents. Beyond RAID-5 log zeroing, it is also possible that some subdisks may need to be recovered, or if RAID-5 logs have failed, that the parity be resynchronized. The following steps are taken when a RAID-5 volume is started:

1. If the RAID-5 volume was not cleanly shut down, it is checked for valid RAID-5 log plexes.

    a. If valid log plexes exist, they are replayed. This is done by placing the volume in the `DETACHED` kernel state and setting the volume state to `REPLAY`, and enabling the RAID-5 log plexes. If the logs can be successfully read and the replay is successful, the system will move on to Step 2.

    b. If no valid logs exist, the parity must be resynchronized. Resynchronization is performed by placing the volume in the `DETACHED` kernel state and the volume state is set to `SYNC`. Any log plexes are left `DISABLED`.

    The volume is not made available while the parity is resynchronized because any subdisk failures during this period would make the volume unusable. This can be overridden by using the `-o unsafe` start option.

Caution – This is considered dangerous as it could easily make the contents of the volume unusable and is not recommended.

If any stale subdisks exist, the RAID-5 volume is unusable.

2. Any existing logging plexes are zeroed and enabled. If all logs fail during this process, the start process is aborted.

3. If no stale subdisks exist or those that exist are recoverable, the volume is put in the `ENABLED` kernel state and the volume state is set to `ACTIVE`. The volume is now started.

4. Some subdisks are stale and need recovery, and if valid logs exist, the volume is enabled by placing it in the `ENABLED` kernel state and the volume is available for use during the subdisk recovery. Otherwise, the volume kernel state is set to `DETACHED` and it is not available during subdisk recovery.

   This is done because if the system were to crash or the volume was ungracefully stopped while it was active, the parity would become stale, making the volume unusable. If this is undesirable, the volume can be started with the `-o unsafe` start option.

Caution – This is considered dangerous as it could easily make the contents of the volume invalid and is not recommended.

5. The volume state is set to `RECOVER` and stale subdisks are restored. As the data on each subdisk becomes valid, the subdisk is marked as no longer stale.

   If any subdisk recovery fails and there are no valid logs, the volume start is aborted because the subdisk would remain stale and a system crash would make the RAID-5 volume unusable. This also can be overridden by using the `-o unsafe` start option.

Caution – This is considered dangerous as it could easily make the contents of the volume invalid and is not recommended.

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

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

E.9 Reinstallation Recovery

Occasionally, your system may need to be reinstalled after some types of failures. Reinstallation is necessary if all copies of your root (boot) disk are damaged, or if certain critical files are lost due to file system damage. When a failure of either of these types occurs, you must reinstall the entire system, since there is currently no method of restoring the root file system from backup.

If these types of failures occur, you should attempt to preserve as much of the original Volume Manager configuration as possible. Any volumes not directly involved in the failure may be saved. You do not have to reconfigure any volumes that are preserved.

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

E.9.1 General Recovery Information

System reinstallation completely destroys the contents of any disks that are reinstalled. Any Volume Manager-related information, such as data in the Volume Manager private areas on removed disks (containing the disk identifier and copies of the Volume Manager configuration), is removed during reinstallation. The removal of this information makes the disk unusable as a Volume Manager disk.

The system root disk is always involved in reinstallation. Other disks may also be involved. If the root disk was placed under Volume Manager control (either during Volume Manager installation or by later encapsulation), that disk and any volumes or volume mirrors on it are lost during reinstallation. In addition, any other disks that are involved in the reinstallation (or that are removed and replaced), may lose Volume Manager configuration data (including volumes and mirrors).

If a disk (including the root disk) is not under Volume Manager control prior to the failure, no Volume Manager configuration data is lost at reinstallation. Any other disks to be replaced can be replaced following the procedures in the section “Performing Service-Related Software Tasks.” Although it simplifies the recovery process after reinstallation, not having the root disk under Volume Manager control increases the likelihood of a reinstallation being necessary.

Having the root disk under Volume Manager control, and creating mirrors of the root disk contents, eliminates many of the problems that require system reinstallation.

