sgi

COPAN™ MAID for DMF Quick Start Guide

New Features in this Guide

This revision contains clarifications.

Record of Revision

Version	Description
001	November 2011 Original publication
002	April 2012 Revised with ISSP 2.6
003	April 2013 Revised with ISSP 3.0

Contents

Figures

Tables

About This Guide

This guide tells you how to configure an SGI® COPAN™ native massive array of idle disks (MAID) system for use with the SGI InfiniteStorage™ Data Migration Facility (DMF) and the SGI OpenVault™ mounting service.

Related Publications

The following publications contain additional information:

- *DMF 6 Administrator Guide for SGI InfiniteStorage*

- *DMF 5 Filesystem Audit Guide for SGI InfiniteStorage*

- *High Availability Guide for SGI InfiniteStorage*

- *OpenVault Administrator Guide for SGI InfiniteStorage*

- *SGI InfiniteStorage Software Platform* release note (the README.txt file, see below to obtain)

- *XVM Volume Manager Administrator Guide*

Also see the *Platform Service Manager Online Help* available from the PSM **Help** menu.

Obtaining Publications

You can obtain SGI documentation as follows:

- See the SGI Technical Publications Library at http://docs.sgi.com. Various formats are available. This library contains the most recent and most comprehensive set of online books, man pages, and other information.

- You can view man pages by typing man *title* at a command line.

- The /docs directory on the ISSP DVD or in the Supportfolio™ download directory contains the following:

  - The ISSP release note: /docs/README.txt

  - DMF release notes: /docs/README_DMF.txt

– A complete list of the packages and their location on the media:
`/docs/RPMS.txt`

– The packages and their respective licenses: `/docs/PACKAGE_LICENSES.txt`

- The release notes and manuals are provided in the `noarch/sgi-isspdocs` RPM and will be installed on the system into the following location:

`/usr/share/doc/packages/sgi-issp-`*ISSPVERSION*/*TITLE*

Conventions

The following conventions are used throughout this document:

Convention	Meaning
`command`	This fixed-space font denotes literal items such as commands, files, routines, path names, signals, messages, and programming language structures.
variable	Italic typeface denotes variable entries and words or concepts being defined.
user input	This bold, fixed-space font denotes literal items that the user enters in interactive sessions. (Output is shown in nonbold, fixed-space font.)
[ ]	Brackets enclose optional portions of a command or directive line.
...	Ellipses indicate that a preceding element can be repeated.
`manpage`(*x*)	Man page section identifiers appear in parentheses after man page names.
GUI	This font denotes the names of graphical user interface (GUI) elements such as windows, screens, dialog boxes, menus, toolbars, icons, buttons, boxes, fields, and lists.

Reader Comments

If you have comments about the technical accuracy, content, or organization of this publication, contact SGI. Be sure to include the title and document number of the publication with your comments. (Online, the document number is located in the front matter of the publication. In printed publications, the document number is located at the bottom of each page.)

You can contact SGI in any of the following ways:

- Send e-mail to the following address:

  techpubs@sgi.com

- Contact your customer service representative and ask that an incident be filed in the SGI incident tracking system.

- Send mail to the following address:

  SGI
  Technical Publications
  46600 Landing Parkway
  Fremont, CA 94538

SGI values your comments and will respond to them promptly.

Introduction

This guide provides an example of how to configure an SGI® COPAN™ native massive array of idle disks (MAID) system for use with the SGI InfiniteStorage™ Data Migration Facility (DMF) and the OpenVault™ mounting service. The procedures in this guide use the sample following DMF configuration files, making as few modifications as possible:

- `dmf.conf.copan_maid`, for using COPAN MAID as permanent secondary storage.

- `dmf.conf.fmc`, for using COPAN MAID as a fast-mount cache in conjunction with at least one permanent migration target. The example file specifies physical tape as the permanent migration target, but you could use another target.

You can divide the COPAN MAID resource into one set of volume groups (VGs) for permanent storage and another set of VGs for fast-mount cache, but that is not directly addressed by this guide.

Note: SGI always recommends that you migrate at least two copies to permanent storage targets in order to prevent data loss in the event that a migrated copy is lost. When using a fast-mount cache, SGI therefore recommends that you migrate at least three copies (one to the temporary cache and two to permanent storage targets).

This guide documents the supported procedures for configuring DMF with COPAN MAID. If you deviate from these procedures, DMF may not function.

This chapter discusses the following:

- "DMF in a COPAN MAID Environment" on page 2
- "Preliminary Requirements" on page 16
- "Overview of the Steps in this Guide" on page 20

For complete details about DMF and its configuration, see the *DMF 6 Administrator Guide for SGI InfiniteStorage* and the `ov_copan`(8) man page.

DMF in a COPAN MAID Environment

This section discusses the following:

- "System Overview" on page 2
- "DMF Overview" on page 5
- "RAID Sets" on page 11
- "Power Budget" on page 11
- "Shelf Management" on page 12
- "Backups and DMF" on page 12
- "Formatting the RAID Sets" on page 12
- "Naming Conventions" on page 13
- "Configuring OpenVault" on page 14
- "Summary of the Completed Configuration" on page 14

System Overview

A COPAN cabinet has up to eight MAID shelves, shelf 0 (bottom) through shelf 7 (top). A site can have multiple cabinets with differing numbers of shelves. Figure 1-1 shows a conceptual drawing of DMF using COPAN MAID with 8 shelves as permanent storage.
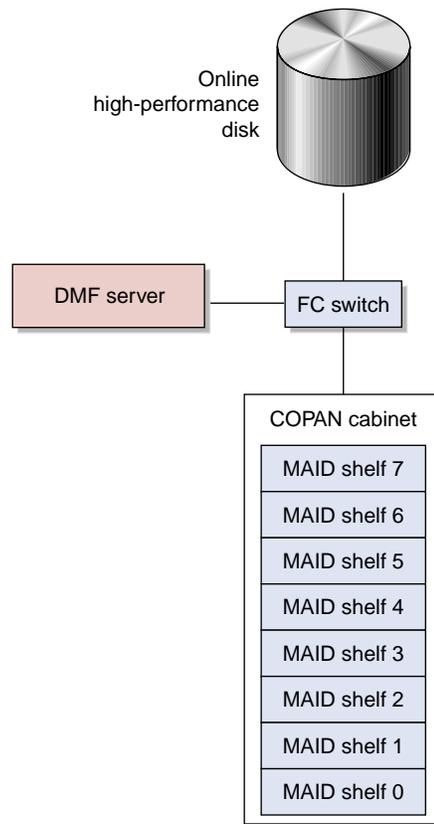
Figure 1-1 DMF with COPAN MAID as Permanent Storage

If you use COPAN MAID in a fast-mount cache configuration, you would also need a permanent migration target, such as physical tape. Figure 1-2 shows a conceptual drawing of DMF using COPAN MAID with 4 shelves as fast-mount cache.
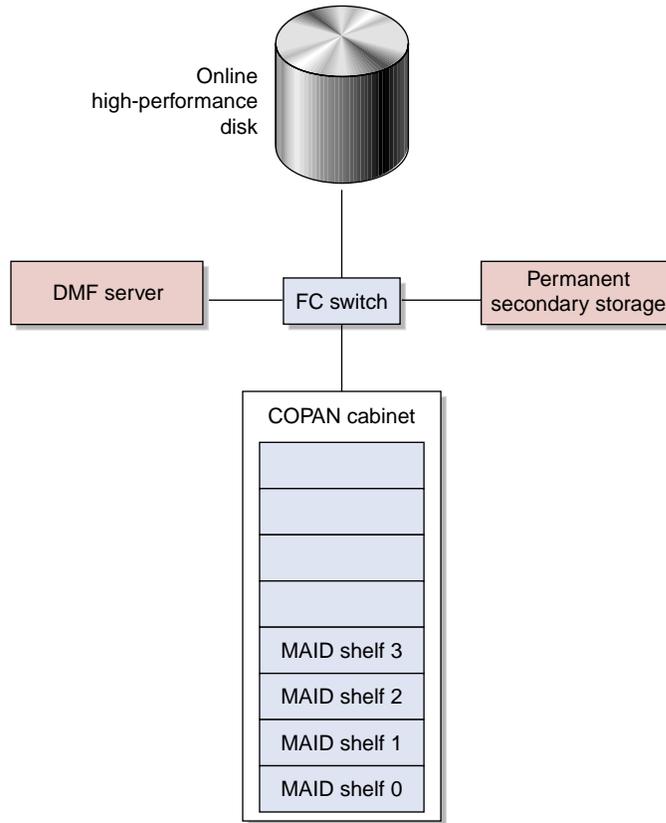
Figure 1-2 DMF with COPAN MAID as a Fast-Mount Cache

For users with higher throughput requirements, the *Parallel Data Mover Option* allows additional data movers on dedicated nodes to operate in parallel with the integrated data mover functionality on the DMF server, increasing data throughput and enhancing resiliency. The dedicated function of a *parallel data mover node* is to move data from the filesystem to secondary storage or from secondary storage back into the primary filesystem. Offloading the majority of I/O from the DMF server improves I/O throughput performance.

DMF Overview

When DMF migrates or recalls files, it issues a mount request to the OpenVault mounting service. OpenVault mounts the correct COPAN MAID storage filesystem and DMF performs I/O to the appropriate volume on that filesystem.

DMF continuously monitors the managed user filesystem according to the policies established in the DMF configuration file. Only the most timely data resides on the higher performance primary filesystem; less timely data is automatically migrated to the secondary storage. However, data always appears to be online to end users and applications, regardless of its actual location.

DMF moves data to secondary storage on the COPAN MAID disk, but it leaves critical metadata (such as index nodes, or *inodes*, and directories) in the primary filesystem. A user retrieves a file simply by accessing it normally through NFS; DMF automatically recalls the file's data from the secondary storage, caching it on the primary filesystem. From a user's perspective, all content is visible all of the time.

You can use COPAN MAID as either of the following:

- "COPAN MAID as Permanent Storage" on page 5
- "COPAN MAID as Fast-Mount Cache" on page 8

COPAN MAID as Permanent Storage

Figure 1-3 and Figure 1-4 show the concepts of migrating and recalling data when using COPAN MAID as permanent storage.

The figures describe the migration concept, showing that data is removed from the primary filesystem after migrating (represented by the dashed lines) and moved to the storage on MAID, but the inode remains in place in the primary filesystem. During the period when the data has been copied to the secondary storage on MAID but has not yet been deleted from the primary filesystem, the file is considered to be *dual-state*. After the data has been deleted from the primary filesystem, the file is considered to be *offline*. SGI recommends that you migrate two copies of a file to prevent data loss in the event that a migrated copy is lost.

Note: For simplicity, these diagrams do not address a second copy. Data will be recalled from a second copy only if necessary.

Before migrating (normal file)

After migrating (dual-state file)
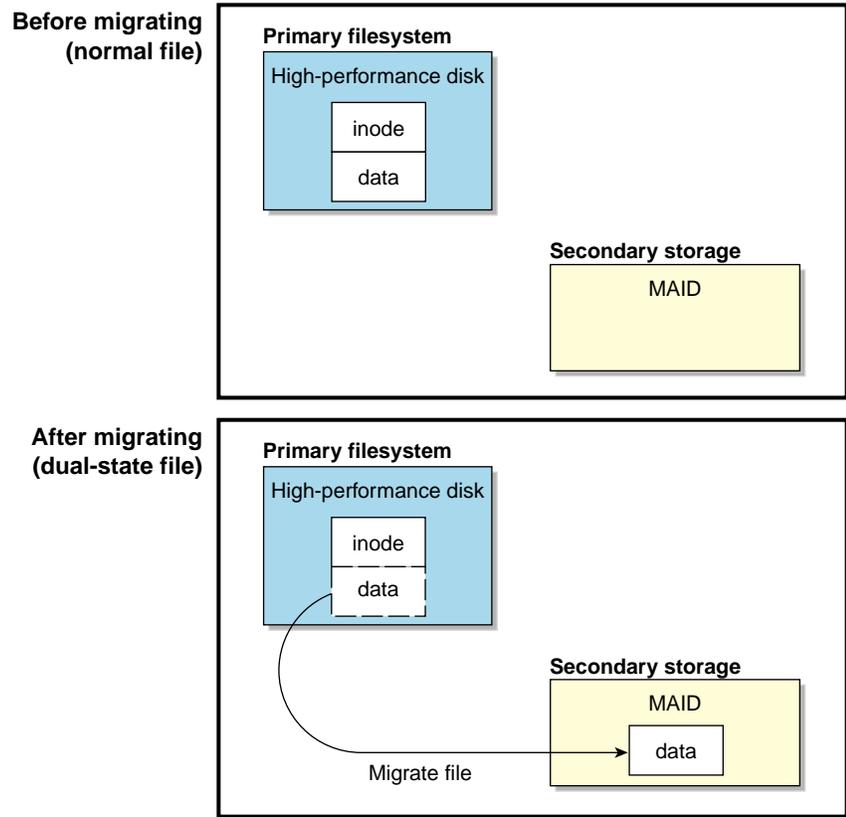
Figure 1-3 COPAN MAID as Permanent Storage: Migrating Data
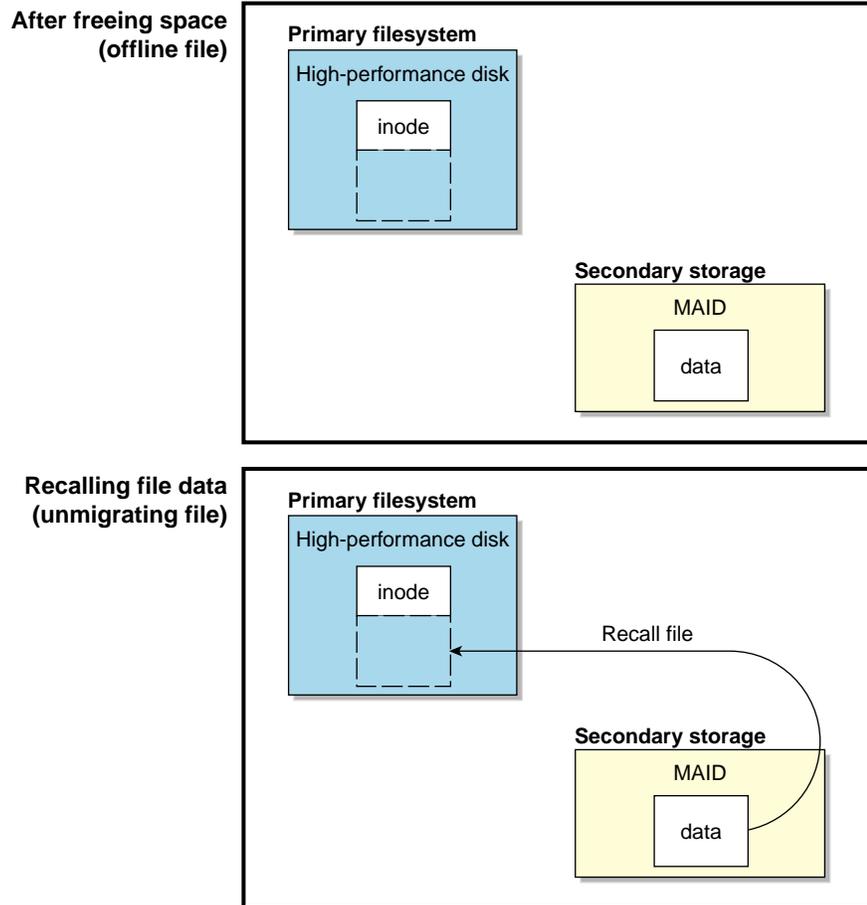
Figure 1-4 COPAN MAID as Permanent Storage: Freeing and Recalling Data

Because the inodes and directories remain online, users and applications never need to know where the data actually resides; migrated files remain cataloged in their original directories and are accessed as if they were still online. In fact, when drilling into directories or listing their contents, a user cannot determine whether a file is online or offline; determining the data's actual residence requires special commands or command options. The only difference users might notice is a slight delay in access time. Therefore, DMF allows you to oversubscribe your online disk in a manner that is transparent to users.

