



# Introduction

Hello

*This chapter is the starting point to configure the Discreet Filesystem, **wire**® networking and mountstone.*

## Summary

In this chapter, you learn about:

- “Documentation Set” on page 2
- “Using This Guide” on page 3
- “Installation Overview” on page 5
- “About “Stone + Wire”” on page 5
- “Getting More Help” on page 5

## Welcome

Welcome to the *Discreet Filesystem and Networking Guide*. This document explains how to configure and troubleshoot the Discreet Filesystem and **wire** networking. It also includes everything you need to know to configure and use the mountstone utility.

Here is a brief description of each of these components:

- The Discreet Filesystem is a required element for all Discreet Editing and Effects products. It provides scalable, resolution-independent storage for noncompressed material. It harnesses the power of **stone**® disk arrays.

- **wire** is an optional add-on component that enables the high-speed transfer of uncompressed video, film, and audio between Discreet workstations, over industry-standard TCP/IP and HIPPI networks.
- **mountstone** is a utility provided with all Discreet applications. It allows you to access unused space on your **stone** array, as if it were normal UNIX disk storage. Discreet installation CDs include SAMBA, a third-party utility to access mountstone partitions from a PC.

## Documentation Set

The documentation set required to install and configure your Discreet system includes:

- *Installation Guide*
- *Discreet Filesystem and Networking Guide*
- *Application Release Notes*

## Documentation Available Online

The complete documentation set is available in PDF (Portable Document Format) for online viewing and printing. Use Adobe Acrobat Reader™ to view and print the PDF files.

You can access the PDF files from any of the following locations.

### On your hard drive

The documentation PDF files are installed on the hard drive of your SGI machine when you install your Discreet application.

#### To access the PDF files installed on your hard drive:

1. Go to the product documentation directory. Type:  
`cd /usr/discreet/<product name>/documentation`
2. View the list of available documentation for the product. Type:  
`ls`  
and press **ENTER**.
3. To view one of the documentation PDF files using Adobe Acrobat Reader, type:  
`acroread <filename>`  
and press **ENTER**. For example, to view the *Discreet Filesystem and Networking Guide*, type:  
`acroread discreet_filesystem_networking_guide.pdf`

### On Tutorial CD 1, “interactive tutorial & documentation”

You can view and print the PDF files on CD1 using Adobe Acrobat Reader on a PC, an SGI, or a Macintosh.

#### To access the PDF files from an SGI:

1. Place CD1, “interactive tutorial & documentation” in the CD-ROM drive of your SGI machine.
2. Go to the *documentation* directory for your discreet product. To view the documentation for your product, position the cursor over the UNIX shell and type:  

```
cd /CDROM/documentation/<product name>
```
3. Perform steps 2 and 3 in the previous procedure to view the list of available documentation and to open the PDF files using Adobe Acrobat Reader.

### On [www.discreet.com](http://www.discreet.com)

Registered users can also find the PDF files on the Discreet web site, [www.discreet.com](http://www.discreet.com). Go to the Support page, open the Documentation Library, and download any of the available files.

### Using the PDF Documentation

The PDF files have several advantages over printed documentation:

- You can quickly search the entire document for any word in the text.
- You can click on cross-references, table of contents items, and page numbers of index entries to go directly to the related information.
- Bookmarks to the left of the document provide quick access to all the topics in the guide.

## Using This Guide

The *Discreet Filesystem and Networking Guide* consists of two thematic sections: *Discreet Filesystem* and *Networking*.

### Section 1 - Discreet Filesystem

This section describes how to configure the Discreet Filesystem and troubleshoot common problems. It also discusses how to recover from a hard disk failure. It includes the following chapters:

- Chapter 2, “Setting Up the Discreet Filesystem”
- Chapter 3, “Healing a Bad Disk in a stone Disk Array”
- Chapter 4, “Troubleshooting the Discreet Filesystem”

## Section 2 - Networking

This section describes how to configure your network to use HIPPI and **wire**, and troubleshoot common problems. It also describes how to install and configure mountstone. This section includes the following chapters:

- Chapter 2, “Setting Up wire”
- Chapter 3, “Setting Up a HIPPI Network”
- Chapter 4, “Troubleshooting wire”
- Chapter 5, “Installing and Configuring mountstone”
- Chapter 6, “Accessing mountstone from a PC”

## Intended Audience

Basic knowledge of UNIX and computing in a professional video/film production environment is assumed throughout this guide. Do not attempt to carry out the procedures outlined in this document if you are not familiar with UNIX or computer hardware issues. Please contact Discreet Customer Support if you require further assistance (see “Getting More Help” on page 5).

**NOTE:** Most procedures described in this guide require root privileges (super-user).

## Notation Conventions

A number of style conventions are used throughout this user’s guide. These conventions and examples of their use are shown below.

Convention:	Example:
Text that you enter into a UNIX shell appears in Courier bold.	<b>swf-tweak</b>
Variable names appear in Courier, enclosed in angle brackets.	<file_name>
Feedback from the IRIX operating system appears in Courier.	limit coredumpsize
Directory and file names, and UNIX utility names appear in italics.	<i>/usr/discreet/bin</i>

## Installation Overview

The Discreet Filesystem and **wire** networking (optional) are automatically installed during the product installation (refer to your software *Installation Guide* for more information).

If you did not initially purchase **wire**, you can obtain a licence anytime and immediately access **wire** functionality by activating the licence (see “wire Licencing” on page 73).

For information about installing mountstone see “Installing mountstone” on page 121.

## About “Stone + Wire”

In the past, the Discreet Filesystem and **wire** were sometimes referred to collectively as “Stone + Wire”. This is because they are developed and implemented together via a single product labelled “SW\_082” (version 082).

## Getting More Help

If you need more information, contact Discreet Customer Support at one of the numbers below. You can also send queries by E-mail.

### Discreet Customer Support

Global:	(514) 954-7199 (Country code = 1)
North America:	1-800-92LOGIC (925-6442)
Fax:	(514) 954-7254
Mail:	discreet.support@autodesk.com
WWW:	www.discreet.com

[illegible]







# Setting Up the Discreet Filesystem

*Getting some throughput*

*The Discreet Filesystem lets you create soft partitions that provide scalable, resolution-independent storage for noncompressed material.*

## Summary

In this chapter, you learn about:

- “Storage Concepts” on page 10
- “Discreet Storage” on page 11
- “Configuration Utility” on page 13
- “Configuring the Disk Array(s)” on page 20

## About Setting Up the Discreet Filesystem

The Discreet Filesystem is a central and required component of the Discreet storage system. It harnesses the power of **stone** disk arrays by storing images of different resolutions without wasting storage capacity or degrading performance.

Before you can configure the Discreet Filesystem, make sure of the following:

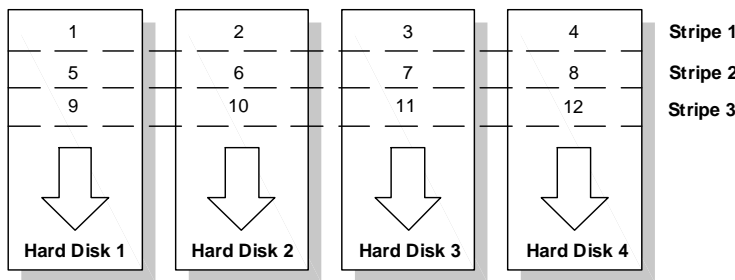
- The **stone** disk array is connected and functional (refer to the *Hardware Configuration Guide* that was shipped with your **stone** disk array for more information).
- The software is installed (refer to your application’s *Installation Guide* for more information).

## Storage Concepts

Although a comprehensive overview of storage concepts is beyond the scope of this guide, you should be familiar with the basic storage concepts described in the section before you configure the Discreet Filesystem.

### RAID

RAID combines many hard disks into a single logical disk to obtain I/O performance levels that cannot be obtained with a single disk. During a typical write operation, the data spans across multiple disks simultaneously (RAID 3 - Parallel transfer with parity).



The space available on each drive is divided into stripes. When the application writes to the logical drive, data is sent to each physical drive in round-robin fashion. For maximum throughput, the Discreet storage system uses dynamic striping to optimize stripe size according to image resolution. Striping units are created on-demand and many striping units can coexist on the same disk, in any order.

In addition to improving I/O performance, RAID can provide fault-tolerance features. Discreet storage products use RAID 3, which provides fault-tolerance by storing parity information on one (or more) disks. If a drive fails, the parity information can be used to rebuild the defective drive. This is done by calculating the exclusive OR value of similarly positioned stripes on the remaining drives.

Conceptually, this is similar to the following:

$$X + 1 = 3$$

Where X is the value of a stripe on a defective drive, 1 is the value of a stripe on healthy drive and 3 is the parity value for this stripe on the parity drive. Obviously, by knowing the value of the stripes on the healthy drive and the parity drive, we can then easily infer the value of the faulty drive. However, if more than a single drive fails, it is no longer possible to determine the exact value of the faulty drives.