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

E.9.2 Reinstallation and Reconfiguration Procedures

To reinstall the system and recover the Volume Manager configuration, perform the following procedure, which are described in the sections that follow:

1. **Prepare the system for installation.**
   This includes replacing any failed disks or other hardware, and detaching any disks not involved in the reinstallation.

2. **Install the operating system.**
   Do this by reinstalling the base operating system and any other non-Volume Manager software packages.

3. **Install Volume Manager.**
   Add the Volume Manager package, but *do not* execute the `vxinstall` command.

4. **Recover the Volume Manager configuration.**

5. **Cleanup the configuration.**
   This includes restoring any information in volumes affected by the failure or reinstallation, and recreating system volumes (`root`, `swap`, `usr,` and so on).

E.9.2.1 Preparing the System for Reinstallation

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

Note – Several of the *automatic* options for installation will access drives other than the root drive without requiring confirmation from the administrator. Therefore, you should disconnect all other disks containing volumes from the system prior to installing the operating system.

Disconnecting the other disks ensures that they are unaffected by the reinstallation. For example, if the operating system was originally installed with a home file system on the second disk, it may still be recoverable. Removing the second disk ensures that the home file system remains intract.

E.9.2.2 Reinstalling the Operating System

Once any failed or failing disks have been replaced and disks uninvolved with the reinstallation have been detached, reinstall the operating system as described in the manuals for your operating system. Install the basic operating system prior to installing Volume Manager.

While the operating system installation progresses, make sure no disks other than the root disk are accessed in any way. If anything is written on a disk other than the root disk, the Volume Manager configuration on that disk could be destroyed.

E.9.2.3 Reinstalling the Volume Manager

The installation of the Volume Manager has two parts:

- Loading Volume Manager from CD-ROM
- Initializing the Volume Manager

If you wish to reconstruct the Volume manager configuration left on the non-root disks, *do not* initialize the Volume Manager after the reinstallation.

To reinstall the Volume Manager, follow the instructions for loading the Volume Manager from CD-ROM in Chapter 2, “Initializing the Software.” *Do not* initialize the Volume Manager with the `vxinstall` command.

E.9.2.4 Recovering the Volume Manager Configuration

Once the Volume Manager package has been loaded, recover the Volume Manager configuration by doing the following:

1. **Shut down the system.**
2. **Reattach the disks that were removed from the system.**
3. **Reboot your system.**
4. **When the system comes up, bring the system to single-user mode by entering:**

```
shutdown -g0 -iS -y
```

You will be asked for the system administration password. Enter the password and press Return to continue.

5. **Remove the files involved with installation that were created when you loaded Volume Manager but are no longer needed by entering:**

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

6. **Once these files are removed, you must start some Volume Manager daemons by entering:**

```
vxiod set 10
```

7. **Start the Volume Manager Configuration Daemon, `vxconfigd`, in disabled mode by entering:**

```
vxconfigd -m disable
```

8. **Initialize the `vxconfigd` daemon by entering:**

```
vxdctl init
```

9. **Enable `vxconfigd` by entering:**

```
vxdctl enable
```

The configuration preserved on the disks not involved with the reinstallation has now been recovered. However, because the root disk has been reinstalled, it appears to the Volume Manager as a non-Volume Manager disk. Therefore, the configuration of the preserved disks does not include the root disk as part of the Volume Manager configuration.

If the root disk of your system and any other disks involved in the reinstallation were not under Volume Manager control at the time of failure and reinstallation, then the reconfiguration is complete at this point. If any other disks containing volumes or mirrors are to be replaced, follow the replacement procedures in “Performing Service-Related Software Tasks.” There are several methods available to replace a disk. Choose the method that you prefer.

If the root disk (or another disk) was involved with the reinstallation, any volume or volume mirrors on that disk (or other disks no longer attached to the system) are now inaccessible. If a volume had only one mirror, contained on a disk that was reinstalled, removed, or replaced, then the data in that volume is lost and must be restored from backup. In addition, the system’s `root` file system, `swap` area, and `usr` file system are *not* located on volumes any longer. To correct these problems, follow the instructions in the section “Configuration Cleanup.”