COPAN MAID as Fast-Mount Cache

The *fast-mount cache* configuration migrates data simultaneously to COPAN MAID as a temporary cache and to permanent storage on another migration target (such as physical tape). Volumes on the cache can be freed immediately when the fullness threshold is reached. SGI always recommends that you migrate at least two copies to permanent storage targets in order to prevent data loss in the event that a migrated copy is lost. SGI therefore recommends that you migrate at least three copies for this configuration (one to the temporary cache and two to permanent storage targets).

Figure 1-5 and Figure 1-6 show the concepts of migrating and recalling data when using COPAN MAID as a fast-mount cache in conjunction with permanent storage, in this case on physical tape.

Note: For simplicity, these diagrams do not address a second permanent storage copy. Data will be recalled from a second permanent storage copy only if necessary.
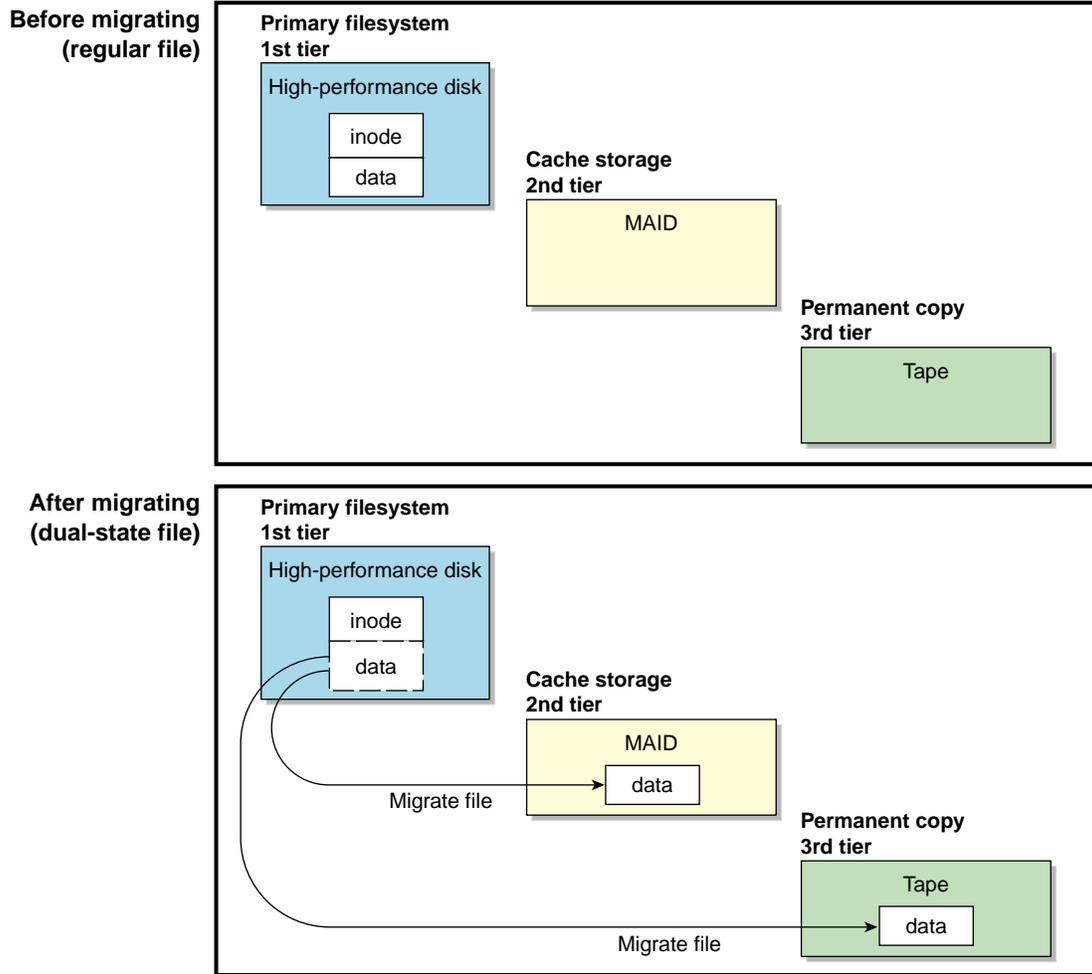
Figure 1-5 COPAN MAID as a Fast-Mount Cache: Migrating Data

After freeing space (offline file)

Primary filesystem
1st tier

High-performance disk

inode

Cache storage
2nd tier

MAID

data

Permanent copy
3rd tier

Tape

data

Recalling file data from cache (unmigrating file)

Primary filesystem
1st tier

High-performance disk

inode

Cache storage
2nd tier

MAID

Recall file

data

Permanent copy
3rd tier

Tape

data

Recalling file data after freeing the cache volume (unmigrating file)

Primary filesystem
1st tier

High-performance disk

inode

Cache storage
2nd tier

MAID

Permanent copy
3rd tier

Tape

Recall file

data

Figure 1-6 COPAN MAID as a Fast-Mount Cache: Freeing and Recalling Data

RAID Sets

COPAN MAID uses *redundant array of independent disks (RAID)* technology. Each MAID shelf has a shelf controller and 27 groups of disks; each group is referred to as a *RAID set*:

- The first 26 RAID sets each appear to the DMF server and any parallel data mover nodes as a SCSI disk device *logical unit (LUN)*. Each RAID set has three data disks plus one parity disk (known as *RAID 5*). These RAID sets are named A-Z, and the associated LUNs are numbered 1–26. (LUN 0 is the shelf controller.)

- The 27th RAID set contains the *always on region* (AOR), which consists of specially mapped ranges of blocks from other host-accessible LUNs (see "Contact SGI to Apply the AOR Files" on page 16).) This RAID set is not presented to the host as a LUN and is not consumed by DMF during data migration.

Figure 1-7 shows a conceptual example of a MAID shelf.
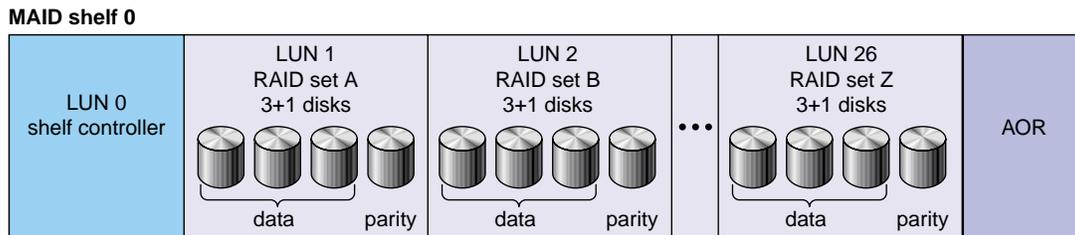
MAID shelf 0

Figure 1-7 MAID Shelf Contents

After you complete the procedures in this guide, each RAID set A-Z will be formatted with an XFS filesystem. Large files, known as *volumes*, will be preallocated on the filesystem. From the point of view of DMF and OpenVault, these volumes are the logical equivalent of a tape cartridge (and therefore are sometimes referred to as *tapes* in command output and documentation.)

Power Budget

COPAN MAID contains power-management features that spin down and power off the least-recently-used individual RAID sets after a period of disuse or when a fixed power budget is exceeded.

DMF carefully controls how many RAID sets are accessed at once and groups migrate/recall requests in order to move data efficiently. If data is required from a disk that is powered off, the drives in the RAID set that contains the data being requested will be powered on; if necessary, the drives in another RAID set will be powered off so that the power budget is maintained.

Shelf Management

Each MAID shelf is managed by one system at a time, either the DMF server or one of the DMF parallel data mover nodes.

Note: If you want to enable failover of this management from one system to another, or failover of the DMF server capability, you must install the required HA products and configure an HA cluster for use with SGI resource agents. For more information, see *High Availability Guide for SGI InfiniteStorage.*

Backups and DMF

DMF does not provide backup services for the primary disks, but instead provides a migration service for data. DMF moves only the data associated with files, not the file inodes or directories. Therefore, you must still perform regular backups to protect files that have not been migrated, as well as inodes and directory structures. You must store the backups on physical tape or in a filesystem that is not managed by DMF, such as on a RAID set that has been formatted for non-DMF use.

Formatting the RAID Sets

You will use the ov_copan(8) command to format the RAID sets for DMF-managed data or (optionally) for backups.

Note: If you wish to use the COPAN MAID for third-tier permanent storage as well as second-tier fast-mount cache storage, you can format one or more RAID sets for a filesystem (that is not managed by DMF) that can then be used for backups. For more information about tiers, see the *DMF 6 Administrator Guide for SGI InfiniteStorage.*

The ov_copan command divides each RAID set into two regions (one of which is usually of zero size):

- The *DMF region*, which consists of a single XFS filesystem. (This filesystem is reserved exclusively for use by DMF, do not modify it directly.)

- The *non-DMF region*, on which you can optionally make an XFS filesystem for storing DMF backups, using `mkfs.xfs`(8) and `xvm`(8).

See Chapter 2, "Format the RAID sets and Create Volumes" on page 21.

Naming Conventions

DMF uses a *volume serial number (VSN)* to uniquely identify a specific volume at your site. A given volume is contained within a single RAID set. The set of volumes may be scattered across any number of RAID sets, MAID shelves, and COPAN cabinets. All VSNs that are managed by a single OpenVault instance must be unique.

When used with COPAN MAID, the VSN has six characters and a specific structure. The following string represents the components of the COPAN MAID VSN:

CCSRVV

where:

- *CC* is the *cabinet identifier* that uniquely represents a cabinet at your site. This guide uses C0 (for "Cabinet 0"). You may use a different naming convention if you prefer, using two uppercase alphanumeric characters (that is, `[0-9][A-Z][0-9][A-Z]`; see "Selecting Appropriate Cabinet Identifiers" on page 21).

- *S* represents the shelf position in the cabinet. There are up to eight shelves in a COPAN cabinet, shelf 0 (bottom) through shelf 7 (top). The three-character string *CCS* (such as C01) is called the *shelf identifier* and can be used as an argument in some OpenVault commands.

- *R* represents the RAID set. Each shelf contains 26 RAID sets available for DMF use, named A through Z. The four-character string *CCSR* (such as C01B) is called the *RAID-set identifier* and can be used as an argument in some OpenVault commands.

- *VV* represents an individual volume within a given RAID set. Each RAID set accommodates up to 1,296 volumes, named 00 through ZZ.

For example, the VSN C01B03 identifies the fourth volume (03) on the second RAID set (B) of shelf 1 in COPAN cabinet 0, as shown in Figure 1-8.

Cabinet
[0-9A-Z][0-9A-Z]

Shelf position
[0-7]

RAID set
[A-Z]

Volume on the RAID set
[0-9A-Z] [0-9A-Z]

VSN   C01B03

RAID-set identifier   C01B03

Shelf identifier   C01B03
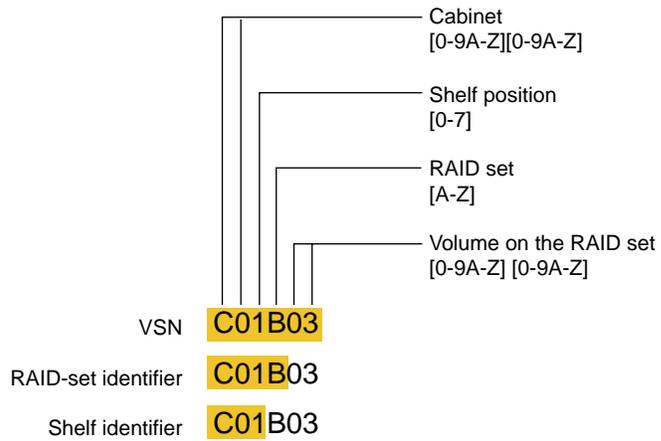
Figure 1-8 VSN Components for COPAN MAID

Configuring OpenVault

You will use the ov_admin(8) command to add the DMF application to OpenVault and the ov_shelf(8) command to configure OpenVault to use the shelves. See Chapter 3, "Configure OpenVault" on page 35.

Summary of the Completed Configuration

As an example, the procedures in this guide will lead you through the process of configuring a single cabinet with multiple shelves. You must adjust the procedures and the DMF configuration file to fit your site-specific situation.

After you complete the procedures in this guide, the cabinet will have up to eight independent OpenVault libraries, C00 through C07, and a set of up to sixteen drives per shelf, such as C00d00 through C00d15 for shelf 0. Each library will be used with a corresponding volume group, such as vg_c00 through vg_c07. (All volumes in a given volume group must reside in the RAID sets on the same COPAN MAID shelf.) The volume groups will be managed as two DMF migrate groups (mg0 and mg1 for COPAN MAID as permanent storage or copan_fmc1 and copan_fmc2 for fast-mount cache).