**NOTE:** stone 1000 arrays that use only 4 disks do not provide fault-tolerance.

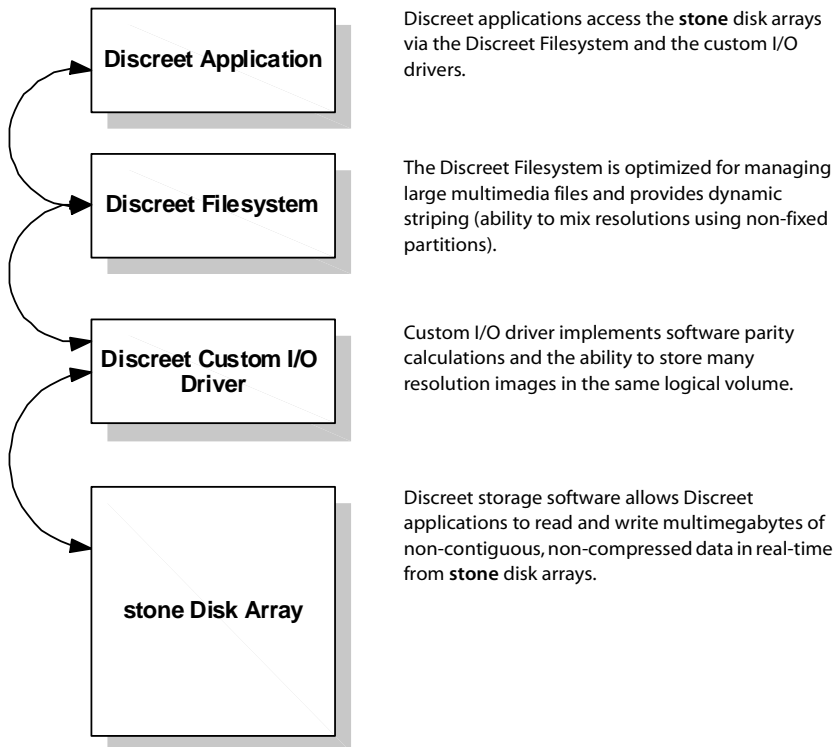
## Soft Partitions

Soft partitions let you store groups of images of the same resolution and bit-depth into logical units. Because stripe size is determined dynamically, there is no need to reserve disk space exclusively for images of a given resolution. Therefore, the size of soft partitions adjusts dynamically (up or down) according to your needs.

This means that clips can be moved from one partition to another with ease. A user working in an NTSC soft partition can open a clip from a 12-bit film partition; the clip is reformatted on-the-fly. With soft partitioning, all unused storage is equally available to all partitions. Soft partitions are created within the main Discreet application.

## Discreet Storage

The Discreet storage system consists of the seamless integration of several components:



### Custom I/O Drivers

The custom I/O drivers are designed for use with **stone** disk arrays. They provide the following services:

- Parity calculations in software
- Optimization of storage for multiple image resolutions
- Dynamic striping
- Maximized storage capacity

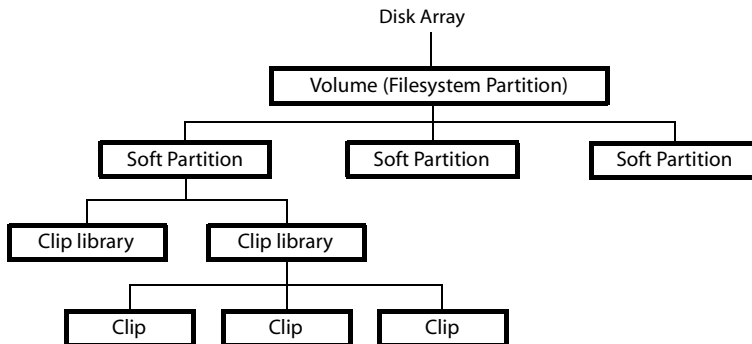
These drivers are installed automatically and do not need to be configured manually. Their use is invisible to the end-user.

### Discreet Filesystem

The Discreet Filesystem is a UNIX kernel driver that is used to control the storage of material on a **stone** disk array. It replaces SGI's XFS for **stone** arrays, and is designed to interact with Discreet's custom I/O drivers. When you configure the Discreet Filesystem, you decide how many volumes you will use and which disks will serve as the parity disk and optional spare.

### Framestore Organization

The term *framestore* refers to a disk array configured to use the Discreet Filesystem. The following is an example of a framestore using one volume:



You can configure a volume to use the Discreet Filesystem, typically at installation time. Each volume is divided further into soft partitions. You store your clip libraries in the soft partitions. Both soft partitions and clip libraries are created from within your Discreet applications, according to your preference.

# Configuration Utility

The configuration utility is a graphical application used to configure Discreet Filesystem volumes. This application must be run from a super user account (root). For step-by-step procedures to configure your disk array, see “Configuring the Disk Array(s)” on page 20.

**To start the configuration utility from the command line:**

- 1. Go to the appropriate directory by typing:

```
cd /usr/dscreet/sw
```

- 2. Type:

```
config [-l <filename>] [-z <zoom>]
```

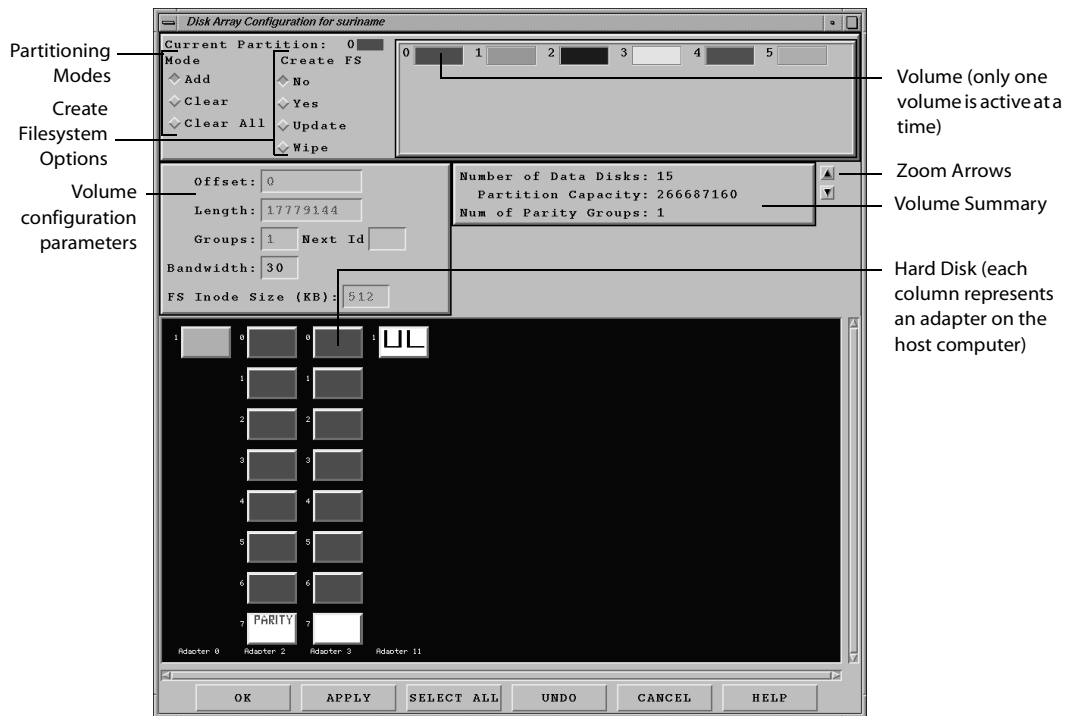
Where:	Is:
<filename>	The Stone+Wire License filename. Defaults to /etc/license. This parameter is optional.
<zoom>	The zoom value for the disk area. Zoom must be an integer. This parameter is optional.

A dialog box with three pull-down menus appears. Each menu contains a single menu item.

Select:	To:
System   Exit	Exit the application without running the configuration utility.
DiskArray   Configuration	Start the configuration utility.
Help   Version	Display the version of the custom Discreet I/O drivers (swr.o and swfs.o).

- 3. Choose Configuration from the Disk Array menu. The configuration utility appears.

## 2 Setting Up the Discreet Filesystem



### Partitioning Modes

Partitioning modes are used to add and remove disks from a volume.

Select:	To:
Add	Add the disk to the volume.
Clear	Remove a disk from the current volume.
Clear All	Removes all volumes from a disk.

#### To add or remove disks:

1. Select the appropriate volume and mode.
2. Click on a disk in the disk area. You can also drag a selection box to select multiple disk.  
See also "Hard Disk Area" on page 17.

### Create FS (Filesystem) Options

These options only take effect when you click the OK or Apply buttons.

Select:	To:
No	Create a new configuration without creating a filesystem (applies the same striping order as the previous configuration).
Yes	Create a new configuration and filesystem (striping order is based on the target and adapter values). This will destroy any existing data in this volume. To use this option, all volumes must be unmounted.
Update	Not active.
Wipe	Remove the previous filesystem from this volume. This button is only relevant if the framestore is to be used in raw mode (i.e. using swr00 as the framestore keyword instead of stonefs in the application).

### Buttons

The effect obtained with OK or Apply depends on the current Create FS options and the volume configuration parameters.

Select:	To:
Ok	Save the configuration (including disk striping order and disk status) to disks. Creates or wipe the filesystem depending on the "Create FS" radio buttons. This will close the configuration utility.
Apply	Save the configuration (including disk striping order and disk status) to disks. Create or wipe the filesystem depending on the "Create FS" radio buttons.
Select All	Selects all disks that are available for the current volume.
Cancel	Close the configuration utility without modifying the configuration. The configuration is not reset unless you exit the configuration utility. Note that disk status (i.e. dead, well, or sick) is saved as soon as you toggle the dead or well flag. Clicking Cancel does not undo this action.