E.9.2.5 Configuration Cleanup

The following sections describe how to clean up the configuration of your system after reinstallation of the Volume Manager.

Rootability Cleanup

To begin the cleanup of the Volume Manager configuration, remove any volumes associated with rootability. This must be done if the root disk (and any other disk involved in the system boot process) was under Volume Manager control. The volumes to remove are:

- `rootvol`, which contains the `root` file system
- `swapvol`, which contains the `swap` area
- `usr`, which contains the `usr` file system

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

```
vxedit -fr rm rootvol
```

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

Volume Cleanup

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

To restore the volumes:

1. Establish which VM disks have been removed or reinstalled by entering:

```
vxdisk list
```

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

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

This display shows that the reinstalled root device, `c0t0d0s0` is not recognized as a VM disk and is marked with a status of `error`. `disk02` and `disk03` were not involved in the reinstallation and are recognized by the Volume Manager and associated with their devices (`c0t1d0s0` and `c0t2d0s0`). The former `disk01`, the VM disk that had been associated with the replaced disk device, is no longer associated with the device (`c0t0d0s0`).

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

2. Once you know which disks have been removed or replaced, all the mirrors on disks with a status of `failed` must be located. Enter:

```
vxprint -sF “%name” -e’sd_disk = “<disk>”’
```

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

3. Check the status of each volume. To print volume information, enter:

```
vxprint -th <volume_name>
```

where *volume_name* is the name of the volume to be examined.

The `vxprint` command displays the status of the volume, its plexes, and the portions of disks that make up those plexes. For example, a volume named `fnah` with only one plex resides on the reinstalled disk named `disk01`. The `vxprint -th` command, applied to the volume `fnah`, produces the following display:

```
DG NAME        GROUP-ID
DM NAME        DEVICE     TYPE       PRIVLEN   PUBLEN   PUBPATH            FLAGS
V  NAME        USETYPE    KSTATE     STATE     LENGTH   READPOL   PREFPLEX
PL NAME        VOLUME     KSTATE     STATE     LENGTH   LAYOUT    NCOL/WIDTH MODE
SD NAME        PLEX       DISK       DISKOFFS  LENGTH   [COL/]OFF FLAGS

v  fnah        fsgen      DISABLED   ACTIVE    24000    SELECT    -
pl fnah-01     fnah       DISABLED   NODEVICE  24000    CONCAT    -          RW
sd disk01-04 fnah-01    disk01     245759    24000    0         -
```

The only mirror of the volume is shown in the line beginning with `pl`. The `STATE` field for the mirror named `fnah-01` is `NODEVICE`. The mirror has space on a disk that has been replaced, removed, or reinstalled. Therefore, the mirror is no longer valid and must be removed. Since `fnah-01` was the only mirror of the volume, the volume contents are irrecoverable except by restoring the volume from a backup. The volume must also be removed. If a backup copy of the volume exists, you can restore the volume later. Keep a record of the volume name and its length, you will need it for the backup procedure.

4. **To remove the volume, use the `vxedit` command. To remove `fnah`:**

```
vxedit -r rm fnah
```

It is possible that only part of a mirror is located on the failed disk. If the volume has a striped mirror associated with it, the volume is divided between several disks. For example, the volume named `foo` has one striped mirror, striped across three disks, one of which is the reinstalled disk `disk31`. The output of the `vxprint -th` command for `foo` returns:

```
DG NAME        GROUP-ID
DM NAME        DEVICE     TYPE     PRIVLEN  PUBLEN    PUBPATH             FLAGS
V  NAME        USETYPE    KSTATE   STATE    LENGTH    READPOL   PREFPLEX
PL NAME        VOLUME     KSTATE   STATE    LENGTH    LAYOUT    NCOL/WIDTH MODE
SD NAME        PLEX       DISK     DISKOFFS LENGTH    [COL/]OFF FLAGS

v  foo         fsgen      DISABLED ACTIVE   10240      SELECT    -
pl foo-01      foo        DISABLED NODEVICE 10240      STRIPE    -         RW
sd pdisk33-07  foo-01     pdisk33  424144   10240      0         -
sd pdisk32-07  foo-01     pdisk32  620544   10240      0         -
pl foo-02      foo        DISABLED NODEVICE 10240      CONCAT    -         RW
sd pdisk31-08  foo-01     pdisk31  262144   10240      0         -
```