Figure 1-9 and Figure 1-10 describe the parts of the configuration that relate to COPAN MAID.

Note: Figure 1-9 corresponds to the sample configuration file `dmf.conf.copan_maid`. In that sample file, conforming to DMF best practices, one copy of migrated data is stored in each of the two migrate groups.

COPAN		OpenVault		DMF		
Cabinet	Shelf name	Library name	Drive names	Drive group	Volume group	Migrate group
shelf 7	C07	C07	C07d00‑C07d15	dg_c07	vg_c07	
shelf 6	C06	C06	C06d00‑C06d15	dg_c06	vg_c06	mg1
shelf 5	C05	C05	C05d00‑C05d15	dg_c05	vg_c05	
shelf 4	C04	C04	C04d00‑C04d15	dg_c04	vg_c04	
shelf 3	C03	C03	C03d00‑C03d15	dg_c03	vg_c03	
shelf 2	C02	C02	C02d00‑C02d15	dg_c02	vg_c02	mg0
shelf 1	C01	C01	C01d00‑C01d15	dg_c01	vg_c01	
shelf 0	C00	C00	C00d00‑C00d15	dg_c00	vg_c00	

Figure 1-9 Sample Configuration for COPAN MAID as Permanent Storage

COPAN		OpenVault		DMF			
Cabinet	Shelf name	Library name	Drive names	Drive group	Volume group	Migrate group	Fast-mount cache
shelf 3	C03	C03	C03d00‑C03d15	dg_c03	vg_c03		copan_fmc2
shelf 2	C02	C02	C02d00‑C02d15	dg_c02	vg_c02		
shelf 1	C01	C01	C01d00‑C01d15	dg_c01	vg_c01	mg_fmc	copan_fmc1
shelf 0	C00	C00	C00d00‑C00d15	dg_c00	vg_c00		

Figure 1-10 Sample Configuration for COPAN MAID as a Fast-Mount Cache

Preliminary Requirements

Before you configure the COPAN MAID system and DMF, do the following:

- "Set Up the DMF Server and COPAN MAID System" on page 16
- "Contact SGI to Apply the AOR Files" on page 16
- "Create the Directories that DMF Requires" on page 16
- "Determine the Backup Requirements for Your Site" on page 18
- "Install the DMF Licenses" on page 19
- "Establish OpenVault Communication" on page 20

Set Up the DMF Server and COPAN MAID System

You must ensure that the DMF server and the COPAN MAID system are on the network and zoned appropriately. The RAID sets must be visible only to the active DMF server, the passive DMF server (if applicable), and the parallel data mover nodes; the RAID sets must not be visible to any other nodes. You must also install the required software for COPAN MAID and DMF. For more information, see the manuals and release notes listed in "About This Guide".

Contact SGI to Apply the AOR Files

The *always on region* (AOR) RAID set allows concurrent access to small, critical sections of all the other LUNs, whether their underlying RAID sets are spinning or not. The special mapping is applied to the MAID shelf from a predefined AOR configuration file that is specific to the host operating system, volume manager, filesystem, application, and capacity of the disks in the RAID set.

Before you can configure DMF for use with COPAN MAID, the AOR configuration files must be applied by SGI service personnel. For assistance, contact SGI Support.

Create the Directories that DMF Requires

Use the information about configuring DMF administrative directories appropriately in the "Best Practices" chapter of *DMF 6 Administrator Guide for SGI InfiniteStorage* to create the required filesystems and directories of the appropriate size on a general-purpose RAID storage system.

In a production system, SGI in most cases recommends that you restrict these directories to DMF use and make them the mountpoint of a filesystem, in order to limit the loss of data in the case of a filesystem failure.

Note: A DMF administrative directory must not be in a DMF-managed filesystem.

You specify the location of these directories by using parameters in the DMF configuration file. The following lists show the directory names used by the sample configuration file (using these names will minimize the number of changes you must make):

- **Required** to be dedicated to DMF use and to be a filesystem mountpoint:

  - *(If used)* /dmf/move for the scratch filesystem that is used to move files between volume groups

    Note: You must specify a scratch filesystem if you intend to use the dmmove(8) command. The scratch filesystem should be dedicated to the dmmove function. You must use the dmi,mtpt=/dmf/move option.

- **Recommended** to be dedicated to DMF use and to be a filesystem mountpoint:

  - /dmf/home for directories in which the DMF daemon database, library server (LS) database, and related files reside

  - /dmf/journals for directories in which the journal files for the daemon database and LS database will be written

  - /dmf/spool for directories in which DMF log files are kept

  - /dmf/tmp for directories in which DMF puts temporary files for its own internal use

  - *(If used)* /dmf/backups for DMF backup files created by DMF backup tasks

You can use the df(1) command to verify that the filesystems required for the DMF administrative directories and the DMF-managed filesystems are mounted.

Determine the Backup Requirements for Your Site

This section discusses the following:

- "Site-Specific Factors to Consider for Backups to Disk" on page 18

- "Backup Filesystem and RAID Set Requirements" on page 18

- "Default Backup Policy for DMF and COPAN MAID" on page 19

Site-Specific Factors to Consider for Backups to Disk

You can configure backups to tape or to disk, such as on a reserved portion of the RAID set that will not be managed by DMF. When backing up to disk, the approximate formula for the amount of disk space that you must reserve for backups is as follows:

$$Backup\_Space\_Needed\_Per\_Day * (Retention\_Period\_In\_Days + 1) = Reserved\_Disk\_Space$$

where *Backup_Space_Needed_Per_Day* is a factor of:

- The amount of data that is not migrated at the time a backup takes place

- The size of the DMF databases, which is a function of the number of migrated files and the number of DMF copies of each file

- The number of full and/or partial backups to retain, as determined by the frequency of backups and the backup retention period

For more information, see the information about configuring DMF directories appropriately in the "Best Practices" chapter of *DMF 6 Administrator Guide for SGI InfiniteStorage*, particularly the information about HOME_DIR size.

Backup Filesystem and RAID Set Requirements

There are the following requirements:

- The backup filesystem must not be a DMF-managed filesystem.

- The backup filesystem must be visible to the DMF server. In an HA environment, it must be a filesystem that is visible from or can be moved to the active DMF server in the event of a failover.

- If the backups are placed on a RAID set, the RAID set must be owned by the DMF server. For details on configuring a RAID set for backups, see "Format a Shelf for Both DMF and Backups *(Optional)*" on page 29.

Default Backup Policy for DMF and COPAN MAID

The sample DMF configuration file for COPAN MAID does the following:

- Writes the backups to disk. The disk can be on a non-DMF region of a RAID set or it can be some existing (non-DMF-managed) filesystem in your environment.

  Note: You can choose to use physical tapes instead if you edit the DMF configuration file.

- Performs a full backup once a week (Sundays at 00:00) and a partial backup the remaining days (Monday—Saturday at 00:00), creating backups of all DMF-managed filesystems.

- Causes all of the data in the DMF-managed filesystems to be migrated before the backups take place (except for files that do not meet the site migration policy).

- Removes the bit-file identifiers from the DMF databases for permanently deleted files.

- Retains the backups for four weeks (28 days). The disk space used by backups is recycled after the retention period is completed.

If you want to change these backup policies, you must modify the DMF configuration file and the procedures in this guide accordingly.

Install the DMF Licenses

To install the DMF licenses, copy the DMF license keys into the `/etc/lk/keys.dat` file on the DMF server. For more information, see the chapter about licensing in *DMF 6 Administrator Guide for SGI InfiniteStorage*.

Establish OpenVault Communication

If this is a new DMF installation, you must use the ov_admin(8) command to do the following, as documented in the chapter about mounting service configuration tasks in *DMF 6 Administrator Guide for SGI InfiniteStorage*:

1. Initially configure the OpenVault server.

2. Configure OpenVault for each parallel data mover node.

Note: The other sections in that chapter do not apply to a COPAN MAID environment. You will use the instructions later in this guide instead; see "Activate the dmf Application Instances in OpenVault" on page 36.

Overview of the Steps in this Guide

This guide leads you through the following steps, which you will perform as the root user:

1. Select appropriate COPAN cabinet identifiers to be used in the naming convention.

2. Format the RAID sets for each shelf, optionally including non-DMF regions for backups.

3. Create the volumes.

4. Configure OpenVault.

5. Configure DMF and import the DMF volumes into OpenVault.

6. *(Optional)* Configure for HA. See *High Availability Guide for SGI InfiniteStorage*.

⚠ **Caution:** After your system is configured and running, you should not stop the disks in a COPAN MAID shelf if those disks are currently being used by an OpenVault LCP. First view the output from dmstat(8) or ovstat(8) to ensure that DMF is not using the disks in that shelf, then stop the OpenVault LCP associated with that shelf, and finally stop the disks. For more information, see the "Best Practices" chapter of the *DMF 6 Administrator Guide for SGI InfiniteStorage*.

Format the RAID sets and Create Volumes

This chapter discusses the following:

- "Selecting Appropriate Cabinet Identifiers" on page 21
- "Matching Device Names to MAID Shelves" on page 23
- "Formatting the RAID Sets for a Shelf and Creating Volumes" on page 24

Selecting Appropriate Cabinet Identifiers

This section discusses the following:

- "Cabinet ID Choice" on page 21
- "Cabinet ID Check *(Existing Site Only)*" on page 22

Cabinet ID Choice

The volume serial numbers (VSNs) for all volumes used by DMF must be unique, whether they are in a library managed by TMF or OpenVault. The following OpenVault component names must also be unique across the site:

- Library names
- Drive names
- Drive groups

When you format the RAID sets on the MAID shelves, the characters you choose for the *CC* component of the VSN naming convention (see "Naming Conventions" on page 13) for each cabinet will also dictate the names that will be chosen for the shelves and the OpenVault components (described in "Summary of the Completed Configuration" on page 14).

You must consider your site's upgrade plans for additional COPAN MAID cabinets and tape libraries when choosing the *CC* component.

Cabinet ID Check *(Existing Site Only)*

If you have an existing DMF environment, you must also ensure that there will be no conflicts with existing OpenVault components by using the `precheck` feature of the `ov_shelf`(8) command to examine the existing OpenVault VSNs and OpenVault components at your site and report any potential conflicts with the cabinet ID you have chosen:

ov_shelf precheck *cabinetID*

This command will report if any LCP, DCP, or cartridge name collisions would occur with future shelves named with the specified cabinet ID.

If there are no conflicts, no output will appear. For example, to check all of the shelves associated with a cabinet ID of `C0` (that is, the eight shelf IDs `C00` through `C07`):

```
# ov_shelf precheck C0
#
```

However, if there are any potential conflicts, `ov_shelf` will report that there are unexpected results. For example, in the following check of the cabinet ID `C1`, the output shows that there are potential conflicts with an already existing OpenVault library named `C13` and already existing drives such as `C13d00`:

```
# ov_shelf precheck C1
There is an unexpected "C13" LIBRARY record in the openvault database
There is an unexpected "C13@syrah" LCP record in the openvault database
There is an unexpected "C13d00" DRIVE record in the openvault database
There is an unexpected "C13d00@syrah" DCP record in the openvault database
There is an unexpected "C13d01" DRIVE record in the openvault database
There is an unexpected "C13d01@syrah" DCP record in the openvault database
There is an unexpected "C13d02" DRIVE record in the openvault database
There is an unexpected "C13d02@syrah" DCP record in the openvault database
There is an unexpected "C13d03" DRIVE record in the openvault database
There is an unexpected "C13d03@syrah" DCP record in the openvault database
There is an unexpected "C13d04" DRIVE record in the openvault database
There is an unexpected "C13d04@syrah" DCP record in the openvault database
There is an unexpected "C13d05" DRIVE record in the openvault database
There is an unexpected "C13d05@syrah" DCP record in the openvault database
There is an unexpected "C13d06" DRIVE record in the openvault database
There is an unexpected "C13d06@syrah" DCP record in the openvault database
#
```

In this case, you must choose a different cabinet ID and rerun the test.

For example, you could test the cabinet ID M1. Rerun the ov_shelf command until no conflicts appear. For example:

```
# ov_shelf precheck M1
#
```

Because the preliminary check of M1 did not show any potential conflicts, you could use it as the cabinet ID naming convention in Chapter 2, "Format the RAID sets and Create Volumes" on page 21.

Note: If you are also using TMF, you must manually verify that the cabinet ID will not conflict with any VSNs.

Matching Device Names to MAID Shelves

To determine the available devices on the COPAN MAID shelves that are visible on the SCSI bus, do the following:

1. Log in to the Platform Service Manager (PSM) GUI, where *ipaddress* is the IP address of the PSM server:

   http://*ipaddress*:8180/psmweb/

   The MAID shelves will be displayed in order from shelf 7 (top) through shelf 0 (bottom), as applicable.

2. Drill down to the shelf details and note the WWPN of both ports on each shelf.

   For example, suppose the port numbers for shelf 0 are as follows:

   ```
   Port 0 WWPN:  50:00:ED:5E:35:C3:43:03
   Port 1 WWPN:  50:00:ED:5E:35:C3:43:04
   ```

   You will look for numbers similar to the above (but using lowercase and without colon delimiters) in step 4 below. Only one of the port WWPNs will be found, even if both ports are connected to the system, because ov_copan chooses one as the primary port.

   For more information about PSM, see the *Platform Service Manager Online Help* available from the PSM **Help** menu.

3. On each node in the DMF configuration, use the `list` action to `ov_copan` and note the `sg` devices:

ov_copan list

For example, the following output from `node1` shows that there are three shelves (`/dev/sg4`, `/dev/sg71`, and `/dev/sg99`):

```
node1# ov_copan list
LUN Device      Vendor   Product Serial Number               Name       DMF OTH FOR UNF VSNs PB
  0 /dev/sg4  COPANSYS 8814    COPANSYS0000000080900283 ---        0   0   0  26    0 16
  0 /dev/sg71 COPANSYS 8814    COPANSYS0000000063800042 ---        0   0   0  26    0 16
  0 /dev/sg99 COPANSYS 8814    COPANSYS0000000051200018 ---        0   0   0  26    0 16
```

This output also shows that all 26 RAID sets on each shelf are unformatted, and that the power budget (`PB`) allows 16 RAID sets to have disks spinning at any given time. (*LUN 0* refers to the location of the shelf controller.)