Select:	To:
Undo	Undo the last action.
Help	Display the UNIX man page for the configuration utility.

**NOTE:** You must unmount any existing volumes before you can click OK or Apply. If the file system is mounted, an error message is displayed. For more information, see “Create and Configure Volume(s)” on page 21.

## Volume Configuration Parameters

### Offset

The volume’s physical starting point on each disk in blocks of 512K. In most cases, it starts at 0. Only change this value if you want to more than 1 volume on a disk.

### Length

The physical size in blocks of 512 bytes of the filesystem partition (volume). Set this value to 0 to use all available spare disks.

### Groups

Not in use.

### Next Id

Not in use.

### Bandwidth

Bandwidth describes how many megabytes per second are available to the system for the current volume. This is calculated by multiplying the number of adapters by the bandwidth of each adapter in MB/sec.

For example:

- Fibre Channel configurations with 2 XT-FC-2P adapters have a total bandwidth of 400 MB/sec (4 controllers x 100 MB/sec per controller).
- SCSI configurations with 1 MSCI board (4 adapters) have a total bandwidth of 180 MB/sec (4 controllers x 45 MB/sec per controller)

We recommend using the default value of 100 x number of adapters.



### FS Inode Size (KB)

Each Inode contains information about a single frame stored in the volume. The FS Inode Size value is used to calculate the total number of Inodes available per volume (for example, 1 Inode for every 512 KB of filesystem space).

In general, the number of Inodes should bear some relation to the frame size, so that Inodes are assigned at the same rate as frames. If a filesystem is exclusively filled with HDTV frames and the Inode Size value is set to 512, only 10% of the Inodes are used and therefore, some space is wasted by unused Inodes. Inode Size was implemented as a means of optimizing the filesystem for HDTV resolutions.

Set the Inode Size value according to the following guidelines:

Total Storage Capacity	Recommended Value
<800 MB	512
>800 MB	1024

The default is 512 and the maximum value is 2048. The FS Inode Size value is rounded up to the nearest 512 KB block (i.e. 513 rounds up to 1024).

### Volume Summary

The following information is displayed in the Volume Summary:

- Number of disks assigned to the current volume.
- Total capacity of current volume expressed in blocks of 512 bytes.
- Number of parity devices assigned to the current volume.

### Zoom Arrows

The Zoom arrows let you increase or decrease scaling in the hard disk area.

### Hard Disk Area

This area is a representation of all disks currently available to your system, including system and audio disks.

Cyan	SGL-mounted filesystem
White	Disk is not part of a Discreet Filesystem partition (usually a spare disk, audio disk or other available device).

Grey	Disks that were present at boot time, but were missing or unavailable when the Configuration utility was started.
UL	Disks that are not licenced for use with the Discreet Filesystem (e.g. audio disk).
S	Stands for "Sick" disk. Sick disks are read protected to prevent the main application from writing on the spare disk while the heal procedure is in progress. The spare disk is labeled as Sick automatically during the heal procedure (see Chapter 3, 'Healing a Bad Disk in a stone Disk Array').
D	Stands for "Dead" disk. When the system detects a problem with a disk, it automatically labels it as Dead. You cannot read or write to a Dead disk.
Parity	Disk contains parity information.
Other Colours	Indicate the disk's volume.

### Disk Menu

The disk menu is a pop-up menu accessed by right-clicking on a disk. It contains the following commands:

Command:	Description:
Partition Info	See "Partition Info" on page 19
Set Sick	The spare disk is set to Sick (writable, read protected) automatically during a heal procedure. This command should not be used.
Set Dead	Sets a disk as Dead (not writable or readable). See "Image Appears Striped Although No Drives Are Marked as Dead" on page 47.
Set Well	Makes a disk writable and readable. See "stone Disk Array Was Accidentally Shut Down" on page 47.

Partition Info

The Partition Info dialog box is accessed from the disk pop-up menu.



Item:	Description:
Controller number	Host adapter number.
Disk address	Position of disk on the host adapter.
Disk size	Size of disks in GBs and blocks of 512 KB.
Vendor	Hard disk vendor (STON+WIR indicates that the disk is licenced for use with the Discreet Filesystem).
Serial number	Hard disk serial number.
Active partititions	Lists all UNIX and Discreet Filesystem volumes (labeled S+W Data) for the current hard disk drive.
Inactive area	Displays the unused area of the current hard disk drive. In almost all cases this area will be empty since the entire disk is usually reserved for the Discreet Filesystem.
Delete	This removes a hard disk from the selected Discreet Filesystem volume.

## Configuring the Disk Array(s)

To set up the Discreet Filesystem on **stone** disk array(s), you must perform the following procedures:

- Verify the information in the host configuration file (*sw\_host\_map*).
- Determine how many volumes you will create.
- Create volumes and assign parity using the Discreet Storage configuration utility.

### Verify the Host Configuration File

Verify that the host configuration file (*sw\_host\_map*) contains the correct host information. The host configuration file is automatically created when the Discreet Filesystem is installed, and contains default host information.

**NOTE:** This is important even if you are not using **wire** networking.

#### To verify the host configuration file:

1. Log in as root.
2. Go to the file directory by typing  

```
cd /usr/discreet/sw/cfg
```
3. Type  

```
jot sw_host_map
```
4. The [HOSTS] section has a default line with the following information:

```
HOST=<hostname> HADDR=<hostIPAddress> ID=<uniqueID>
```

Where:	Is:
<hostname>	The name of the host machine.
<hostIPAddress>	The IP (Internet Protocol) address of the host machine. The address consists of a series of four numbers separated by a period.
<uniqueID>	The Discreet Filesystem unique identifier. The last field of the IP address is automatically assigned as the unique Discreet Filesystem ID. You can change this ID if you choose, however, this ID must differ from all other machines on the network.

For example:

```
HOST=Beet HADDR=192.0.2.30 ID=30
```

**NOTE:** The unique Discreet Filesystem ID is automatically assigned. If you are going to change it, do it before configuring your framestore. When you configure the framestore, the ID is stamped on each disk. If you change the ID at a later date, you must also recreate the Discreet Filesystem. Be advised that recreating the Discreet Filesystem will destroy soft partitions and clips stored on the framestore.

## Determine the Number of Volumes You Need

Typically, you will configure your disk array(s) as a single volume. If you have more than one array, this links the arrays together. If you add a **stone** to your system at a later date, simply create another volume for the new **stone**.

## Create and Configure Volume(s)

Once you verify the contents of the Host Configuration file, you can create your disk array volume(s) and configure the volume(s) to use the Discreet Filesystem.

**NOTE:** The terms *hard partition* and *volume* are interchangeable. They should not be confused with *soft partitions* which are part of the filesystem but are created in the main Discreet application. Soft partitions are not visible in the configuration utility.

### To create and configure volume(s):

1. Log in as root.
2. Stop the **wire** network daemon by typing  

```
/usr/discreet/sw/sw_stop
```
3. If you have already configured a Discreet Filesystem, you must also unmount the Discreet Filesystem. To unmount a specific filesystem, type  

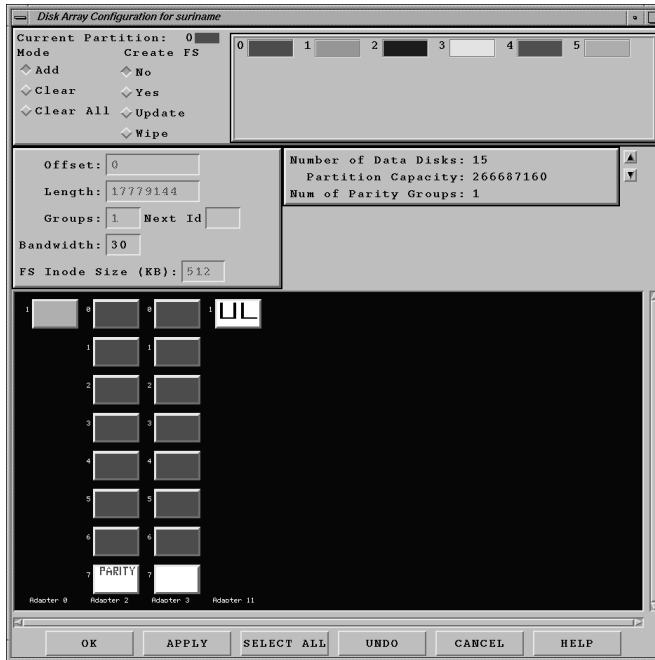
```
/usr/discreet/sw/sw-unmount <partition number>
```

 To unmount all filesystems, type  

```
/usr/discreet/sw/sw-unmount all
```
4. Start the configuration utility by typing  

```
/usr/discreet/sw/config
```
5. A menu appears with the choice of System, DiskArray, or Help. Click on DiskArray and then select Configuration.

The disk array configuration window appears (see “Configuration Utility” on page 13 for descriptions of all interface elements).



**Repeat the following sequence of steps for every volume you want to create:**

1. Select a volume by clicking one of the coloured volumes in the upper-right corner of the window.

**NOTE:** You must start with volume 0 to select volumes in sequential order.

2. Under Mode in the upper-left corner of the window, click Add.
3. Under Create FS in the upper-left corner of the window, click Yes.