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

5. **Use the `vxedit` command to remove the volume, as described earlier.**
A volume that has one mirror on a failed disk may also have other mirrors on disks that are still valid. In this case, the volume does not need to be restored from backup, since the data is still valid on the valid disks. The

output of the vxprint -th command for a volume with one plex on a failed disk (pdisk31) and another plex on a valid disk (pdisk32) would look like this:

```
DG NAME        GROUP-ID
DM NAME       DEVICE    TYPE     PRIVLEN  PUBLEN   PUBPATH                 FLAGS
V  NAME        USETYPE    KSTATE    STATE     LENGTH    READPOL      PREFPLEX
PL NAME        VOLUME     KSTATE    STATE     LENGTH    LAYOUT       NCOL/WIDTH MODE
SD NAME        PLEX       DISK      DISKOFFS LENGTH     [COL/]OFF    FLAGS

v  foo         fsgen      DISABLED ACTIVE   10240      SELECT     -
pl foo-01      foo        DISABLED ACTIVE 10240      STRIPE     -        RW
sd pdisk32-07 foo-01      pdisk32  620544   10240      0          -
pl foo-02      foo        DISABLED NODEVICE 10240      CONCAT     -        RW
sd pdisk31-08 foo-02      pdisk31  262144   10240      0          -
```

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

6. **To remove an invalid plex, the plex must be dissociated from the volume and then removed.**
   This is done with the vxplex command. To remove the plex foo-02, enter:

```
vxplex -o rm dis foo-02
```

7. **Once all the volumes have been cleaned up, you must clean up the disk configuration as described in the section "Disk Cleanup."**

Disk Cleanup

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

To remove the disk, use the `vxdg` command. To remove the failed `disk01`, enter:

```
vxdg rmdisk disk01
```

If the `vxdg` command returns an error message, some invalid volume mirrors exist. Repeat the processes described in "Volume Cleanup" on page E-39 until all invalid volumes and volume mirrors are removed.

E.9.2.6 *Rootability Reconfiguration*

Once all the invalid disks have been removed, the replacement or reinstalled disks can be added to Volume Manager control. If the root disk was originally under Volume Manager control (the `root` and `stand` file systems and the `swap` area were on volumes), or you now wish to put the root disk under Volume Manager control, add this disk first.

To add the root disk to Volume Manager control, use the Volume Manager Support Operations (`vxdiskadm`). Enter:

```
vxdiskadm
```

and select menu item `2`, `Encapsulate a disk`. Follow the instructions and encapsulate the root disk for the system. For more information see the `vxdiskadm`(1M) man page.

When the encapsulation is complete, reboot the system to multi-user mode.

E.9.2.7 *Final Reconfiguration*

Once the root disk is encapsulated, any other disks that were replaced should be added using `vxdiskadm`. If the disks were reinstalled during the operating system reinstallation, they should be encapsulated; otherwise, simply add them.

Once all the disks have been added to the system, any volumes that were completely removed as part of the configuration cleanup can be recreated using their contents restored from backup. The volume recreation can be done using `vxassist` or the GUI.

To recreate the volumes fnah and foo using the vxassist command, enter:

```
vxassist make fnah 24000 vxassist make foo 4224 layout=stripe
nstripe=3
```

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

Any volumes that had plexes removed as part of the volume cleanup can have these mirrors recreated following the instructions for mirroring a volume for the interface (vxassist or GUI) you choose in “Changing Components of the Software Configuration.”