4. For each `/dev/sg`*N* device on each node, obtain its WWPN and match that up with a corresponding shelf WWPN obtained from the PSM by using the `sg_inq`(8) command and searching for the relevant information in the output:

sg_inq --page=0x83 *device* **| grep -A2 "NAA 5"**

For example, for `/dev/sg4` on `node1`, suppose the output is as follows:

```
node1# sg_inq --page=0x83 /dev/sg4 | grep -A2 "NAA 5"
            NAA 5, IEEE Company_id: 0xed5
            Vendor Specific Identifier: 0xe35c34303
            [0x5000ed5e35c34303]                    <<<<<<< WWPN location
```

In this case, the WWPN for `/dev/sg4` is `5000ed5e35c34303`.

5. Match the device WWPN obtained in step 4 with the shelf WWPN obtained in step 2. For example, the `/dev/sg4` WWPN `5000ed5e35c34303` corresponds to the shelf 0 port 0 WWPN of `50:00:ED:5E:35:C3:43:03`. Therefore, in this example, device `/dev/sg4` equates to shelf 0.

Formatting the RAID Sets for a Shelf and Creating Volumes

This section discusses the following:

- "Summary of the `ov_copan` Command Syntax" on page 25

- "Summary of `ov_copan format` Command Syntax" on page 25

- "Format an Entire Shelf for DMF Use" on page 27

- "Format a Shelf for Both DMF and Backups *(Optional)*" on page 29

For more details, see the `ov_copan`(8) man page.

⚠️ **Caution:** Formatting a RAID set will destroy any existing data on that RAID set. The `ov_copan` command will prompt you for confirmation if the operation is about to destroy existing data. When you initially configure DMF and COPAN MAID, this is not an issue because there is no data on the RAID sets.

If there are unrecognized XVM slices on the RAID set, the format operation will fail rather than destroy the data. You must use the `xvm` command to remove any such XVM slices before you can format the RAID set with `ov_copan`.

Summary of the `ov_copan` Command Syntax

Note: You must run the `ov_copan` (8) commands `format` and `create` from the node that you want to own the shelf (the *owner node*). DMF will use the owner node to perform migrations and recalls on that shelf. (In a high-availability environment, ownership is transferred as part of HA failover.)

This section discusses the following:

- "Summary of `ov_copan format` Command Syntax" on page 25

- "Summary of `ov_copan create` Command Syntax" on page 27

For more information, see "Naming Conventions" on page 13 and the `ov_copan`(8) man page.

Summary of `ov_copan format` Command Syntax

When you format a RAID set for DMF use, the formatting process initializes an XFS filesystem on the RAID set.

> **Note:** This filesystem is reserved exclusively for use with DMF, do not modify it directly.

To format the RAID sets:

```
ov_copan format device_or_shelfID [-N shelfID] [-m raidlist] [-t type [-r reserved_nonDMFsize]]
```

where:

- *device_or_shelfID* is the device name associated with a shelf (such as /dev/sg4), or, after a shelf has been named in ov_copan, the shelf identifier (such as C00). See "Matching Device Names to MAID Shelves" on page 23.

  > **Note:** You should use the recommended naming convention unless the ov_shelf precheck command found potential conflicts, as documented in "Selecting Appropriate Cabinet Identifiers" on page 21.

- *shelfID* is the shelf identifier (such as C00). You will only use the -N option the first time that you format a RAID set for a given shelf.

- *raidlist* is the list of RAID sets on the shelf to be acted upon. By default, all RAID sets are chosen. *raidlist* is a comma-separated list of ranges, such as:

  ```
  A,D-F,L-P
  ```

- *type* is the type of formatting, either:

  - dmf for DMF-managed filesystems and filesystems for DMF administrative directories (default)

  - other for non-DMF regions, such as for backups from the daily DMF backup

    > **Note:** To use a RAID set for a backups, the filesystem must not be managed by DMF.

  If you specify dmf, you can choose to reserve a portion of the disk for non-DMF use by specifying -r. If you do not specify -r, the entire disk is formatted for DMF use.

  If you specify other, the entire disk is formatted for non-DMF use.

- *reserved_nonDMFsize* is the amount of the disk to be reserved for non-DMF use. (The -r option only applies when the *type* is dmf.) By default, no space is reserved for non-DMF use. If you specify -r, you must specify a size. You can specify the following units (the default unit is b):

  b = bytes
  s = sectors (512 bytes)
  k = kilobytes (2^10 bytes)
  m = megabytes (2^20 bytes)
  g = gigabytes (2^30 bytes)
  t = terabytes (2^40 bytes)
  p = petabytes (2^50 bytes)
  e = exabytes (2^60 bytes)
  % = percentage

Summary of `ov_copan create` Command Syntax

DMF volumes must be at least 512 MB. There can be at most 1,296 DMF volumes in a RAID set. To create DMF volumes on the formatted RAID sets of the specified shelf:

`ov_copan create` *shelfID* [-m *raidlist*] [-c *count_of_volumes*] [-s *size_of_volumes*]

where:

- *shelfID* is the shelf identifier (such as C00).

- *raidlist* is the list of RAID sets on the shelf to be acted upon. By default, all RAID sets are chosen. *raidlist* is a comma-separated list of ranges, such as:

  A,D-F,L-P

- *count_of_volumes* is the number of volumes to create on each of the specified RAID sets. Normally, you specify either -c or -s but not both.

- *size_of_volumes* is the size of volumes to be created (using the same unit specifications as listed above for *reserved_nonDMFsize*). The value can contain a trailing modifier of "+" or "-", which means to adjust the given size up or down, respectively, in order to use the remaining available space on the RAID set.

Format an Entire Shelf for DMF Use

To format an entire shelf for use as DMF secondary storage, do the following:

1. Format all 26 RAID sets for the shelf and name the shelf.

   For example, to format all 26 RAID sets for device /dev/sg4 and name the shelf
   C00 (because device /dev/sg4 equates to MAID shelf 0, as determined in
   "Matching Device Names to MAID Shelves" on page 23):

   ```
   ownernode# ov_copan format /dev/sg4 -N C00
   format /dev/sdj     as C00A fmt DMF
   format /dev/sdk     as C00B fmt DMF
   format /dev/sdl     as C00C fmt DMF
   format /dev/sdm     as C00D fmt DMF
   ...
   ```

2. Verify that the RAID sets have been formatted.

   For example, the following output shows that device /dev/sg4 (shelf 0) has the
   shelf name C00 and 26 RAID sets that are formatted for DMF:

```
ownernode# ov_copan list
 LUN Device      Vendor   Product Serial Number            Name       DMF OTH FOR UNF VSNs PB
   0 /dev/sg4  COPANSYS 8814     COPANSYS0000000080900283 C00         26   0   0   0    0 16
   0 /dev/sg71 COPANSYS 8814     COPANSYS0000000063800042 ---          0   0   0  26    0 16
   0 /dev/sg99 COPANSYS 8814     COPANSYS0000000051200018 ---          0   0   0  26    0 16
```

3. Create DMF volumes on the shelf. The recommended size is 128 GB, specified
   with a trailing "+" or "-" to allow ov_copan to adjust the size to fill all available
   space in the DMF region. For example, the following will create volumes that are
   at least 128 GB:

   ```
   ownernode# ov_copan create C00 -s 128g+
   ```

4. Verify that the volumes have been created. For example, the following shows that
   there are 1,040 volumes on shelf C00:

```
ownernode# ov_copan list
 LUN Device      Vendor   Product Serial Number            Name       DMF OTH FOR UNF VSNs PB
   0 /dev/sg4  COPANSYS 8814     COPANSYS0000000080900283 C00         26   0   0   0 1040 16
   0 /dev/sg71 COPANSYS 8814     COPANSYS0000000063800042 ---          0   0   0  26    0 16
   0 /dev/sg99 COPANSYS 8814     COPANSYS0000000051200018 ---          0   0   0  26    0 16
```

Format a Shelf for Both DMF and Backups *(Optional)*

If you want to put DMF backups on a MAID shelf, you must format a portion of a RAID set or an entire RAID set as non-DMF. Which method you choose depends upon the amount of space needed, as determined in "Determine the Backup Requirements for Your Site" on page 18. This section discusses the following:

- "Formatting Considerations for Backup" on page 29
- "Dedicate Entire RAID sets for Backups" on page 30
- "Divide a RAID set into DMF and Backup Regions" on page 32

To use physical tapes for DMF backups, see the *DMF 6 Administrator Guide for SGI InfiniteStorage*.

Formatting Considerations for Backup

If you will store DMF backups on a RAID set, note the following:

- The shelf containing the RAID set must be owned by the DMF server.

- You may format an entire RAID set or reserve just a portion of a RAID set for non-DMF use. When formatting a given RAID set for both DMF and non-DMF use, the DMF region must be at least 200 GB. You must use the mkfs.xfs(8) and xvm(8) commands to make use of the non-DMF regions.

- The backup task can make multiple copies of the backups (specified by the DUMP_MIRRORS parameter in the DMF configuration file). You can choose to increase redundancy by formatting multiple RAID sets to store backups.

- SGI recommends that you leave the backup filesystem unmounted when not in use; the backup task will mount it automatically if it is in /etc/fstab.

Note: If the backup filesystem is always mounted, OpenVault will assume that the RAID set is in use and will throttle DMF's use of the shelf back to avoid going over the power budget. The same is true for any non-DMF RAID set. SGI recommends that you dedicate an entire RAID set for use with DMF.

Dedicate Entire RAID sets for Backups

To dedicate one or more entire RAID sets on a shelf for backups, do the following:

1. Format most of the RAID sets on the shelf for DMF use by using the -m *raidlist* option.

   For example, to format the first 25 RAID sets (named from A through Y) of device /dev/sg71 for DMF use and name the shelf C01 (for cabinet 0, shelf 1):

   ```
   dmfserver# ov_copan format /dev/sg71 -N C01 -m A-Y
   format /dev/sdj      as C01A fmt DMF
   format /dev/sdk      as C01B fmt DMF
   format /dev/sdl      as C01C fmt DMF
   format /dev/sdm      as C01D fmt DMF...
   ```

2. Format the remaining RAID sets entirely as non-DMF for use with backups. For example, to format the one remaining RAID set (Z) as a non-DMF region suitable for backups:

   ```
   dmfserver# ov_copan format C01 -m Z -t other
   format /dev/sdab     as C01Z fmt Other
   ```

3. Verify by viewing the output from the list action. For example, the following output shows that for shelf C01, there are 25 RAID sets that are formatted for DMF use and 1 RAID set formatted for other use:

```
dmfserver# ov_copan list C01
 LUN Device     Vendor   Product  Serial Number              Name        DMF OTH FOR UNF VSNs PB
   0 /dev/sg71 COPANSYS 8814     COPANSYS0000000063800042 C01          25   1   0   0    0 16
```

4. Create DMF volumes on most of the RAID sets by using the -c or -s option. The recommended size is 128 GB, specified with a trailing "+" or "-" to allow ov_copan to adjust the size to fill all available space in the DMF region. For example, to create as many 128-GB volumes as will fit on RAID sets A-Y of shelf C01:

   ```
   dmfserver# ov_copan create C01 -s 128g+
   create 40 VSNs of size 128.92g on C01A
   create 40 VSNs of size 128.92g on C01B
   create 40 VSNs of size 128.92g on C01C
   ```

```
create 40 VSNs of size 128.92g on C01D
...
skip C01Z: not formatted for DMF...
```

5. Verify that the volumes have been created. For example, the following shows that there are 1,000 volumes on shelf C01:

```
dmfserver# ov_copan list C01
 LUN Device     Vendor   Product Serial Number            Name    DMF OTH FOR UNF VSNs PB
   0 /dev/sg71 COPANSYS 8814    COPANSYS0000000063800042 C01      25   1   0   0 1000 16
```

6. Create a local-domain XVM slice that covers the whole usable space of the non-DMF RAID set and name the XVM volume that will be used for backups. The physical volume (or *physvol*) will be named with the RAID set identifier:

```
dmfserver# xvm -domain local slice -volname backup_volume_name -all phys/copan_RAID_set_ID
```

For example, to create a slice for the last RAID set in cabinet 0, shelf 1 (which has a *RAID_set_ID* of C01Z) and name the backup volume dmf_backups:

```
dmfserver# xvm -domain local slice -volname dmf_backups -all phys/copan_C01Z
</dev/lxvm/dmf_backups>  slice/copan_C01Zs0
```

For more information, see the xvm(8) man page and the *XVM Volume Manager Administrator Guide*.

7. Make the filesystem for the backup volume.

   For example, to make the filesystem for dmf_backups in the /dev/lxvm
   directory:

```
dmfserver# mkfs.xfs /dev/lxvm/dmf_backups
meta-data=/dev/lxvm/dmf_backups  isize=256   agcount=32, agsize=11443303 blks
         =                       sectsz=512  attr=2
data     =                       bsize=4096  blocks=366185696, imaxpct=5
         =                       sunit=0     swidth=0 blks
naming   =version 2              bsize=4096  ascii-ci=0
log      =internal log           bsize=4096  blocks=178801, version=2
         =                       sectsz=512  sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096  blocks=0, rtextents=0
```

   For more information, see the mkfs.xfs(8) man page.

8. Add the filesystem to /etc/fstab with the noauto mount option. For example:

```
/dev/lxvm/dmf_backups /dmf/backups xfs defaults,noauto
```

Divide a RAID set into DMF and Backup Regions

To reserve a portion of one RAID set for backups and allow DMF use on the
remainder of that RAID set and all other RAID sets on the shelf, do the following:

1. Format most of the RAID sets on the shelf for DMF use by using the -m *raidlist*
   option.

   For example, to format the first 25 RAID sets (named from A through Y) of device
   /dev/sg99 for DMF use and name the shelf C02 (for cabinet 0, shelf 2):

```
dmfserver# ov_copan format /dev/sg99 -N C02 -m A-Y
format /dev/sdj     as C02A fmt DMF
format /dev/sdk     as C02B fmt DMF
format /dev/sdl     as C02C fmt DMF
format /dev/sdm     as C02D fmt DMF...
```

2. Reserve a portion of the remaining RAID sets for backups by using the -m and -r
   options.

For example, to reserve 500 GB for backups of the last RAID set (Z) of shelf C02:

```
dmfserver# ov_copan format C02 -m Z -r 500g
format /dev/sdac    as C02Z fmt DMF
```

3. Verify by viewing the output from the list action. For example, the following shows that all 26 RAID sets are formatted at least in part for DMF use:

```
dmfserver# ov_copan list C02
 LUN Device     Vendor   Product  Serial Number              Name        DMF OTH FOR UNF VSNs PB
   0 /dev/sg99 COPANSYS 8814      COPANSYS0000000051200018 C02           26   0   0   0    0 16
```

4. Create DMF volumes on the DMF-regions of the RAID sets by using the -c or -s option. The recommended size is 128 GB, specified with a trailing "+" or "-" to allow ov_copan to adjust the size to fill all available space in the DMF region. For example, to create as many 128–GB volumes as will fit on RAID sets A-Y and the DMF-region of RAID set Z of shelf 2:

```
dmfserver# ov_copan create C02 -s 128g+
create 40 VSNs of size 128.92g on C02A
create 40 VSNs of size 128.92g on C02B
create 40 VSNs of size 128.92g on C02C
create 40 VSNs of size 128.92g on C02D
...
create 20 VSNs of size 129.48g on C02Z
```

Note: The size and number of volumes are different for C02Z because the DMF region on that RAID set is smaller.