**WARNING:** Using "Create FS: Yes", then OK or Apply will delete all existing data on the selected disks.

4. In the Offset and Length fields, verify the values to create your volume.

If you are using the entire disk array for the volume, make sure that the offset value is 0 and the length value is 0.

If you are creating more than one volume, enter the appropriate offset and length values for the current volume. For example, if the size of the disk is 18 GB, and you want to create two equal volumes, you would enter the following values (each block is 512 bytes in length):

- For the first volume, the offset is 0, and the length is 18,000,000 blocks.
  - For the second volume, the offset is 18,000,000, and the length is 0 (0 uses the available space left on the drive).
5. In the FS Inode Size box, change the default value to 1024 if the total storage capacity of the volume is greater than 800 MB.
  6. Select the disks that you want to add to the volume.

To:	Do This:
Select a single disk	Move the mouse cursor over the disk in the Disk Display area and click the left mouse button. See "Configuration Utility Error: 'Offset/Length out of range (X-+XXXXXXXX)'" on page 46.
Select multiple disks	Click the left mouse button in the Disk Display area, but not on a specific disk. Then drag the cursor over the area of the disks you want to select (marquee selection). See "Configuration Utility Error: 'Offset/Length out of range (X-+XXXXXXXX)'" on page 46.
Select all disks	Click the Select All button at the bottom of the window.

To avoid bandwidth issues, we recommend that you select an equal number of disks per adapter; that is, select the same number of disks in every column.

Before you save a volume or volumes, you can remove a drive from the volume by clicking either Clear or Clear All, and dragging your mouse over the affected disks in the Disk Display area. Use Clear to remove a drive from the volume, and Clear All to remove a drive from all volumes.

7. Use the following table to determine recommended spare and parity usage for your **stone** disk array.

Disks:	Use:
4 disks	No spare, no parity
8 disks	No spare, 1 parity
10 disks	No spare, 1 parity
16 disks or more	1 spare, 1 parity

8. Add a parity disk to the volume. Move the cursor over the disk and click the middle mouse button.

The word Parity appears in the selected disk.

**NOTE:** You can add a parity disk without recreating the file system.

9. Save your changes by either clicking Apply or OK.

Apply saves your configuration and keeps the window open so you can create another volume.

OK exits from the configuration window and brings you to the menu with the choice of System, Disk Array, or Help. Click on System and then Exit.

10. Restart the **wire** network daemon by typing:

```
/usr/discreet/sw/sw_start
```



# Healing a Bad Disk in a stone Disk Array

*Medicine for your disk array*

*If you encounter a bad disk in your framestore you need to use the heal utility to recover the data from the failed disk. This chapter explains how to use the heal utility to safely recover material on your framestore.*

## Summary

In this chapter, you learn about:

- “Error Message in Discreet Applications” on page 26
- “Using the Correct Driver Version” on page 26
- “Using Heal” on page 27
- “Alternate Heal Procedure” on page 36
- “Standardizing Block Sizes Across Disks” on page 38

We recommend that you read and understand this chapter before attempting to heal a bad drive. This will ensure smooth recovery of your data.

## About Heal

This chapter is intended as a general heal procedure. Since each heal procedure has its own particularities, we suggest that you contact Discreet Customer Support if you encounter any difficulties. Be sure to carefully record all the steps you have taken before calling customer support. Undoing a heal that did not work without knowing which steps were taken is almost impossible.



**WARNING:** The heal procedure may have changed since this guide was printed. Check the Hardware section of the Documentation Library in the Customer Support area of the Discreet web site for a more recent version of this procedure.

**NOTE:** If you have a dead disk in your array but you still can access your images, you should make an archive of your images before attempting the heal.

For more detailed information about physically adding or removing hard disk drives from a disk array, refer to the disk array's hardware configuration manual, if available.

## Error Message in Discreet Applications

The following error message in the status bar of a Discreet application is an indication of a bad disk in the framestore:

```
VOLUMEMGT: WARNING: The disk array has 1 bad disk(s)
```

## Using the Correct Driver Version

This chapter can be used with releases as early as SW\_064. However, release SWM\_082 or later is strongly recommended.



**WARNING:** Running a heal on a D9 disk array with SW product below SW\_064 is not recommended and could lead to data loss. Contact Discreet Customer Support if you need to upgrade your SW product.

**To find out the version of the SW product you are running:**

1. In a UNIX shell type:

```
cd /usr/discreet/sw
```

2. Type:

```
./sw_ping -h <local host>
```

Where <local host> should be replaced by the name of your machine. The version number of your SW product is located on the first line of the shell output.

**Example:**

```
Using stonefs 1.05 Release; ProductSWM_082
```

## Using Heal



**WARNING:** Never abort a heal. Doing so could result in loss of data.

### Using heal with a Spare Disk in the Framestore Configuration

Use this procedure if you already have a spare drive in your disk array configuration. If you have a spare drive outside of your disk array, refer to “Alternate Heal Procedure” on page 36. If you do not know if you have a spare drive in the configuration, start with this procedure.

### Get Disk Information

Before running the heal utility, you need to get information about the dead disk, the spare disk (if any) and the parity disk.

#### To get disk information:

1. Log in as root.
2. In a UNIX shell, type:  
`cd /usr/discreet/sw`
3. Open the Configuration utility by typing:  
`./config`
4. Select Configuration from the Disk Array menu.
5. Locate a spare disk (white disk).

**NOTE:** You may need to zoom out to see all your disks. Use the zoom control arrows at the upper right of the window.

6. Write down the adapter number and the ID number of the spare disk.

• Spare adapter: \_\_\_\_\_

• Spare SCSI ID: \_\_\_\_\_

If you do not have a spare disk (either inside or outside your array), you cannot run the heal procedure. Contact Discreet Customer Support for assistance.

**NOTE:** If you have a spare disk that is not already installed in your disk array, you will need to remove the bad disk and put the spare in its place before running the heal. See “Alternate Heal Procedure” on page 36.

7. Locate the dead disk. The dead disk is indicated by D or S. It may also be grey.

**NOTE:** If you do not see a disk marked with a D or S, you will have to locate it manually. For more information, refer to “Locate a Bad Disk Manually” on page 29.

If you have a dead disk that is completely missing from the inventory, look in the shell where you started the Configuration utility. It will report the serial number of any missing disks. You can refer to the bad disk by serial number (instead of by adapter and ID number) to run the heal.

8. Write down the adapter number and the ID number of the dead disk.

- Dead adapter: \_\_\_\_\_
- Dead SCSI ID: \_\_\_\_\_
- or Dead Serial Number: \_\_\_\_\_

9. Locate a good data disk (usually a red disk).

**NOTE:** Your data disks may have other colours if you have multiple volumes.

10. Write down the adapter number and the ID number of a good data disk.

- Good adapter: \_\_\_\_\_
- Good SCSI ID: \_\_\_\_\_

11. Locate your parity disk (disk with the word PARITY on it).

**NOTE:** If you do not have a parity disk, you cannot run the heal. You will need to replace the bad disk with the spare disk and remake the filesystem. Contact Discreet Customer Support for assistance.

If you have multiple volumes (for example a red and a green volume), you should see the word “Parity” for each volume.

12. Write down the adapter number and the ID number of your parity disk.

- Parity adapter: \_\_\_\_\_
- Parity SCSI ID: \_\_\_\_\_

## Locate a Bad Disk Manually

If the bad disk does not appear in the Configuration utility, do one of the following:

- Open the file `/var/adm/SYSLOG` and look for disk-related error messages.
- Remove the covers from the disk arrays and look at the disk's LED lights:
  - In Fibre Channel disk arrays, if an LED light is constantly off, it is probably the dead disk.
  - In all other disk arrays, if an LED light is constantly on, it is probably the dead disk.

## Verify the Spare Disk

Before you use heal, you need to verify if the spare disk is suitable for recovery.

1. While still in the Configuration utility, right-click on a spare disk (white disk).
2. Select "Partition Info".
3. Write down the following values:

- Disk size in blocks: \_\_\_\_\_

Example:

Disk Size 4.0 GB 8383114 blocks

- Length of XFS partition 7 in blocks: \_\_\_\_\_

Example:

Id	Offset	Length	Description
7	2575	2058970	XFS

4. Right-click on a good drive (a red disk, if you only have one volume).
5. Select "Partition Info".
6. Write down the following values:

- Disk size in blocks: \_\_\_\_\_
- Length of "S+W data" partitions(s) in blocks: \_\_\_\_\_

**NOTE:** If you have more than one partition labeled "S+W data," take down the sum of the length of all the "S+W data" partitions instead of only partition 0.

7. Repeat steps 1-6 for the spare disk and compare the data drive with the spare drive.  
Your spare drive should be partitioned the same way as the data drive. It is normal for a spare drive not to have partitions labeled "S+W data". Make sure that both your spare and your

good disks have the same type of UNIX partitions (either XFS or EFS). Having mixed EFS/XFS configurations is not recommended.

The length of partition 7 on the spare disk should be equal to or greater than the length of partition 0 (or the sum of all the “S+W data” partitions) on your good disks. If it is smaller, then you should contact Discreet Customer Support to get a proper spare drive.