To replace the plex removed from the volume foo using vxassist, enter:

```
vxassist mirror foo
```

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

Index

D

E

F

N

O

P

Q

R

W

X

Glossary

analysis

The method used by the GUI to monitor Volume Manager disk, subdisk, and volume activity.

associate

The process by which a relationship is established between Volume Manager objects. For example, a subdisk that has been created and defined as having a starting point within a plex is referred to as being *associated* with that plex.

associated plex

A plex associated with a volume.

associated subdisk

A subdisk associated with a plex.

button

Either a button on a mouse or a “button” on a window or menu.

cascading menu

A sub-menu accessed through menu options from the pull-down menus. An arrow at the end of a menu selection indicates the availability of a cascading menu.

check box

The icon used to indicate or change the setting of optional control (for example, default settings). The check box is shaded or contains a check mark to indicate its selection.

concatenation

A layout style characterized by subdisks that are sequentially and contiguously arranged within the plex address space.

configuration database

A set of records containing detailed information on existing Volume Manager objects (such as disk and volume attributes).

cylinder

The iconic representation of disks or volumes within the GUI.

detached plex

A plex that is inaccessible for reads and writes, but is still associated with the volume. The last plex in a volume is not allowed to be detached.

dirty region logging

The procedure by which the Volume Manager monitors and logs modifications to a plex. A bitmap of changed regions is kept in an associated subdisk called a *log subdisk*.

disk

A collection of read / write data blocks that are indexed and can be quickly and randomly accessed. Each disk on a system is given a unique ID that can identify the disk, even if it is moved.

disk access name

The device name or address used to access a physical disk, such as `c0t0d0s0`. The `c#t#d#s#` syntax identifies the controller, target address, disk, and partition. The term *device name* can also refer to the disk access name.

disk access record

A configuration record used to specify the access path to a particular disk. Each disk access record contains a name and type, and may include some type-specific information.

disk group

A collection of disks that share a common configuration. A disk group configuration is a set of records containing detailed information on existing Volume Manager objects (such as disk and volume attributes) and their relationships. Each disk group has an administrator-assigned name and an internally defined unique ID.

disk media name

An administrative name chosen for the disk, such as `disk03`. The term *disk name* can also be used to refer to the disk media name.

disk media record

A configuration record that identifies a particular disk by an administrative name and a unique ID.

dissociate

The process by which any link that exists between two Volume Manager objects is removed. For example, dissociating a subdisk from a plex removes the subdisk from the plex and adds the subdisk to the free space pool.

dissociated plex

A plex dissociated from a volume.

dissociated subdisk

A subdisk dissociated from a plex.

drag and drop

The icon manipulation method whereby the icon is selected using the LEFT mouse button, moved to another location in the configuration, and dropped there by releasing the mouse button.

file system

A collection of files organized together into a structure. The UNIX file system is a hierarchical structure consisting of directories and files.

free space

An area of a disk under Volume Manager control that is not allocated to any volume or reserved for use by any Volume Manager object.

free subdisk

A subdisk that is not associated with any plex and has an empty putil[0] field.

icon

The graphical representation of the GUI system configuration entities.

iconify

The act of turning a window into an icon, or changing the shape and view of a GUI object icon.

log subdisk

A subdisk to log recent disk activity using a dirty region log. In the GUI, log subdisks are represented iconically as rectangles with double borders.

menu bar

The rectangular area located at the top of the window that contains the menu options for that window.

mirror

A duplicate copy of a volume and the data therein. There can be several mirrors per volume. In the GUI, mirrors are represented iconically as rectangles with heavy borders. The terms *mirror* and *plex* are used synonymously.

mirroring

A layout technique that mirrors the contents of a volume onto multiple plexes. Each plex duplicates the data stored on the volume, but the plexes themselves may have different layouts.

object

An entity that is defined to and recognized internally by the Volume Manager. The Volume Manager objects are: volume, plex, subdisk, disk, and disk group.

partition

The standard division of a logical disk device. In the GUI, partitions are represented iconically as rectangles within physical disk icons. The terms *partition* and *slice* are sometimes used synonymously.

physical disk

A disk that is not currently under Volume Manager control and assigned to a disk group. This might be a newly installed disk or a disk that has been removed from Volume Manager access. In the GUI, physical disks are represented iconically as cylinders labeled `PD`.

plex

An ordered collection of from one to 256 subdisks. Each plex is one copy of the volume with which it is associated. In the GUI, plexes are represented iconically as rectangles with heavy borders. The terms plex and mirror are used synonymously.

pop up

To open a window.

projection

The method used to illustrate the relationships between icons representing Volume Manager objects. When a particular icon is selected for projection, all icons related to it are highlighted (either with color or a bitmap pattern).

pull-down menu

The menu selections accessed through choices located in the title bar of each window.

radio buttons

A set of buttons, only one of which can be selected at any given time. These buttons toggle on or off.