5. Verify by viewing the output from the list action. For example:

```
dmfserver# ov_copan list C02
 LUN Device     Vendor   Product  Serial Number              Name        DMF OTH FOR UNF VSNs PB
   0 /dev/sg99 COPANSYS 8814      COPANSYS0000000051200018 C02           26   0   0   0 1020 16
```

6. Find the sector where the slice will start by examining the output of the show action to xvm:

```
dmfserver# xvm -domain local show -v phys/copan_RAID_set_ID | grep -A5 "Physvol Usage"
```

For example, for the C02Z RAID set:

```
dmfserver# xvm -domain local show -v phys/copan_C02Z | grep -A5 "Physvol Usage"
Physvol Usage:
Start        Length       Name
---------------------------------------------------
0            1880909568   slice/copan_C02Zs0
1880909568   1048576000   (unused)
```

For more information, see the xvm(8) man page and the *XVM Volume Manager Administrator Guide*.

7. Create a local-domain XVM slice that begins at the slice location displayed in the previous step and name the XVM volume that will be used for backups. The physical volume (or *physvol*) will be named with the RAID set identifier:

```
dmfserver# xvm -domain local slice -volname backup_volume_name -start sector phys/copan_RAID_set_ID
```

For example, to create a slice for the last RAID set in shelf 2 (which has a *RAID_set_ID* of C02Z) beginning at sector 1880909568 and name the backup volume dmf_backups:

```
dmfserver# xvm -domain local slice -volname dmf_backups -start 1880909568 phys/copan_C02Z
</dev/lxvm/dmf_backups>  slice/copan_C02Zs1
```

8. Make the filesystem for the backup volume. For example:

```
dmfserver# mkfs.xfs /dev/lxvm/dmf_backups
meta-data=/dev/lxvm/dmf_backups  isize=256    agcount=16, agsize=8192000 blks
         =                       sectsz=512   attr=2
data     =                       bsize=4096   blocks=131072000, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0
log      =internal log           bsize=4096   blocks=64000, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
```

For more information, see the mkfs.xfs(8) man page.

9. Add the filesystem to /etc/fstab with the noauto mount option. For example:

```
/dev/lxvm/dmf_backups /dmf/backups xfs defaults,noauto
```

Configure OpenVault

To configure OpenVault for use with COPAN MAID, do the following:

- "Use ov_shelf to Configure OpenVault for Each Shelf" on page 35

- "Activate the dmf Application Instances in OpenVault" on page 36

Note: The example output for some commands shown here has been edited for length and readability.

Use ov_shelf to Configure OpenVault for Each Shelf

You will use the ov_shelf(8) command to add the required library control program (LCP), drive control programs (DCPs), and OpenVault drive group for each MAID shelf. From the node that owns the shelf, enter the following command once for each MAID shelf:

```
ownernode# ov_shelf create shelfID
```

For example, if the parallel data mover node node1 owns shelves 0 and 1, and node2 owns shelves 2 and 3:

```
node1# ov_shelf create C00
node1# ov_shelf create C01

node2# ov_shelf create C02
node2# ov_shelf create C03
```

Note: Before ov_shelf takes any action, it verifies that there will be no conflicts with existing volumes at your site. If it detects a potential conflict, it will exit with an error. Because you already verified that there would be no problems by using the precheck operation in "Selecting Appropriate Cabinet Identifiers" on page 21, you should not see any errors.

Activate the dmf Application Instances in OpenVault

Following is an example of the steps you will take to add an unprivileged and a privileged instance of the dmf application to OpenVault, using an example host named dmfserver as the OpenVault server. The characters **###** in the right margin highlight comments related to the steps, which follow the example.

```
dmfserver# ov_admin                                                           ### STEP 1


Name where the OpenVault server is (or will be) listening? [dmfserver]        ### STEP 2


    OpenVault Configuration Menu for server "dmfserver"

      Configuration on Machines Running LCPs and DCPs
         1 - Manage LCPs for locally attached Libraries
         2 - Manage DCPs for locally attached Drives


      Configuration on Admin-Enabled Machines
        11 - Manage Cartridge Groups
        12 - Manage Drive Groups
        13 - Import Media


      Configuration on the OpenVault Server Machine
        21 - Manage Applications
        22 - Manage OpenVault Client Machines


         q - Exit.


    Which operation would you like to do: 21                                   ### STEP 3



    Manage Applications Menu

         1 - Create a new Application
         2 - Delete an Application
         3 - Show all existing Applications

         4 - Activate another Application Instance for an existing Application
         5 - Deactivate an Application Instance
         6 - Show all activated Application Instances

         r - Return to Main Menu.
```

```
      q - Exit.


   Which operation would you like to do: 4                              ### STEP 4


   Select the Application for which you want to activate a new Instance

      1 - dmf
      2 - ov_umsh

      r - Return to Previous Menu.
      q - Exit.


   Which operation would you like to do: 1                              ### STEP 5

Enter the name of the Host where an instance of Application "dmf" will run [dmfserver] *  ### STEP 6


Enter the Application's instance name or "*" [] *                       ### STEP 7

Should this Instance of the Application "dmf" be "privileged"? [No]     ### STEP 8


What security key will the Application use [none]                       ### STEP 9

Unprivileged Instance "*" of Application "dmf"
was successfully activated on "dmfserver".

Press enter to continue...

   Manage Applications Menu

      1 - Create a new Application
      2 - Delete an Application
      3 - Show all existing Applications

      4 - Activate another Application Instance for an existing Application
      5 - Deactivate an Application Instance
      6 - Show all activated Application Instances
```

```
        r - Return to Main Menu.
        q - Exit.


    Which operation would you like to do: 4                          ### STEP 10


    Select the Application for which you want to activate a new Instance

        1 - dmf
        2 - ov_umsh

        r - Return to Previous Menu.
        q - Exit.

    Which operation would you like to do: 1                          ### STEP 11

Enter the name of the Host where an instance of Application "dmf" will run [dmfserver] * ### STEP 12

Enter the Application's instance name or "*" [] *                     ### STEP 13

Should this Instance of the Application "dmf" be "privileged"? [No] yes    ### STEP 14

What security key will the Application use [none]                     ### STEP 15

Privileged Instance "*" of Application "dmf"
was successfully activated on "dmfserver".

Press enter to continue...

    Manage Applications Menu

        1 - Create a new Application
        2 - Delete an Application
        3 - Show all existing Applications

        4 - Activate another Application Instance for an existing Application

        5 - Deactivate an Application Instance
        6 - Show all activated Application Instances
```

```
      r - Return to Main Menu.
      q - Exit.


 Which operation would you like to do: q                                    ### STEP 16
```

Comments:

1. Log in to the OpenVault server and invoke the OpenVault administration tool `ov_admin`(8).

2. Enter the name associated with the IP address on which the OpenVault server is listening.

3. Enter `21` to manage applications.

4. Enter `4` to activate another application instance for an existing application.

5. Enter `1` to select `dmf`.

6. Enter the wildcard `*` character to allow the `dmf` application to be used from any host. Alternatively, you can repeat these steps to create a privileged and an unprivileged application instance for each system that DMF runs on (each DMF server and each parallel data mover node).

7. Enter the wildcard `*` character for the application instance name.

8. Use the default `no` to create the unprivileged instance.

9. Optionally provide a security key to prevent unauthorized clients from masquerading as the `dmf` application. SGI recommends that you do this if you used the wildcard in step 6. You must use this same key when later running `dmov_keyfile`.

10. Enter `4` again to activate another application instance for an existing application.

11. Enter `1` to select `dmf`.

12. Enter the wildcard `*` character to allow the `dmf` application to be used from any host. (See step 6.)

13. Enter the wildcard `*` character for the application instance name.

14. Enter `yes` to make the application privileged.

15. Optionally provide a security key. (See step 9.)

16. Enter q to exit.

For more details, see *OpenVault Administrator Guide for SGI InfiniteStorage.*

Configure DMF

This chapter describes the minimum steps required to get DMF up and running with a COPAN MAID system. It discusses the following:

- "Configure DMF Parameters" on page 41
- "Provide DMF with the OpenVault Security Keys" on page 52
- "Import the DMF Volumes into OpenVault" on page 53

For more information, see the following:

- *DMF 6 Administrator Guide for SGI InfiniteStorage*
- *DMF 5 Filesystem Audit Guide for SGI InfiniteStorage*
- `dmf.conf`(5) man page

Configure DMF Parameters

This section discusses the following:

- "Summary of Configuration Parameters" on page 41
- "Configure DMF for a New Site" on page 43
- "Configure COPAN MAID for an Existing DMF Site" on page 48

Summary of Configuration Parameters

Table 4-1 and Table 4-1 summarize parameters of particular interest when using COPAN MAID, grouped by the object in the sample file that contains the parameter. For details about setting these parameters appropriately, see the information in *DMF 6 Administrator Guide for SGI InfiniteStorage*.

Note: There is a log and a database with CAT and VOL records for every library server (LS). Depending upon your preferences and the number of shelves and cabinets at your site, you can choose to configure any number of LSs. You could have an LS for each shelf or two, or an LS for each MAID cabinet, or a single LS for all cabinets. However, a given LS can support either COPAN MAID or physical tape, but not both.

Table 4-1 Parameters of Particular Interest for COPAN MAID as Permanent Storage

Sample Object Name	Object Type	Parameter
DMF (shield icon)	base	ADMIN_EMAIL SERVER_NAME
nodename	node	INTERFACE *(Parallel Data Mover option only)*
maid_ls	libraryserver	COPAN_VSNS DRIVE_GROUPS
vg_c00 **through** vg_c07	volumegroup	MAX_PUT_CHILDREN RESERVED_VOLUMES ZONE_SIZE
mg0 **and** mg1	migrategroup	GROUP_MEMBERS

Table 4-2 Parameters of Particular Interest for COPAN MAID as Fast-Mount Cache

Sample Object Name	Object Type	Parameter
DMF (shield icon)	base	ADMIN_EMAIL SERVER_NAME
nodename	node	INTERFACE *(Parallel Data Mover Option only)*
copan_ls	libraryserver	COPAN_VSNS DRIVE_GROUPS

Sample Object Name	Object Type	Parameter

`vg_c00` through `vg_c03`	`volumegroup`	`MAX_PUT_CHILDREN`
		`MERGE_THRESHOLD`
		`RESERVED_VOLUMES`
		`ZONE_SIZE`

`mg_fmc`	`migrategroup`	`GROUP_MEMBERS`

`fmc_task1` and `fmc_task2`	`taskgroup`	`FILE_RETENTION_DAYS`
		`FMC_MOVEFS`
		`FMC_NAME`
		`FREE_VOLUME_MINIMUM`
		`FREE_VOLUME_TARGET`

`copan_fmc1` and `copan_fmc2`	`fastmountcache`	`CACHE_MEMBERS`

Configure DMF for a New Site

To create the DMF configuration using the DMF Manager graphical user interface, do the following:

1. Start the DMF Manager service:

   dmfserver# **service dmfman start**

2. Invoke DMF Manager by pointing your browser to the following location:

   `https://`*your_hostname*`:1179`

3. Accept the security certificate.

   For more information, see the chapter about using DMF Manager in *DMF 6 Administrator Guide for SGI InfiniteStorage.* You can use the **Help** button in DMF Manager to access the **Getting Started** information or the entire *DMF 6 Administrator Guide for SGI InfiniteStorage.*

4. Populate DMF Manager with the appropriate sample DMF configuration file . To do this, right-click in the **Overview** panel and select one of the following:

   Pre-Configured
   > **COPAN MAID LS Sample**

Pre-Configured
> **Fast-Mount Cache Sample**

COPAN MAID LS Sample creates the following:

- One library server (LS): `maid_ls`

- Eight drive groups: `dg_c00` through `dg_c07`

- Eight volume groups (VGs): `vg_c00` through `vg_c07`

- Two migrate groups: `mg0` and `mg1`

- The DMF administrative directories, as described in "Create the Directories that DMF Requires" on page 16

Fast-Mount Cache Sample creates the following:

- One library server: `copan_ls`

- Two fast-mount caches: `copan_fmc1` and `copan_fmc2`

- Four COPAN drive groups: `dg_c00` through `dg_c03`

- Four COPAN volume groups: `vg_c00` through `vg_c03`

- One tape library server: `tape_ls`

- Two tape drive groups: `dg1` and `dg2`

- Four tape volume groups: `vg1a`, `vg1b`, `vg1c`, and `vg2`

- Three migrate groups: `mg0`, `mg1`, and `mg_fmc`

- One resource watcher: `rw`

- The DMF administrative directories, as described in "Create the Directories that DMF Requires" on page 16

Depending upon your site-specific configuration, or if you chose a different naming convention ("Selecting Appropriate Cabinet Identifiers" on page 21), you must modify the DMF configuration accordingly.