Alternately, if the length of partition 7 on the spare disk is smaller but the total disk size is equal to or greater than the disks already in the partition, follow the instructions in “Standardizing Block Sizes Across Disks” on page 38.

8. To close the Configuration utility, click Cancel and choose Exit.

## Back Up the Volume Header Information

1. Go to the tools directory by typing:

```
cd /usr/discreet/sw/tools
```

2. Back up your volume header by typing:

```
./swr_get_config <good controller> <good target> header.bak
```

**NOTE:** In the above example “good adapter” and “good target” refer to the SCSI controller and ID number of a data drive that is neither bad nor spare.

3. Verify if the backup worked properly by typing:

```
jot header.bak
```

The *swr2cfg* file should be visible in jot. If it is, close the jot window. If not, try to do the backup again. Attempting to do a heal without a backup of the volume header is not advised.

## Run the Heal Utility

1. Go to the tools directory by typing:

```
cd /usr/discreet/sw/tools
```

2. Type:

```
./heal -h <dead controller>,<dead id> -n <spare  
controller>,<spare id>
```

Example:

If the dead disk is controller 4, ID 3 and the spare is controller 3, ID 5, type:

```
heal -h 4,3 -n 3,5
```

**NOTE:** If you do not know the adapter and ID of your dead disk, you can use its serial number in the heal command:

```
./heal -s <dead serial #> -n <spare controller>,<spare id>
```

**NOTE:** Wait until the heal is complete. The duration of a heal depends on the size of the volume and the bandwidth of the controller(s).

Sample Output:

```
./heal: There are 1 partition[s] to be healed
Partition number 0 will take about 58 minute[s] to heal
Would you like to heal it? (y/n) y
Total Time = 58 Minute[s]
Defaulting SW_HEAL_NUM_DISK_IO_PROCS to 2
Partition number 0 will take about 58 minute[s]
Heal In Progress:
0% ----- 100%
.....
./heal: Disk heal successful on partition 0
```

## Verify the Heal Operation

1. Open the Configuration utility by typing:  
**cd /usr/discreet/sw**
2. Type:  
**./config**
3. Select Configuration from the Disk Array menu.
4. Make sure your spare drive is no longer white and that the disk that was previously marked as dead is now white.
5. To close the Configuration utility, click Cancel and choose Exit.
6. Check if the partition is mounted by typing:  
**./swf-df**

Sample output:

```
Partition      free %use  purgeable %use  ....
      0      2015764    2          0    0  ....
```

**NOTE:** If you do not see numbers, the filesystem partition is unmounted. Try to mount the filesystem partition manually using the `./sw-mount all` command.

7. Start the software and verify that the images are OK.
- If there are stripes in the images, or if you have an error on startup, contact Discreet Customer Support. If everything looks OK, go to the next section.

**NOTE:** At this point, the software may still generate an error message stating that a bad disk is present. This is normal since the dead disk is still part of the configuration even though it is no longer used by the system.

Physically Locate the Bad Disk

1. Log in as root.
2. Type:  
**fx -x**
3. Press **ENTER** at the “device-name” prompt.
4. Provide the following information when prompted and press **ENTER**:

ctrlr#	Adapter ID number
drive#	Drive ID Number
lun#	Type “0”

5. At the “fx” prompt, type:  
**exe**
6. At the “fx/exercise” prompt, type:  
**b**
7. Simply press **ENTER** for the remaining four prompts.
8. Remove the covers from the framestore.  
The dead disk's LED should be flashing rapidly.



9. Physically mark the disk as dead (stick a Post-It on it for example).

**NOTE:** It is possible that your disk is damaged in a way that will prevent the LED from lighting up. If the bad disk is completely missing from the configuration, you can locate its serial number in the shell where you launch the Configuration utility. You can use the serial number to physically locate the dead drive. Alternately, you may have to light up every other drive. The drive that never lit up is your bad drive.

10. Stop the butterfly test by pressing **CTRL + C**

11. Exit the FX utility by typing:

**/exi**

and press **ENTER**.

## Physically Locate the Spare Disk

1. Log in as root.
2. Type:  
**fx -x**
3. Press **ENTER** at the “device-name” prompt.
4. Next, provide the following information when prompted and press **ENTER**:

ctlr#	Adapter ID number.
drive#	Drive ID Number
lun#	Type “0”

5. At the “fx” prompt, type:  
**exe**
6. At the “fx/exercise” prompt, type:  
**b**
7. Simply press **ENTER** for the remaining four prompts.
8. Remove the bezel from the framestore.  
The spare disk's LED should be on.
9. Physically mark the disk as spare (stick a Post-It on it for example).
10. Stop the butterfly test by pressing **CTRL + C**.

11. Exit the FX utility by typing:

```
/exi
```

and press **ENTER**.

## Remove the Bad Disk From the Array

The procedure for removing a defective disk differs according to the type of disk array. If you have a STONE 1000 or STONE 2000 array, you must shut down the array. If you have a STONE 3000, STONE 4000, D Series, or R Series array, you could leave the array powered on to remove the bad disk.

**NOTE:** For more information about physically removing a drive, see the hardware configuration guide that was shipped with your disk array. This chapter is also available in PDF format on the Discreet web site and the installation CD.

### STONE 1000 series (single controller) and 2000 Series (dual controllers):

1. Turn off the computer and the disk array.
2. Pull out the dead disk.
3. Move the spare disk to the dead disk's location.

### STONE 3000 series (3 controllers), 4000 series (4 controllers), D series (deskside FibreChannel) and R series (rackmount FibreChannel):

These disk arrays are "hot swappable". There is no need to turn off the computer or the disk array.

1. Pull out the dead disk.

**NOTE:** Since the drives are spinning when you pull them out, make sure to pull them out slowly.

2. Move the spare disk to the dead disk's location.

## Apply the New Configuration

1. Restart the computer
2. Log in as root:
3. Type:
 

```
cd /usr/discreet/sw
```

4. Type:
 

```
./sw_stop
```

```
./sw-unmount all
```
5. Open the Configuration utility by typing:
 

```
./config
```
6. Select Configuration from the Disk Array menu.
7. Make sure the spare has moved to the dead disk location and that there is no longer a disk at the spare location.
8. Click OK.

**NOTE:** Make sure that “Create FS” is set to “NO”. If it is set to “Yes” you will wipe your array.

9. Exit the configuration utility.
10. Type:
 

```
./sw_start
```
11. Start your software as usual and verify that the images on the disk array appear normal.

## Get a New Spare Drive

Once the heal procedure is finished, you need to get a replacement for your spare drive. If you have a valid hardware support contract, you can contact the Discreet Technical Assistance Centre to exchange your dead disk for a new spare. Before contacting Discreet, make sure the following information is available:

- Serial number of your SGI computer
- Model number of your dead disk (usually in the form of STXXXXXX for Seagate drives)
- Valid shipping address

## Install the New Spare Drive

When you get your new spare drive, you need to include it in your configuration.

1. Turn off your computer and your disk array.
2. Put your new spare drive in your disk array where your old spare drive used to be.
3. Turn on the disk array and the computer.
4. Log in as root.

5. In a UNIX shell, type:  
`cd /usr/discreet/sw`
6. Type:  
`./sw_stop`  
`./sw-unmount all`
7. Open the Configuration utility by typing:  
`./config`
8. Select Configuration from the Disk Array menu.
9. Make sure you can see your new spare drive. It should be white.
10. Click Apply.

**NOTE:** Make sure that “Create FS” is set to “NO”. If it is set to “Yes” you will wipe your array.

11. Type:  
`./sw_start`
12. Close the Configuration utility.

## Alternate Heal Procedure

Use this procedure if you do not have a spare drive in your configuration. If you do not know if you have a spare drive in the configuration, start with the first heal procedure (see “Using heal with a Spare Disk in the Framestore Configuration” on page 27).

**NOTE:** This procedure will also work if the bad disk prevents your system from booting and you need to remove the bad disk before starting the heal procedure.

You need a spare disk to run a heal. However, some configurations (Stone 1000, Single Stone 2000, Single D9 and some D9Ws) do not have spares configured in the actual disk array. If this is the case, you will need an external spare (a spare not already installed in your array). If you do not have a spare, contact Discreet Technical Assistance to know how to get one. Please note that you absolutely need a parity in the configuration. The heal utility cannot work without a parity drive.

1. Locate the SCSI addresses of the disks involved in the heal. For more information, see “Get Disk Information” on page 27.
2. Back up your volume header information. For more information, see “Back Up the Volume Header Information” on page 30.

3. Physically locate the bad disk. For more information, see “Physically Locate the Bad Disk” on page 32.

If your bad disk is preventing the system from booting, look at the disk LEDs. If a disk's LED is constantly ON (or OFF for Fibre Channel disks) despite the fact that the computer is turned off, the disk may have failed. If you have no physical evidence of a dead disk, then turn on the computer with the disk array turned off and look at `/var/adm/SYSLOG` for error messages that might help you locate a defective drive.

4. Physically remove the bad disk from the array. For more information, see “Remove the Bad Disk From the Array” on page 34.

**NOTE:** Do not turn off D9 disk arrays when removing the dead disk.

5. Put the spare where the bad disk was located.
6. Restart the computer.
7. Verify if the spare drive is suitable for a heal. For more information, see “Verify the Spare Disk” on page 29.