RAID

A Redundant Array of Inexpensive Disks (RAID) is a disk array set up with part of the combined storage capacity used for storing duplicate information about the data stored in that array. This makes it possible to regenerate the data if a disk failure occurs.

sector

A unit of size, which can vary between systems. A sector is commonly 512 bytes.

select-operate

The icon manipulation method that generally begins by selecting an icon using the LEFT or MIDDLE mouse button. The LEFT button is then used to access the available menu options, and an operation is applied to the selected object(s).

slice

The standard division of a logical disk device. The terms *partition* and *slice* are sometimes used synonymously.

spanning

A layout technique that permits a file system or database too large to fit on a single disk to span across multiple physical disks.

stripes

Relatively small, equally-sized areas that are allocated alternately to the subdisks of a striped plex.

striping

A layout technique that spreads data across several physical disks using stripes. The data is allocated alternately to the stripes within the subdisks of a plex.

subdisk

A consecutive set of contiguous disk blocks that form a logical disk segment. Subdisks are associated with plexes to form volumes. In the GUI, subdisks are represented iconically as small rectangles within VM disk or plex icons.

surface analysis

A form of disk analysis that scans a disk to identify bad tracks or sectors.

VM disk

A disk that is under Volume Manager control and assigned to a disk group. VM disks are sometimes referred to as *Volume Manager disks* or simply *disks*. In the GUI, VM disks are represented iconically as cylinders labeled `D`.

view window

A special GUI window that display icons representing all or a subset of the objects currently known to the GUI. View windows permit the manipulation of physical and logical views of the mass storage subsystem. A set of default views always exists; users have the option of creating additional user-created views.

views subwindow

A smaller window within the GUI root window that contains the view buttons.

GUI root window

The GUI window through which the user accesses views and menu options. The GUI root window is the first interactive window that the user sees when the GUI is started.

volume

A virtual disk, which represents an addressable range of disk blocks used by applications such as file systems or databases. A volume can be composed of from one to eight plexes. In the GUI, volumes are represented iconically as cylinders labeled `V`.

window menu button

The box generally located in the upper left-hand corner of a window that controls the physical properties of that window.

Reader Comments

We welcome your comments and suggestions to help improve this manual. Please let us know what you think about the *SPARCstorage Array User's Guide*, part number *802-2042-10*.

- The procedures were well documented.

Strongly Agree	Agree	Disagree	Strongly Disagree	Not Applicable
❑	❑	❑	❑	❑

Comments ____________________

- The tasks were easy to follow.

Strongly Agree	Agree	Disagree	Strongly Disagree	Not Applicable
❑	❑	❑	❑	❑

Comments ____________________

- The illustrations were clear.

Strongly Agree	Agree	Disagree	Strongly Disagree	Not Applicable
❑	❑	❑	❑	❑

Comments ____________________

- The information was complete and easy to find.

Strongly Agree	Agree	Disagree	Strongly Disagree	Not Applicable
❑	❑	❑	❑	❑

Comments ____________________

- Do you have additional comments about the *SPARCstorage Array User's Guide*?

____

____

____

____

Name: ____________________

Title: ____________________

Company: ____________________

Address: ____________________

____

Telephone: ____________________

Email address: ____________________

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS MAIL PERMIT NO. 1 MOUNTAIN VIEW, CA

POSTAGE WILL BE PAID BY ADDRESSEE

SUN MICROSYSTEMS, INC.
Attn: Manager, Hardware Publications
MS MPK 14-101
2550 Garcia Avenue
Mt. View, CA 94043-9850