5. Make the following changes to the sample configuration, based on the configuration at your site:

Note: This procedure describes the minimum changes you must make. If your configuration differs from the sample configuration, you may need to make more modifications.

a. For each filesystem that DMF will manage, copy and paste the /dmi_fs object, renaming it with the actual filesystem name. Right-click the /dmi_fs object and select:

Configure...
> **Copy**

In the copy, delete the /dmi_fs name and replace it with the name of your filesystem. Figure 4-1 shows an example using the name /myfs.
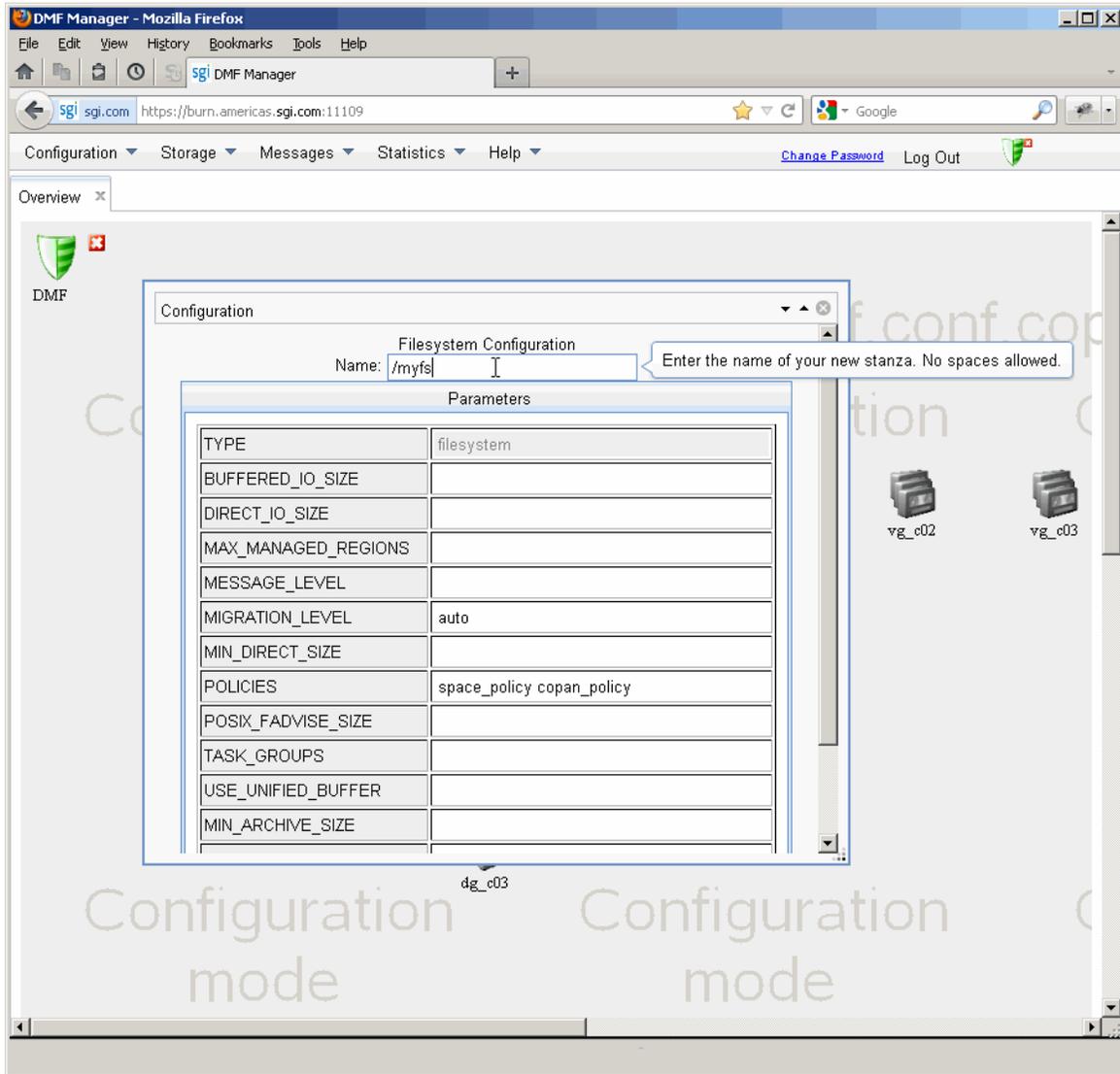
Figure 4-1 Replace the Filesystem Name

Note: For help on a parameter, right-click the parameter name.

After you have created an object for each filesystem, delete the sample `/dmi_fs` object.

b. Change the `ADMIN_EMAIL` parameter to an appropriate address for your site. Right-click the DMF shield icon and select:

Configure ...
> **Modify Base**

Delete the default address and enter in the address for your site. Click **Continue**.

c. Modify the number of drive groups and volume groups according to the number of COPAN shelves in your configuration. Using the procedures in this guide, there should be one drive group and one volume group per shelf, and the migrate groups should contain a balanced number of volume groups.

Note: All volumes in a given VG must reside in the RAID sets on the same COPAN MAID shelf.

For example, if you have six shelves (0-5) and you are using COPAN MAID as permanent storage, you would do the following:

- Delete `dg_c06`, `dg_c07`, `vg_c06`, and `vg_c07`

- Edit `maid_ls` and remove `dg_c06` and `dg_c07` from `DRIVE_GROUPS`

- Edit `mg0` to include `GROUP_MEMBERS` of `vg_c00`, `vg_c01` and `vg_c02`

- Edit `mg1` to include `GROUP_MEMBERS` of `vg_c03`, `vg_c04` and `vg_c05`

d. If you are using the Parallel Data Mover Option:

- Right-click the DMF shield icon and select:

Configure ...
> **Modify Base**

Set `SERVER_NAME` to the system hostname of your DMF server. Click **Continue**.

- For each node in your DMF configuration, right-click anywhere within the **Overview** panel and select:

Add Object...
> **Node...**

Set the name of the object to the node's system hostname. If you are using alternative networks for DMF communication, set INTERFACE appropriately.

6. For fast-mount cache, modify the parameters for the permanent storage target (such as a tape library) as appropriate for your site.

7. Validate your changes. Right-click anywhere within the **Overview** panel and select:

Configure
> **Validate Configuration**

You will see errors about OpenVault and a DMF database, because they are not yet configured:

```
ERROR:   No such file /dmf/home/ov_keys.
ERROR:   Database (/dmf/home/maid_ls/tpvrdm.dat) does not exist.
```

The above errors are expected at this point, and you can disregard them. However, if there are any other errors or warnings, fix them and then rerun the validation. Proceed to step **8** when there are no errors and preferably no warnings, other than the expected OpenVault and database errors.

8. Save the configuration. Right-click anywhere within the **Overview** panel and select:

Configure
> **Apply Configuration**

Configure COPAN MAID for an Existing DMF Site

This section provides an overview of the steps you must take to modify an existing DMF configuration to incorporate COPAN MAID as permanent storage or as a fast-mount cache. For brevity, it describes the changes using stanza excerpts from a modified sample DMF configuration file. You can edit this file by using DMF Manager.

⚠ **Caution:** For instructions about starting and stopping DMF in an HA environment, see *High Availability Guide for SGI InfiniteStorage.*

Do the following:

1. Stop DMF:

   ```
   dmfserver# service dmf stop
   ```

2. Modify the DMF configuration file as follows:

   a. Define a drive group and volume group for each shelf. The names of these objects should be dg_*XXX* and vg_*XXX* respectively, where *XXX* is the lowercase shelf identifier.

      For example, for the shelf identifier c00 for shelf 0 of cabinet C0:

      ```
      define dg_c00
              TYPE                    drivegroup
              VOLUME_GROUPS           vg_c00
              MOUNT_SERVICE           openvault
              OV_INTERCHANGE_MODES    compression
              COMPRESSION_TYPE        snappy
      enddef

      define vg_c00
              TYPE                    volumegroup
              MAX_PUT_CHILDREN        6
              RESERVED_VOLUMES        4
              ZONE_SIZE               4g
      enddef
      ```

      Note: If you are using COPAN MAID as a fast-mount cache, ensure that the names of the libraryserver, drivegroup, and volumegroup objects that define the VGs to be used as a fast-mount cache are the first names listed in LS_NAMES, DRIVE_GROUPS, and VOLUME_GROUPS respectively.

      Define the OV_INTERCHANGE_MODE or COMPRESSION_TYPE parameters only if you want to use compression.

b. Define a new library server, such as `maid_ls`. Adjust `DRIVE_GROUPS` based on the drive groups you added above in step a.

For example:

```
define maid_ls
        TYPE                    libraryserver
        CACHE_SPACE             16g
        COMMAND                 dmatls
        COPAN_VSNS              ON
        DRIVE_GROUPS            dg_c00 dg_c01 ...
        MAX_CACHE_FILE          1048576
        TASK_GROUPS             ls_tasks
enddef
```

c. Add the name of the new `libraryserver` object to the `dmdaemon` object. In general, for optimizing recalls, you should list the name of the COPAN MAID `libraryserver` object after any disk `msp` or disk cache manager (DCM) `msp` objects but before any `libraryserver` objects that contain physical tapes.

For example:

```
define daemon
        TYPE                    dmdaemon
        LS_NAMES                ... maid_ls ...
        ...
enddef
```

d. Enable metrics in the `base` object:

```
define   base
        TYPE                    base
        ...
        EXPORT_METRICS          ON
enddef
```

The `RESERVED_VOLUMES` parameter (used in `vg_XXX`) requires `EXPORT_METRICS` to be `ON`.

e. Update the volume group selection policy to include the new shelves using one of the following methods:

• Using a series of volume groups. For example:

```
define vg_policy
        ...
        SELECT_VG                       vg_c00 vg_c01
enddef
```

- Using one or more migrate groups to easily spread migrated data across multiple shelves, each containing a list of volume groups. For example:

```
define mg0
        TYPE                            migrategroup
        GROUP_MEMBERS                   vg_c00 vg_c01
        ROTATION_STRATEGY               ROUND_ROBIN_BY_BYTES
enddef
define mg1
        TYPE                            migrategroup
        GROUP_MEMBERS                   vg_c02 vg_c03
        ROTATION_STRATEGY               ROUND_ROBIN_BY_BYTES
enddef
```

This method requires that you also define the volume group selection policy. For example:

```
define vg_policy
        ...
        SELECT_VG                       mg0 mg1
enddef
```

f. If you are using COPAN MAID as permanent storage and if you do not already have one, add a taskgroup object for volume merging (named ls_tasks in step 2b above). For example:

```
define ls_tasks
        TYPE                    taskgroup
        RUN_TASK                $ADMINDIR/run_tape_merge.sh at 1:00
        RUN_TASK                $ADMINDIR/run_merge_stop.sh at 3:00
        THRESHOLD               40
enddef
```

Note: If you need a more flexible way of merging volumes, consider using run_merge_mgr.sh instead of run_tape_merge.sh.

g. If you are using COPAN MAID as a fast-mount cache:

 • Add a `fastmountcache` object and specify the members of the fast-mount cache. For example:

```
define  copan_fmc
        TYPE            fastmountcache
        CACHE_MEMBERS   mg_fmc
enddef
```

 • Add a `taskgroup` object to empty the fast-mount cache. For example, the following specifies that when only 10% of the volumes in the fast-mount cache are free, DMF will free the volumes with the oldest write dates until 20% of the volumes are free:

```
define fmc_task
        TYPE                    taskgroup
        RUN_TASK                $ADMINDIR/run_fmc_free.sh at 23:00
        FMC_NAME                copan_fmc
        FREE_VOLUME_MINIMUM     10
        FREE_VOLUME_TARGET      20
enddef
```

3. Validate the configuration. You can do this within DMF Manager or use the following command:

```
dmfserver# dmcheck
```

4. Start DMF:

```
dmfserver# service dmf start
```

Provide DMF with the OpenVault Security Keys

If this is a new site, do the following to allow DMF to communicate with OpenVault:

```
dmfserver# dmov_keyfile
What is the unprivileged key? [none] unpriv_key
What is the privileged key? [none] priv_key
```

If you did not configure keys, press Enter to use the default value of `none`.

Import the DMF Volumes into OpenVault

Do the following to import the DMF volumes into OpenVault:

1. Import the volumes for the library by type into OpenVault and add them to DMF according to the DMF volume group (VG) name (such as vg_c00 for library C00) by entering the following dmov_loadtapes(8) command, once per shelf:

   dmfserver# **dmov_loadtapes -t Copan-MAID -l** *OV_library* **-s** *volumesize_in_bytes VGname*

   Note: The *volumesize_in_bytes* value is used by other DMF tools to estimate the capacity of non-full volumes for reporting purposes only.

   This command will invoke an editor session. If there are any volumes that you do not want added to the library server database, you should delete those VSNs from the list. Otherwise, you should exit the editor.

   For example, you would enter C00 (the library name), vg_c00 (the corresponding VG name), and 128000000000 (for 128 GB):

   ```
   dmfserver# dmov_loadtapes -t Copan-MAID -l C00 -s 128000000000 vg_c00
   ...
   (editor session is invoked)
   Unused tapes in library C00.
   Delete any you do not wish to use for allocation or volume group vg_c00.
   C00AAA
   C00BAB
   C00CAC
   ...
   (exit the editor session)
   ...
   (wait a few moments for the process to complete)
   ...
   Loading tapes into allocation/volume group vg_c00.
   Creating Copan-MAID cartridges in cartgroup carts for vg_c00.
   ```

   The volumes for the library are now imported and ready for use. A volume that has been imported into OpenVault can be referred to as a *tape volume*.

2. Verify that the volumes now have OpenVault cartridge IDs assigned to them by examining the output from the following command:

   dmfserver# **ov_stat -L** *OV_library* **-s**

For example, the following output shows that the volume with VSN C00AAA now has an OpenVault cartridge ID of ooA69kzOtNYADeH5:

```
dmfserver# ov_stat -L C00 -s
Library Name    Broken    Disabled    State      LCP State
C00             false     false       ready      ready
Library: 'C00'

Library  Slot Name  Slot Type  Occupied PCL     Cart ID
C00      slot 1000  LTO        true     C00AAA   ooA69kzOtNYADeH5
C00      slot 1001  LTO        true     C00BAB   ooA69kzOtNcAATXk
C00      slot 1002  LTO        true     C00CAC   ooA69kzOtNcAAwqN
...
C00      slot 1675  LTO        true     C00ZT1   ooA69kzOtSkAAQGF
C00      slot 1676  LTO        true     C00AT2   ooA69kzOtSkAAtZU
C00      slot 1677  LTO        true     C00BT3   ooA69kzOtSkABKsZ

Library     Bay Name   Slot Type  Total Slots  Free Slots
C00         bay 1      DLT        0            0
```

3. Verify that the volumes are available for DMF to use in the appropriate shelf volume group using the following dmvoladm(8) command:

```
dmfserver# dmvoladm -l libraryserver -c "list all and vg=VGnames"
```

For example:

```
dmfserver# dmvoladm -l maid_ls -c "list all and vg=vg_c00"
                                     DATA    EOT    EOT            WR/FR
VSN        VOLGRP LB    DATA LEFT    WRITTEN CHUNK  ZONE  HFLAGS   AGE
------------------------------------------------------------------------
C00A00     vg_c00 al    0.000000     0.000000   1      1    ---------   16m
C00A02     vg_c00 al    0.000000     0.000000   1      1    ---------   16m
...
C00Z08     vg_c00 al    0.000000     0.000000   1      1    ---------   16m
C00Z09     vg_c00 al    0.000000     0.000000   1      1    ---------   16m
```

Test the DMF System

This chapter describes how to test the DMF system:

- "Test that OpenVault Can Mount a Migration Volume" on page 55
- "Validate the DMF Configuration" on page 55
- "Enable Audits of the DMF-Managed Filesystems" on page 57
- "Test the DMF Configuration" on page 61

Test that OpenVault Can Mount a Migration Volume

To test that OpenVault can mount a volume from a library for the dmf application, enter the following ov_mount(8) command:

```
dmfserver# ov_mount -A dmf -V VSN
```

For example, to test that OpenVault can mount the volume with the VSN C00BAB and then exit:

```
dmfserver# ov_mount -A dmf -V C00BAB
Mounted C00BAB on /var/opt/openvault/clients/handles//tEyedOqkcssi
dmfserver# exit
exit
```

Repeat the steps in this section for each MAID shelf.