8. In a UNIX shell, type:

```
cd /usr/discreet/sw
```

9. Open the Configuration utility by typing:

```
./config
```

10. Select Configuration from the Disk Array menu.

Make sure that a white disk is located where the dead disk used to be. If the dead disk is still there and your white disk is somewhere else, you moved the wrong disks.

11. Run the heal utility. For more information, see “Run the Heal Utility” on page 30.

**NOTE:** We recommend using the serial number of the dead drive when performing the “Alternate Heal Procedure” with a D9 disk array.

Example:

```
./heal -s <serial number> -n <spare adapter>,<spare ID>
```

Alternately, if you are using the SCSI ID instead of the serial number, you should use the same address for the dead disk and the spare disk.

Example:

```
./heal -h 4,3 -n 4,3
```

12. Verify if the heal worked. For more information, see “Verify the Heal Operation” on page 31.

13. Start the software and verify that the images appear normal.
14. Get a new spare drive. For more information, see “Get a New Spare Drive” on page 35.
15. Install the new spare drive. For more information, see “Install the New Spare Drive” on page 35.

**NOTE:** Ignore this step if your disk array configuration does not have a spare drive.

## Standardizing Block Sizes Across Disks

As stated on page 30, before doing a heal, it is extremely important to verify that the number of blocks available on the spare disk is equal to or greater than the overall length of the framestore. If it is smaller, contact Discreet Customer Support to get a proper spare drive.

If the length of partition 7 on the spare disk is smaller but the total disk size is equal to or greater than the disks already in the partition, then follow the instructions in this section to standardize the block size across all disks.

1. Start the Configuration utility.
2. Make a note of the Length value (this value may be greyed).
3. Click the spare disk with the right mouse button to display the Disk pop-up menu.
4. Choose Partition Info. The Partition Window dialog box appears.

The maximum number of blocks you can format on your disk is indicated at the top of the list beside "the disk size is \_\_\_\_". If the number shown is smaller than the overall length, contact Discreet Customer Support, you may need a replacement drive.

If the disk size number is equal or bigger than the overall length, see the table (located below the ID "7" line). This length should be as big as the length of the framestore. If it is not, then you will have to resize the disk.

## Resizing a Disk

You need the following information to resize a disk:

- Controller ID number (ctrl#)
- Disk ID number (drive#)
- Total length of the framestore

### To resize a disk:

1. Start a UNIX shell.
2. Log in as super user.
3. Type:  
**fx -x**
4. Enter a device name at the prompt.
5. Provide the following information, when prompted:

ctrl#	Adapter ID number
drive#	Drive ID Number
lun#	Type "0"

6. Go to the Repartition/Resize menu, type:  
**r/re**
7. At the warning prompt, type:  
**y**
8. To choose block as the partitioning method, type:  
**block**
9. Enter the disk's new length value.
10. To exit the *fx* utility, type:  
**/exit**

You can go back to the configuration utility to make sure the repartition was successful.

**NOTE:** Only the standard partitions may be changed with this function. Type " ? " at the prompts, for a list of possible choices.

[illegible]



# Troubleshooting the Discreet Filesystem

*If it ain't broke...*

*This chapter explains how to identify and correct Discreet Filesystem problems. It also lists common troubleshooting tools that you can use to solve system problems.*

## Summary

In this chapter, you learn about:

- “Basic Troubleshooting Tools” on page 42
- “Problems and Solutions” on page 45
- “Discreet Filesystem Utilities” on page 50

## About Troubleshooting the Discreet Filesystem

Before you begin troubleshooting, it may be useful to answer the following questions:

- What version of the Discreet Filesystem software is running?
- What application software is running?
- Has any software been recently installed?
- Have there been any recent hardware or configuration changes?
- What is the exact error message or nature of the failure?
- What information is in the Console window or error logs?
- When did the problem start happening?

- Is the problem repeatable or persistent? If so, is the problem consistent (does it happen at the same place each time)?
- What type, size, and number of disk arrays are on the system?
- What was happening at the time of the error, or what are the steps leading to the problem?

This information will also be useful in the event that you contact Discreet Customer Support.

## Basic Troubleshooting Tools

The basic troubleshooting tools for the **stone** disk array, Discreet Filesystem, and other system components are summarized in the following table:

Area of Use	Name of Tool	Function
Configuration	Configuration utility	A utility for configuring your disk array. For more information, see "Create and Configure Volume(s)" on page 21.
	<i>swr-set-config</i>	A utility that sets the system's configuration file. For more information, see "swr_set_config" on page 66.
	<i>sw-mount</i>	Enables a Discreet Filesystem. To start any Discreet application, you must have a mounted filesystem. For more information, see "sw-mount" on page 51.
Error Reporting	<i>swr-set-debug</i>	A utility that provides error reporting for the custom i/o drivers (swr). For more information, see "swr-set-debug" on page 65.
	<i>swf-tweak</i>	A utility that can provide error reporting for the Discreet Filesystem. For more information, see "swf-tweak" on page 62.

Area of Use	Name of Tool	Function
Hardware	<i>disk-summary</i>	A utility that displays disk information. For more information, see “disk-summary” on page 53.
	<i>fx</i>	An IRIX utility used to display and modify disk partition sizes and drive parameters. For more information, see “Configuration Utility Error: “Partition 7 does not exist”” on page 46.
	<i>hinv</i>	An IRIX command that displays a machine’s hardware inventory.
	<i>stone-test</i>	Measures the performance of the Discreet Filesystem. For more information, see “stone-test” on page 55.
	<i>sw_print_config</i>	Can be used to obtain the number of errors recovered or unrecovered by the driver. For more information see “swr_print_config” on page 61.
Disk Failure	<i>heal</i>	A utility that rebuilds data lost from the failure of a single disk. For more information, see Chapter 3, “Healing a Bad Disk in a stone Disk Array”.

Area of Use	Name of Tool	Function
Software	<i>swf-df</i>	Displays the total, free, and used space of all mounted filesystems. For more information, see “swf-df” on page 53.
	<i>disk_serials</i>	Can be used to test all connected drives for a valid licence. For more information, see “disk_serials” on page 55.
	<i>swf-frag</i>	Evaluates and reports fragmentation of the Discreet Filesystem. For more information, see “swf-frag” on page 57.
	<i>sw-fsck</i>	Audits the Discreet Filesystem and reports any inconsistencies or errors. For more information, see “sw-fsck” on page 59.
	<i>swr_print_config</i>	Displays the active configuration of a <b>stone</b> disk array. For more information, see “swr_print_config” on page 61.
	<i>versions</i>	An IRIX command that provides the version of software installed on your system.

For a comprehensive list of Discreet Filesystem utilities, including descriptions, syntax and examples, see “Discreet Filesystem Utilities” on page 50.

## Enable Verbose Error Reporting

You can use the *swr-set-debug* utility to report I/O errors if you experiences problems.

### To run *swr-set-debug*:

1. Run *swr-set-debug* by typing:

```
/usr/discreet/sw/tools/swr-set-debug 1
```

The number 1 in the argument lists every entry the system encounters. You can also set this to 2 to receive additional information about each error.

2. Check the Console window and the */var/adm/SYSLOG* file for error information.
3. Once your system is operating normally, return *swr-set-debug* to the default reporting level by typing:

```
/usr/discreet/sw/tools/swr-set-debug 0
```

## Problems and Solutions

### Disk/Controller Is Not Available in the Configuration Utility

Once the **stone** array is connected to the host computer and the system is rebooted, the hard disks should be enabled and visible in the configuration utility. To verify the status of your hard disks, carry out the following procedure:

1. Log in to the system.
2. Use the `hinv` command to view all controllers and hard disk drives connected to your system. Type:

```
hinv -c disk
```

A listing of all controllers and hard disks attached to your system appears. For example, the listing for two **stone** D9 arrays running on an SGI Octane should be similar to the following:

```
Integral SCSI controller 0: Version QL1040B (rev. 2), single
ended
```

```
    Disk drive: unit 1 on SCSI controller 0
```

```
    Disk drive / removable media: unit 2 on SCSI controller 0:
    720K/1.44M floppy
```

```
Integral SCSI controller 1: Version QL1040B (rev. 2), single
ended
```

```
Integral SCSI controller 11: Version QL1040B (rev. 2), single
ended
```

```
    Disk drive: unit 1 on SCSI controller 11
```

```
Integral SCSI controller 2: Version Fibre Channel AIC-1160,
revision 2
```

```
    Disk drive: unit 0 on SCSI controller 2
```

```
    Disk drive: unit 1 on SCSI controller 2
```

```
    Disk drive: unit 2 on SCSI controller 2
```

```
    Disk drive: unit 3 on SCSI controller 2
```

```
    Disk drive: unit 4 on SCSI controller 2
```

```
    Disk drive: unit 5 on SCSI controller 2
```

```
    Disk drive: unit 6 on SCSI controller 2
```

```
    Disk drive: unit 7 on SCSI controller 2
```

```
Integral SCSI controller 3: Version Fibre Channel AIC-1160,
revision 2
```

```
    Disk drive: unit 0 on SCSI controller 3
```

```
    Disk drive: unit 1 on SCSI controller 3
```

```

Disk drive: unit 2 on SCSI controller 3
Disk drive: unit 3 on SCSI controller 3
Disk drive: unit 4 on SCSI controller 3
Disk drive: unit 5 on SCSI controller 3
Disk drive: unit 6 on SCSI controller 3
Disk drive: unit 7 on SCSI controller 3

```

If all SCSI controllers and hard disks for a given configuration are not visible, consult your *stone* Disk Array Hardware Configuration Guide.