Validate the DMF Configuration

To verify the DMF configuration after the volumes have been created and imported into OpenVault, repeat the validation check in DMF Manager (as in step 7 of Chapter 4, "Configure DMF" on page 41) by right-clicking anywhere within the **Overview** panel and selecting:

Configure
   > **Validate Configuration**

You could instead run the dmcheck(8) command on the DMF server, which provides similar functionality. There should not be any errors or warnings. For example:

```
dmfserver# dmcheck -h

Checking DMF installation.
  Linux dmfserver 2.6.32.45-0.3-default #1 SMP 2011-08-22 10:12:58 +0200 x86_64 x86_64 x86_64 GNU/Linux - dmfserver
  SuSE-release: SUSE Linux Enterprise Server 11 (x86_64)
  SuSE-release: VERSION = 11
  SuSE-release: PATCHLEVEL = 1
  sgi-issp-release:    SGI InfiniteStorage Software Platform, version 2.4, Build 704r10.sles11-1104042004
  sgi-foundation-release:     SGI Foundation Software 2.4, Build 704rp10.sles11-1104242013
  lsb-release:
LSB_VERSION="core-2.0-noarch:core-3.2-noarch:core-4.0-noarch:core-2.0-x86_64:core-3.2-x86_64:core-4.0-x86_64"
  DMF version 5.5.0 rpm dmf-5.5.0-sgi250a74.sles11 installed.


Checking DMF config file
 Scanning for non-comment lines outside define/enddef pairs
 Scanning for DMF parameters without values
 Checking all objects for invalid names
 Checking base
 Checking daemon
 Checking policy space_policy
 Checking policy copan_policy
 Checking filesystem /dmi_fs
 Checking Library Server maid_ls
 Checking Drive Group dg_c00
 Checking Drive Group dg_c01
 Checking Volume Group vg_c00
 Checking Volume Group vg_c01
 Checking Migrate Group mg0
 Checking selection rules in policy space_policy.
 Checking selection rules in policy copan_policy.
 Checking Task Group daemon_tasks
 Checking Task Group dump_tasks
 Checking Task Group ls_tasks
 Checking for unreferenced objects

 Checking Task Group ls_tasks
 Checking for unreferenced objects
 Cross-checking LSs and task groups for duplicate VSNs
```

```
Checking other daemons.
 Checking OpenVault
 Checking chkconfig

No errors found.
No warnings found
```

If there are any errors or warnings, fix them and then rerun the validation. For more information, see the *DMF 6 Administrator Guide for SGI InfiniteStorage*.

Enable Audits of the DMF-Managed Filesystems

The DMF configuration file for COPAN MAID performs a nightly audit at 23:00 to check for database consistencies. To enable these audits of the DMF-managed filesystems, you must initialize the dmaudit(8) working environment. For more information about dmaudit, see *DMF 5 Filesystem Audit Guide for SGI InfiniteStorage*.

For example, on DMF server dmfserver (the characters **###** in the right margin highlight comments related to the steps below):

```
dmfserver# service dmf start                                          ### STEP  1

dmfserver# dmaudit                                                    ### STEP  2

This program must create a working directory named 'working_dir' to hold
several very large work files.  The files may be needed in subsequent
executions of this program, so the directory should be placed in a file system
that is not cleared frequently.

Please enter the full path name of an existing directory in which subdirectory
'working_dir' can be created (<CR> to quit): /dmf/home

You must select the file systems to be scanned when this program searches for
migrated files.  By default the list contains all file systems currently
mounted.

IF THERE ARE ANY FILE SYSTEMS CONTAINING MIGRATED FILES THAT DO NOT APPEAR IN
THIS LIST, EXIT THIS PROGRAM AND MOUNT THEM BEFORE CONTINUING!!!

Failure to do so could mean the loss of the data in those files, because this
```

program sometimes removes database entries if it cannot find matching migrated
files for them. File systems that you are sure do not contain migrated files
may be removed from this list to speed execution. Remember that using restore
to load files dumped from a DMF-configured file system can result in migrated
files in other file systems. If there is any question whether a file system
contains migrated files, leave it in the list.

```
Select:
   <view>     View the current file system scan list
   <edit>     Edit a fresh copy of the file system scan list
   <accept>   Accept the current file system scan list
   <quit>     Quit
```

Please enter your selection: **edit**   **### STEP 3**

```
/dmf/move
/dmf_fs
```

Delete /dmf/move   **### STEP 4**
```
/dmi_fs
```

```
Select:
  <view>       View the current file system scan list
  <edit>       Edit a fresh copy of the file system scan list
  <accept>     Accept the current file system scan list
  <quit>       Quit
```

Please enter your selection: **accept**   **### STEP 5**

```
MAIN MENU
---------
```

```
Select:
   <snapshot>   Take a snapshot and report status of file systems and databases
   <config>     Examine or modify configuration information
   <quit>       Quit
```

Please enter your selection: **snapshot**   **### STEP 6**

DAEMON DATABASE ERROR REPORT

```
----------------------------

No errors were discovered comparing the file systems against the daemon
database.


DATA MIGRATION CONFIGURATION
----------------------------

Data migration home directory:          /dmf/home
Data migration binaries directory:      /usr/sbin

Server name:                            daemon
Server home directory:                  /dmf/home/daemon
Server spool directory:                 /dmf/spool/daemon

Data migration daemon process ID:        19275

No MSPs are configured

Name        Type
--------    --------
maid_ls     dmatls
vg_c00      volume group   in Library Server maid_ls
vg_c01      volume group   in Library Server maid_ls
vg_c02      volume group   in Library Server maid_ls
vg_c03      volume group   in Library Server maid_ls
vg_c04      volume group   in Library Server maid_ls
vg_c05      volume group   in Library Server maid_ls
vg_c06      volume group   in Library Server maid_ls
vg_c07      volume group   in Library Server maid_ls
mg0         migrate group  Group Members: vg_c00
mg1         migrate group  Group Members: vg_c01

LIST OF FILE SYSTEMS SCANNED
----------------------------

/dmf_fs   xfs

MAIN MENU
---------
```

```
Select:
   <inspect>     Inspect and correct file system and database errors
   <report>       Reprint status report for the current snapshot
   <verifymsp>   Check the dmatls tape library server databases against the
                 daemon databases
   <snapshot>    Take a snapshot and report status of file systems and
                 databases
   <free>        Release all file space used by the current snapshot
   <config>      Examine or modify configuration information
   <quit>        Quit

Please enter your selection: quit                                          ### STEP 7
```

Comments:

1. Start the DMF service:

   dmfserver# **service dmf start**

2. Run the dmaudit(8) command to verify the filesystems:

   dmfserver# **dmaudit**

   In particular, set up the file system scan list and run a simple snapshot.

3. Enter edit to edit a fresh copy of the filesystem scan list.

4. Delete the /dmf/move filesystems, leaving just your DMF-managed filesystem
   (such as /dmf_fs).

5. Enter accept.

6. Enter snapshot.

7. Enter quit.

Test the DMF Configuration

To test the completed DMF configuration, do the following from the DMF server:

1. Change to a DMF-managed filesystem directory and create a test file of at least 1 MB. For example, to create the file mytestfile in the filesystem /dmi_fs:

   ```
   # cd /dmi_fs
   # /usr/lib/dmf/support/dmtwf -l 1m mytestfile
   ```

2. Compute the checksum of the test file by using the cksum(1) command. For example:

   ```
   # cksum mytestfile
   3368297266 1000000 mytestfile
   ```

3. Migrate the file and release its data blocks after the offline copies are made by using the dmput(1) command:

   ```
   # dmput -r mytestfile
   ```

4. Idle the DMF daemon so that the dmput request will take place immediately by using the dmdidle(8) command:

   ```
   # dmdidle
   ```

5. List the progressing status of the file by executing the dmls(1) command multiple times:

   ```
   # dmls -l
   ```

For example, the following repeated execution of dmls shows that mytestfile moves from migrating (MIG) to offline (OFL):

```
# dmls -l
total 980
-rw-r--r--  1 root      root    1000000 2010-11-05 12:36 (MIG) mytestfile
#
# dmls -l
total 980
-rw-r--r--  1 root      root    1000000 2010-11-05 12:36 (OFL) mytestfile
```

6. Retrieve the file by using the dmget(1) command:

```
# dmget mytestfile
```

7. Use dmls again to show that the file is now dual-state (DUL):

```
# dmls -l
total 980
-rw-r--r--  1 root      root    1000000 2010-11-05 12:36 (DUL) mytestfile
```

8. Verify that the checksum on the file has not changed:

```
# cksum mytestfile
3368297266 1000000 mytestfile
```

9. Delete the test file. For example:

```
# rm mytestfile
```

Maintenance Tasks

This chapter discusses the following:

- "Deleting a Volume from a RAID set" on page 63
- "Unformatting a RAID set" on page 65
- "Adding a COPAN MAID Cabinet or Shelf" on page 67
- "Verifying the OpenVault Components for a MAID Shelf" on page 67
- "Updating OpenVault Components After a Power Budget Change" on page 68
- "Removing OpenVault Components" on page 68
- "Stopping OpenVault Components" on page 69

Deleting a Volume from a RAID set

To delete a volume from a RAID set, do the following:

1. Determine if the volume has any data on it by executing the dmvoladm(8) command on the DMF server and examining the output for the DATA LEFT column:

   dmvoladm -l *libraryserver_name* -c "list *VSNlist*"

   For example, the following shows that C00A00 has data but C00A01 does not:

   ```
   dmfserver# dmvoladm -l maid_ls -c "list C00A00-C00A01"

                                       DATA      EOT   EOT              WR/FR
   VSN      VOLGRP LB    DATA LEFT    WRITTEN   CHUNK  ZONE  HFLAGS     AGE
   ------------------------------------------------------------------------
   C00A00  vg_c00 al   4847.408715  14932.408715    9     5  ---------   3d
   C00A01  vg_c00 al      0.000000   5500.000000    4     3  ---------   7d
   Database was not modified.
   ```

   If the volume is empty, skip to step 3.

2. If the volume has data on it, do the following on the DMF server:

   a. Flag the volume as sparse:

      ```
      dmfserver# dmvoladm -l LCPname -c "update VSN to hsparse on hro on"
      ```

      For example:

      ```
      dmfserver# dmvoladm -l maid_ls -c "update C00A00 to hsparse on hro on"
      Updated 1 record.
      ```

   b. Verify the flag settings, which should show r and s. For example:

      ```
      dmfserver# dmvoladm -l maid_ls -c "list C00A00"
                                          DATA    EOT   EOT            WR/FR
      VSN      VOLGRP LB    DATA LEFT    WRITTEN  CHUNK  ZONE   HFLAGS    AGE
      ------------------------------------------------------------------------
      C00A00   vg_c00 al   4847.408715  4847.408715    9      5   ---r---s-   3d
      ```

   c. Wait for the library server to merge data off the volume. This will take some
      time, depending on the amount of data on the volume and the number of
      available drives. Repeat step 1 until there is no data left on the VSN.

      Note: If the volume is damaged and the merge fails, use the dmemptytape(8)
      command to attempt to recover as much data as possible from the volume.

      ---

3. Delete the empty volume from the DMF database:

   ```
   dmfserver# dmvoladm -l LCPname -c "delete VSN"
   ```

   For example, for VSN C00A01:

   ```
   dmfserver# dmvoladm -l maid_ls -c "delete C00A01"
   Deleted 1 record.
   ```

4. Delete the empty volume from the OpenVault database by using the ov_vol(8)
   command from the OpenVault server:

   ```
   dmfserver# ov_vol -D -a dmf -v VSN
   ```

For example, if `dmfserver` is the OpenVault server:

```
dmfserver# ov_vol -D -a dmf -v C00A00
Volume deleted:
        volume name = 'C00A00', application name = 'dmf'
        cartridgeID = 'ooBC9k53kkIADV25', side = 'SideA', partition = 'PART 1'
```

Take note of the cartridge ID number for this VSN, which you will use in the following step.

5. Purge the cartridge ID from the OpenVault database by executing the `ov_purge`(8) command from the OpenVault server:

```
dmfserver# ov_purge -C 'CartridgeID'
```

For example, step 4 above shows that the cartridge ID for VSN C00A00 is `ooBC9k53kkIADV25`:

```
dmfserver# ov_purge -C 'ooBC9k53kkIADV25'

Are you sure you want to purge all information for cartridge with
cartridge ID = ooBC9k53kkIADV25 (Y/N)? y
Deleted partition PART 1
Deleted cartridge ooBC9k53kkIADV25
```

6. Delete the volume from a formatted RAID set by executing the `delete` operation of the `ov_copan` command on the node that owns the shelf:

```
ov_copan delete VSN
```

For example, if `node1` owns shelf C00, to delete the volume with VSN C00A00:

```
node1# ov_copan delete C00A00
delete VSNs on C00A
```

Unformatting a RAID set

After you delete all of the volumes on a RAID set (see "Deleting a Volume from a RAID set" on page 63), you can unformat the RAID set to remove all label and GPT partition information.

⚠️ **Caution:** Unformatting a RAID set will destroy all data on that RAID set. By default, the `unformat` operation will ask you to confirm the action if it is about to destroy existing data.