### Configuration Utility Error: "Partition 7 does not exist"

This error indicates that the disk does not contain the correct drive-level partition information. This means that the disk has not been partitioned in UNIX. If this is the case, format the disk using the following procedure:

1. Log in to the system as root.
2. Type:

```
fx -x -c "dksk(x,y)" INITIALIZE
```

Where x is the controller and y is the hard disk drive.



**WARNING:** All contents of the hard disk are lost when it is initialized.

### Configuration Utility Error: "Offset/Length out of range (X-+XXXXXXXX)"

If the disk has been partitioned in UNIX, this error message could indicate that the number of blocks for this disk is smaller than the number of blocks of other disks in the volume (occasionally, some disks may vary in size, depending on the date they were manufactured). You can work around this problem by selecting the smallest disk first, and then selecting the remaining disks in the configuration utility.

Alternately, the drive-level partitions may be sized incorrectly. See "Standardizing Block Sizes Across Disks" on page 38 for more information.

## stone Disk Array Was Accidentally Shut Down

When a disk array is powered down accidentally while the main Discreet application is running, the system automatically marks all disks as Dead (disk are not writable nor readable). To correct this problem, mark all disks as Well using the following procedure:

1. Exit the Discreet application.
2. Start the Configuration utility by typing:  
`/usr/discreet/sw/config`
3. Right-click on a Dead disk.  
The Disk pop-up menu appears.
4. Choose Set Well.
5. Repeat for all Dead disks.
6. Click Cancel.



**WARNING:** Clicking Apply or OK while the Create FS option is set to Yes will destroy all data in the current volume.

## Image Appears Striped Although No Drives Are Marked as Dead

If the images appear striped (stripe of noise is visible on all frames in the framestore), it is likely that a disk has failed but that the system was not able to detect the problem.

### To locate a defective disk manually:

1. Exit the Discreet application.
2. Start the Configuration utility by typing:  
`/usr/discreet/sw/config`
3. Select the appropriate volume.
4. Right-click on a disk (of the same colour as the current volume).  
The Disk pop-up menu appears.
5. Choose Set Dead.
6. Click Cancel.



**WARNING:** Clicking Apply or OK while the Create FS option is set to Yes will destroy all data in the current volume.

7. Start the Discreet application.

If the Dead disk is truly defective, the parity drive will be enabled and the image will appear as it should (no stripe in the image). If the image does not appear normal (i.e. a second stripe appears), return to the Configuration utility, mark the Dead disk as Well using the Set Well command, and mark the next disk as Dead. Repeat this procedure until the defective disk is identified.

8. Heal the Dead disk using the procedure described in Chapter 3, “Healing a Bad Disk in a stone Disk Array”.

## Cannot Access a Discreet Filesystem Volume

If the Discreet application cannot access a volume, consider the following.

### Are the Volume and Drives Correctly Configured?

Make sure you do not have multiple dead disks or missing drives. The shell where you launch the configuration utility will report errors about missing drives (see “Configuring the Disk Array(s)” on page 20).

### Is the Volume Mounted?

Normally, volumes are mounted automatically by the *sw\_start* script. To check if a Discreet Filesystem volume is mounted, use the *swf-df* command (see “swf-df” on page 53). This command displays the total, free, and used space of all mounted filesystems.

If no volumes are mounted, use *sw-mount all* (see “sw-mount” on page 51). This command mounts all Discreet Filesystem volumes.

### Is the Filesystem Full?

You cannot write to Discreet Filesystem volumes that are completely full. Use the *swf-df* command to check disk usage (see “swf-df” on page 53). This command also reveals the presence of purgeable space. Purgeable space is not accessible by the framestore and consists mainly of recently deleted files or temporary files that should have been deleted automatically. Use the *sw-purge* command to free disk space and reduce the amount of purgeable space to zero (see “swf-purge” on page 50). Once in the Discreet application, you should delete or archive older material from the clip libraries to free disk space.

**NOTE:** The VTR archiving module in Discreet applications requires free space on the framestore to generate slates, test patterns, headers, and audio encoding.



## VOLUMEMGT: WARNING: The disk array has 1 bad disk(s)

This error message in the status bar of the main Discreet application is an indication of a bad disk in your framestore:

```
VOLUMEMGT: WARNING: The disk array has 1 bad disk(s)
```

If you encounter a bad disk in your framestore, you need to use the heal utility to recover the data from the failed disk. To ensure proper recovery of your data, refer to Chapter 3, “Healing a Bad Disk in a stone Disk Array”.

## stone Disk Is Marked as UL in the Configuration Utility

**NOTE:** This procedure is intended for the SGI Onyx 1 and Indigo2 platforms running pre-**stone** Discreet arrays only.

UL indicates that a disk is unlicensed for use with the Discreet Filesystem.

### To obtain a licence:

1. In a shell, type:

```
cd /usr/discreet/sw
./disk_summary
```

This lists the drives on the **stone** array and reveals information that is required by Discreet to issue a licence.

2. Contact Customer Support with this information (see “Discreet Customer Support” on page 5).

### To install a licence:

1. Log in with root privileges.

2. Type:

```
cd /etc
jot swr-license
```

3. Enter the licence number (case sensitive) and save the file. If you received your licence number by e-mail, it may be easier to copy the string and paste it in the text editor.
4. At the prompt, type:

```
cd /usr/discreet/sw/tools
```

5. Activate the licence by typing:

```
./swr-set-license
```

## Discreet Filesystem Utilities

### swf-purge

*swf-purge* releases frames that are ready for reuse back to the free frame pool. These frames will no longer be recoverable.

#### Synopsis

```
swf-purge [-partition <partition>] [-retired <timeout>] [-
unused <timeout>]
```

Part:	Description:
-partition <partition>	Source hard partition (volume). The default is 0.
-retired <timeout>	Defines a newer retired frames timeout value. See “swf-tweak” on page 62.
-unused <timeout>	Defines a newer unused frame timeout value. See “swf-tweak” on page 62.

#### Examples

The following example purges all purgeable frames on volume 0:

```
./swf-purge -r 0 -u 0
```

#### Location

/usr/discreet/sw

#### Related Utility

swf-tweak

## sw-mount

*sw-mount* enables a Discreet Filesystem. Unmounted filesystems are not visible to Discreet applications but cannot be accessed. Filesystems are usually mounted with the *sw\_start* script.

### Synopsis

```
sw-mount [-r <retries>] <partition>..<partition>
```

Part:	Description:
-r <retries>	Defines the number of allowable CheckSum errors. By default, the filesystem will fail to mount if there is a checksum error. Using -r, the sw-mount utility will discard all "corrupted" (bad checksum) frames, up to the value specified after -r. The default value is 1. You should archive your data and recreate the filesystem before using this option.
<partition>	ID number of the partitions you wish to mount (separated by a space). Alternatively, you can specify "all" to mount all known partitions, although the response is less verbose than listing each partition explicitly.

### Examples

To mount all partitions, use the following command:

```
sw-mount all
```

To mount partitions 0 and 2, use the following command:

```
sw-mount 0 2
```

To mount partition 0, and discard up to 2000 checksum errors, use the following command:

```
sw-mount -r 2000 0
```

### Location

/usr/discreet/sw

### Related Utilities

sw-unmount, sw\_start, swf-df

sw-unmount

sw-unmount disables a Discreet Filesystem. Typically, you unmount a filesystem to accomplish tasks such as modifying the disk configuration.

Synopsis

sw-unmount <partition>

Part:	Description:
<partition>	<p>Defines the partition numbers that you wish to unmount (separated by a space). This is a required parameter; there is no default.</p> <p>You can specify "all" to unmount all mounted partitions, although specifying specific partitions will be more verbose.</p> <p>You should verify that all products using Discreet Filesystem are stopped, including <b>wire</b> and mountstone. For more information, see "sw_stop" on page 110.</p>

Examples

To unmount all your partitions, execute the following command:

sw-unmount all

To unmount partitions 0 and 2, execute the following command:

sw-unmount 0 2

Location

/usr/discreet/sw

Related Utilities

sw-mount, sw\_start, sw\_stop, swf-df

## disk-summary

When run by the root account, `disk-summary` lists information about all disk drives on the system. This tool is used to display more extensive information about the drives installed on a machine and is very useful to ensure that all drives have similar configurations and are labelled “STON+WIR”.

*disk-summary* provides the following information:

- Adapter Number
- Target Number
- Disk Label
- Disk Model Number
- Revision Number
- Serial Number

### Synopsis

`disk-summary`

### Location

`/usr/discreet/sw`

## swf-df

*swf-df* displays the total, free, and used space of all mounted filesystems. It also displays information according to a resolution specified on the command line. This is often used to verify that filesystems have been mounted or unmounted using *sw-mount* and *sw-unmount*.

### Synopsis

`swf-df [<frame-size> | NTSC | PAL | FILM | NTSC+ALPHA | PROXYNTSC]`

Part:	Description:
[<frame-size>   NTSC   PAL   FILM   NTSC+ALPHA   PROXYNTSC]	Selects the framesize for output. You can specify frame-size by specifying the number of bytes per frame. The default resolution is in blocks of 512 bytes.