If there are unrecognized XVM slices on the RAID set, the `unformat` operation will fail rather than destroy the data. You must remove any such XVM slices manually using `xvm`(**8**) before unformatting the RAID set with `ov_copan`.

Enter the following:

ov_copan unformat *device_or_shelfID* **[-m** *raidlist***]**

For example, to unformat all RAID sets on shelf 1:

```
# ov_copan unformat C01
Device C01A is a DMF partition with 40 VSNs
Continuing will destroy existing data.
Continue (y or n)? y
Device C01B is a DMF partition with 40 VSNs
Continuing will destroy existing data.
Continue (y or n)? y
...
unformat C01A
unformat C01B
...
```

For example, to unformat RAID sets `W` and `X` on shelf 1:

```
ownernode# ov_copan unformat C01 -m W,X
Device C01W is a DMF partition with 40 VSNs
Continuing will destroy existing data.
Continue (y or n)? y
Device C01X is a DMF partition with 40 VSNs
Continuing will destroy existing data.
Continue (y or n)? y
unformat C01W
unformat C01X
```

Adding a COPAN MAID Cabinet or Shelf

Before you add a new COPAN MAID cabinet or additional shelves to your environment, you should verify that there are no existing OpenVault components that will conflict with the new shelf IDs you want to use. Do the following:

- If adding a new cabinet:

  # **ov_shelf precheck** *newcabinetID*

  For example, to check a cabinet ID of C1:

  # **ov_shelf precheck C1**

- If adding a new shelf to an existing cabinet:

  # o**v_shelf precheck** *newshelfID*

  For example, to check a shelf ID of C07:

  # **ov_shelf precheck C07**

For more details, see "Selecting Appropriate Cabinet Identifiers" on page 21.

Verifying the OpenVault Components for a MAID Shelf

To verify the existing OpenVault components for a formatted MAID shelf, enter the following from the node that owns the shelf:

ownernode# **ov_shelf check** *shelfID*

For example, if node1 owns shelf C07 and there are no errors, there will be no output:

node1# **ov_shelf check C07**
node1#

If there are errors, you may need to update the OpenVault configuration. See "Updating OpenVault Components After a Power Budget Change" on page 68.

Updating OpenVault Components After a Power Budget Change

If you increase or decrease the power budget, run the following command to update the OpenVault components:

```
ownernode# ov_shelf update shelfID
```

Removing OpenVault Components

If you want to remove the OpenVault components for a shelf, execute the following command from the node that owns the shelf:

```
ownernode# ov_shelf delete shelfID
```

To also delete any cartridge records cartridge/partition/volume/side records, include the -c option:

```
ownernode# ov_shelf -c delete shelfID
```

Removing these records can be time-consuming. The ov_shelf command will report its progress by printing a dot character every few seconds. For example:

```
ownernode# ov_shelf -c delete C01
.....
```

If the command is successful, there will be no other output.

Stopping OpenVault Components

Note: For instructions about starting and stopping services in an HA environment, see *High Availability Guide for SGI InfiniteStorage*.

To stop all of the local LCPs and DCPs, and the OpenVault server (if running on the server node):

```
dmfserver# service openvault stop
Stopping OpenVault ...
OpenVault stopped
```

To stop individual LCPs or DCPs, run ov_admin on the owner node and use the relevant menu selections:

```
ownernode# ov_admin

Name where the OpenVault server is listening? [dmfserver]

    OpenVault Configuration Menu for server "dmfserver"

      Configuration on Machines Running LCPs and DCPs
          1 - Manage LCPs for locally attached Libraries
          2 - Manage DCPs for locally attached Drives

      ...
```

For more information, see the ov_admin(8) man page.

Troubleshooting

This chapter discusses the following:

- "Displaying Information about MAID Shelves and RAID Sets" on page 71
- "Repairing OpenVault Components" on page 71
- "Unable to Delete a Volume" on page 72

Displaying Information about MAID Shelves and RAID Sets

To display summary information for the MAID shelves and RAID sets (LUNs):

```
ov_copan list [device_or_shelfID]
ov_copan listluns [device_or_shelfID] [-m raidlist]
```

To display more detailed information about RAID sets, volumes, or drives:

```
ov_copan listinfo [device_or_shelfID] [-m raidlist] [-v]
ov_copan listvols [device_or_shelfID] [-m raidlist]
ov_copan listdrives [device_or_shelfID] [-m raidlist]
```

For details about the variables, see "Summary of the ov_copan Command Syntax" on page 25. For more information, see the ov_copan(8) man page.

Repairing OpenVault Components

If you accidentally delete an OpenVault component, you can repair the damage by updating the OpenVault component information for the shelf:

```
ownernode# ov_shelf update shelfID
```

Unable to Delete a Volume

If you try to delete a volume that contains data, you will get an error message. For example:

```
dmfserver# dmvoladm -l maid_ls -c "delete C00A00"
delete: VSN C00A00 ignored - usage is not zero.
Database was not modified.
```

You must flag the volume as sparse in order to cause DMF to move the data to a new volume before you can delete it. See "Deleting a Volume from a RAID set" on page 63.

Glossary

ADMIN_EMAIL

The e-mail address to receive output from administrative tasks.

administrative directories

See *DMF administrative directories.*

AGE_WEIGHT

A floating-point constant and floating-point multiplier used to calculate the weight given to a file's age (for MSP/VG user filesystem).

AOR

The *always on region*, a section of COPAN MAID disk that will always be spinning and available for access. This region is not used by DMF.

base **object**

The configuration file object that defines pathname and file size parameters necessary for DMF operation.

bit-file identifier

(BFID) A unique identifier, assigned to each file during the migration process, that links a migrated file to its data on alternate media.

block

Physical unit of I/O to and from media, usually tape. The size of a block is determined by the type of device being written. A block is accompanied by a header identifying the chunk number, zone number, and its position within the chunk.

CACHE_DIR

The directory in which the VG stores chunks while merging them from sparse volumes.

`CACHE_MEMBERS`

The single VG or one or more MGs to be used as a fast-mount cache.

chunk

That portion of a user file that fits on the current media volume. Most small files are written as single chunks. When a migrated file cannot fit onto a single volume, the file is split into chunks.

configuration file object

A series of parameter definitions in the DMF configuration file that controls the way in which DMF operates. By changing the parameters associated with objects, you can modify the behavior of DMF.

configuration parameter

A string in the DMF configuration file that defines a part of a configuration object. By changing the values associated with these parameters, you can modify the behavior of DMF. The parameter serves as the name of the line. Some parameters are reserved words, some are supplied by the site.

configuration stanza

A sequence of configuration parameters that define a configuration object.

COPAN MAID

Power-efficient long-term data storage solution based on an enterprise massive array of idle disks (MAID) platform.

COPAN VTL

Power-efficient long-term data storage solution based on an enterprise MAID platform using a virtual tape library (VTL).

`COPAN_VSNS`

A parameter that specifies that the fourth character of the volume serial number (VSN) indicates the RAID in the COPAN VTL or COPAN MAID that contains the VSN. This specification applies for all VSNs in this library server.

daemon

A program that is run automatically by the system for a specific purpose.

daemon database

A database maintained by the DMF daemon.

`DATABASE_COPIES`

One or more directories into which a copy of the DMF databases will be placed.

DG

See *drive group.*

DMF administrative directories

The directories in which DMF stores databases, log files, journal files, and temporary files. The DMF configuration file specifies these directories using the following parameters:

```
HOME_DIR
JOURNAL_DIR
SPOOL_DIR
TMP_DIR
MOVE_FS
CACHE_DIR
STORE_DIRECTORY
```

DMF daemon

The program that accepts requests to migrate data, communicates with the operating system kernel in order to maintain a file's migration state, determines the destination of migrated data, and requests the return of offline copies.

DMF server

A node running the required DMF server software that provides DMF administration, configuration, and data mover functionality. (When using the Parallel Data Mover Option, data mover functionality is optional on the DMF server.)

drive

A hardware device that reads and writes data.

drive group

One of the components of an LS. The drive group is responsible for the management of a group of interchangeable drives located in the library. These drives can be used by multiple VGs and by non-DMF processes, such as backups and interactive users. However, in the latter cases, the drive group has no management involvement; the OpenVault mounting service is responsible for ensuring that these possibly competing uses of the drives do not interfere with each other. The main tasks of the drive group are to monitor I/O for errors, to attempt to classify them (as volume, drive, or mounting service problems), and to take preventive action.

dual-state file

A file whose data resides both online and offline.

`DUMP_FILE_SYSTEMS`

One or more filesystems to back up. If not specified, the tasks will back up all the DMF-managed user filesystems configured in the configuration file.

`DUMP_MIRRORS`

One or more directories in which to place a copy of disk-based backups.

fast-mount cache

A migration target with fast mount/position characteristics (such as COPAN MAID) that is used in conjunction with another permanent storage target (such as physical tape). Data copies are placed on the fast-mount cache and the permanent storage target at the time of initial migration.

file

An inode and its associated data blocks; an empty file has an inode but no data blocks.

`FILE_RETENTION_DAYS`

The age (in days) of a file that must not be deleted from the fast-mount cache.

`FMC_MOVEFS`

The specific scratch `MOVE_FS` directory to be used when moving files to be retained in the fast-mount cache.

`FREE_VOLUME_MINIMUM`

The minimum percentage of free volumes in the COPAN MAID or COPAN VTL fast-mount cache that will cause `run_fmc_free.sh`to begin freeing full volumes in order to meet the percentage set for `FREE_VOLUME_TARGET`.

`FREE_VOLUME_TARGET`

The percentage of volumes in the COPAN MAID or COPAN VTL fast-mount cache that `run_fmc_free.sh` tries to free when the `FREE_VOLUME_MINIMUM` threshold is reached.

HA

High availability.

`HOME_DIR`

The base pathname for directories in which DMF databases and related files reside.

inode

The portion of a file that contains the bit-file identifier, the state field, and the data pointers.

LCP

Library control program.

library server

A daemon-like process by which data blocks are copied onto a volume and that maintains the location of the migrated data. An LS can be configured to contain one or more drive groups. Each drive group contains one or more VGs. A VG is responsible for copying data blocks onto alternate media. A VG is capable of managing a single copy of a user file.

LS

See *library server.*

LS database

The database containing catalog and volume records associated with a library server (LS).

MAID

Massive array of idle disks.

managed filesystem

A DMAPI-mounted XFS or CXFS filesystem, configured in a `filesystem` object in the DMF configuration file, on which DMF can migrate or recall files. (When using the Parallel Data Mover Option, it must be CXFS.)

`MAX_PUT_CHILDREN`

The maximum number of write child (`dmatwc`) processes that will be simultaneously scheduled for the VG.

merging

See *volume merging.*

`MERGE_THRESHOLD`

The integer percentage of active data on a volume less than which DMF will consider a volume to be sparse and allow merging, which should be `0` for a `volumegroup` that is part of a fast-mount cache.

migrated file

A file that has one or more complete offline copies and no pending or incomplete offline copies.

migrate group

A logical collection of VGs that you combine into a set in order to have a single destination for a migrate request.

MULTIPLIER

The amount of data to be sent to a group member relative to the other members.

offline file

A file whose inode contains a bit-file identifier but whose disk blocks have been removed. The file's data exists elsewhere in copies on alternate media.

OpenVault

A storage library management facility that improves how applications can manage, store, and retrieve removable media.

owner node

The node that DMF will use to perform migrations and recalls on a given shelf. The node on which you run `ov_copan` becomes the owner node for that shelf. In an HA environment, ownership is transferred as part of HA failover.

parallel data mover node

A node, installed with DMF data mover software and underlying CXFS client-only software, that provides dedicated data mover functionality in addition to the DMF server, increasing data throughput and enhancing resiliency.

Parallel Data Mover Option

Optional software and licenses available for purchase that allow you to run parallel data mover nodes in order to increase data throughput and enhance resiliency.

parameter

See *configuration parameter*.

power budget

The maximum number of RAID sets that can be powered on at any one time.

RAID

Redundant array of independent disks.

RAID set

A group of 3 data disks and 1 parity disk on a MAID shelf.

RAID-set identifier

The first four characters of a VSN (which uniquely identifies a specific RAID set within a specific MAID shelf on a specific COPAN cabinet) that can be used as an argument to some OpenVault commands.

resource agent

The software that allows an application to be highly available without modifying the application itself.

shelf identifier

The first three characters of a VSN (which uniquely identifies a specific MAID shelf on a specific COPAN cabinet) that can be used as an argument to some OpenVault commands.

recall

To request that a migrated file's data be moved back onto the filesystem disk, either by explicitly entering the dmget(1) command or by executing another command that will open the file, such as the vi(1) command.

sparse volume

A volume containing only a small amount of active data.

task

A process initiated by the DMF event mechanism. Configuration tasks that allow certain recurring administrative duties to be automated are defined with configuration file parameters.

VG

See *volume group.*

volume group

A component of an LS that is responsible for managing a pool of volumes that are capable of storing single copies of user files. Multiple copies of the same user files require the use of multiple VGs. See also *LS*.

volume merging

The mechanism provided by the LS for copying active data from volumes that contain largely obsolete data to volumes that contain mostly active data. Volume merging is not appropriate for a fast-mount cache.

volume serial number

A label that uniquely identifies a specific volume.

VSN

Volume serial number.

write child

A data mover process that migrates data to secondary storage.

zone

A logical grouping of chunks. Zones are separated by file marks and are the smallest block-addressable unit on the tape volume. The target size of a zone is configurable by media type.

`ZONE_SIZE`

The parameter that specifies about how much data the write child should put in a zone.

Index

shelf mapping to device names, 23
shelf order, 2, 23
shelf update, 68
stop OpenVault components, 69
system overview, 2
system requirements, 16

T

testing, 55
theory of migration, 5
THRESHOLD, 51
troubleshooting, 71

U

unexpected records in the OpenVault database, 22
unformat, 66
unformatting a RAID set, 65
units, 27
unpriviledged instance of the dmf application, 36
update a shelf, 68

V

validation of DMF configuration, 55

validation of the configuration, 52
vg_c00 through vg_c07, 44
volume creation with ov_copan, 27
volume deletion from a RAID set, 63
volume groups, 44
volume serial number
    See "VSN", 13
volumes, 11
VSN, 13
VV volume identifier, 13

W

wildcard characters in ov_admin, 39
WWPN of shelf ports, 23

X

xvm, 13, 34, 66

Z

zone requirements, 16