### Examples

To find current space usage in NTSC frames, use the following command:

```
swf-df NTSC
```

You should obtain output similar to the following:

```
Partition free  %use  purgeable%use  total   ifree  itotal  %Iuse
0          22214 82    0          0      129774 14374  260436 0
```

Where:	Is:
Partition	Discreet Filesystem volume
free	Available frames
use%	Ratio of frames in use to total volume capacity
purgeable	Purgeable frames
%use	Ratio of purgeable frames to total volume capacity
total	Total volume capacity
ifree	Number of free Inodes
itotal	Total number of Inodes
%luse	Ratio of Inodes in use to total Inodes

To display the report in 1920 x 1080 x 3 frames (HDTV), use the following command:

```
swf-df 6220800
```

### Location

```
/usr/discreet/sw
```

### Related Utilities

```
swf-frag, sw_mount, sw_unmount
```

## disk\_serials

*disk\_serials* is only used when installing the Discreet Filesystem and **wire** software.

### Synopsis

```
disk_serials [-v] [-t] [-r]
```

Part:	Description:
-r	Displays a list of all devices that require a Stone+Wire licence in order to be used with the Discreet Filesystem.
-t	Tests all connected drives for a valid Stone+Wire licence. Outputs the number of unlicensed drives.
-v	Depending on various options, this turns on verbose mode. When the application is in verbose mode, it lists all unlicensed serial numbers in a column.

### Location

/usr/discreet/sw/tools

## stone-test

Measures the performance of the Discreet Filesystem.

### Synopsis

```
stone-test [-a <processes>] [-B <bandwidth>] [-b <bandwidth>]
[-C] [-c] [-d <device>] [-f NTSC|PAL|HDTV|FILM|NTSC+ALPHA|
PROXYNTSC] [-i <interval>] [-n <frames>] [-P <method>] [-p
<processes>] [-R <testcase>]
[-r] [-t] [-v] [-w] [-z]
```

Part:	Description:
-a <processes>	When a specified number of processes have completed, another specified group of processes will be executed (see -p).
-B <bandwidth>	Defines Bandwidth Reservation in frames per second (Frames/S). Default is no Bandwidth reservation.
-b <bandwidth>	Defines Bandwidth Reservation in megabytes per second (MB/S). Default is no Bandwidth reservation.
-C	Displays output in columns, without text.

Part:	Description:
-c	Displays CPU usage during the test.
-d <device>	Specifies the device name. The default value is /dev/swr00. You can specify several devices.
-f (NTSC   PAL   HDTV   FILM   NTSC+ALPHA   PROXYNTSC)	Sets the frame size to a predefined value based on resolution. The default is NTSC. Options include: NTSC (720 x 486 x 3), PAL (720 x 576 x 3), HDTV (1920 x 1080 x 3), Film (2048 x 1556 x 3), NTSC+Alpha (720 x 486 x 4), ProxyNTSC (NTSC resolution divided by 4).
-i <interval>	Sets the frame size to the number of intervals multiplied by the number of drives in the array. The default values are frames of NTSC size (see the -f option).
-n <frames>	The number of frames to be processed. Default is 300.
-P <method>	Defines the method used for selection of the next partition. 0 will define a random selection, whereas 1 will define sequential selection. This is selected for each operation handled.
-p <processes>	This defines the number of simultaneous read and write requests to be executed at any given time. The default is 16.
-R <testcase>	0: Head, forward 1: Tail, forward 2: Head, backward 3: Tail, backward 4: Thrash forward 5: Thrash backward 6: Random 7: Skip 2, forward 8: Skip 2, back 9: Skip 3, forward 10: Skip 3, back The default is "all".
-r	Only performs the "read" test.
-t	Application waits for data instead of polling.



Part:	Description:
-v	Enables verbose mode.
-w	Only performs the “write” test. This is destructive.
-z	Displays additional information regarding the time it takes to execute each read/write request.

### Examples

For a general, non-destructive test of your Discreet Filesystem, use the following command:

```
stone-test -r
```

For a more in depth, non-destructive test, perform a Random Read of 500 PAL frames with the following command:

```
stone-test -r -R 6 -f PAL -n 500
```

### Location

```
/usr/discreet/sw/tools
```

### swf-frag

This tool assesses the fragmentation of the Discreet Filesystem.

### Synopsis

```
swf-frag [-v] [<framesize> | MEG | NTSC | PAL | FILM |  
NTSC+ALPHA | PROXYNTSC]
```

Part:	Description:
-v	Verbose mode.
[<frame-size>   MEG   NTSC   PAL   FILM   NTSC+ALPHA   PROXYNTSC]	Specifies the frame size used to express output. You can also use bytes.

### Examples

To display the report in NTSC frames, use the following command:

```
swf-frag PAL
```

To display the report in 1920 x 1080 x 3 frames (HDTV), use the following command:

```
swf-frag 6220800
```

### Location

/usr/dscreet/sw/tools

### Related Utility

swf-df

## heal

Rebuilds a failed device. The Discreet Filesystem heal utility inserts a spare replacement disk into the configuration, and then proceeds to "heal" the replacement device. You must be super-user to run heal. Never stop heal once it is running. Refer to Chapter 3, “Healing a Bad Disk in a stone Disk Array” for more information.

### Synopsis

```
heal [-h <adapter>,<target> | -s <serial>] -n
<adapter>,<target>
```

Part:	Description:
-h <adapter>,<target>	Specifies the adapter and target information for the failed disk.
-n <adapter>,<target>	Specifies the adapter and target information for the disk that replaces the failed disk.
-s <serial>	Specifies the serial number of the failed disk.

### Examples

To heal the SCSI disk on Adapter 2, Target 3 and replace it with your spare disk on Adapter 5, Target 4:

```
heal -h 2,3 -n 5,4
```

To heal the disk labeled Serial Number NJ043377 with your spare disk on Adapter 5, Target 4, use the following command:

```
heal -s NJ043377 -n 5,4
```

### Location

/usr/dscreet/sw/tools

### Related Utilities

swr\_restore\_vh

**sw-fsck**

*sw-fsck* audits a Discreet Filesystem and reports any inconsistencies or errors similar to the UNIX *fsck* utility. *sw-fsck* automatically tests all partitions but will not repair the filesystem.

**Synopsis**

```
sw-fsck
```

**Location**

```
/usr/discreet/sw/tools
```

**Related Utilities**

```
swf-backupfs
```

**swf-backupfs**

*swf-backupfs* is used to back up the Discreet Filesystem to a file for analysis (using *swf-analys*) or for restoration (using *swf-restorefs*). Before superblocks are backed up, data integrity is checked in a way that is similar to *sw-fsck*.

**NOTE:** This only back ups the superblock information; it does not back up the actual frames.

**Synopsis**

```
swf-backupfs [-w <Superblock>] <filename>
```

Part:	Description:
-w <Superblock>	Specifies the Superblock number (0, 1, or 2). Each partition has 3 Superblocks. By default, the "most current" Superblock is used.
<filename>	Specifies the destination file. The partition number is appended to the filename (i.e.: Test will become Test-0 for partition 0).

### Examples

To back up the most current Superblock to the file *MyBackup* (which will become *MyBackup-0* for partition 0, *MyBackup-1* for partition 1, etc), use the following command:

```
swf-backupfs MyBackup
```

To back up Superblock 1 to the file *MyBackupSB1*, use the following command:

```
swf-backupfs -w 1 MyBackupSB1
```

### Location

```
/usr/discreet/sw/tools
```

### Related Utilities

```
swf-analfs, swf-fsck, swf-restorefs
```

### swf-restorefs

*swf-restorefs* restores the Discreet Filesystem superblock information to the same partition that it was backed up to using *swf-backupfs*. This only restores the superblock information; it does not restore actual frames.

**NOTE:** You should stop the Discreet Filesystem and **wire** and unmount your framestore before using this command (see “*sw\_stop*” on page 110 and “*sw-unmount*” on page 52.)

### Synopsis

```
swf-restorefs <filename>
```

Part:	Description:
<filename>	Specifies the filename of the input source. Whereas <i>swf-backupfs</i> takes the base filename, <i>swf-restorefs</i> takes the true filename. Therefore, you must specify the filename including the partition (i.e.: If you specified the filename <i>Test</i> in <i>swf-backup</i> , you will need to enter <i>Test-0</i> to restore partition 0).

### Example

To restore the backup made for Partition 1 from the file *MyBackup-1*, use the following command:

```
swf-restorefs MyBackup-1
```

**Location**

/usr/discreet/sw/tools

**Related Utilities**

sw\_stop, sw-unmount, swf-analys, swf-backupfs

**swr\_print\_config**

*swr\_print\_config* gets the active striping order for every partition configured within the **stone** array and displays the active configuration. It provides the following information for each disk:

Entry:	Description:
serial	Serial number
a	Adapter number
t	Target number
l	Lun number
status	Bitwise status where: <ul style="list-style-type: none"> <li>• 7 = good disk</li> <li>• 54= missing disk</li> <li>• 55= dead disk</li> </ul>
recovered	The number of errors from which the driver recovered. This counter resets when you reboot the system.
unrecovered	The number of errors from which the driver could not recover. This counter resets when you reboot the system.

**Synopsis**

swr\_print\_config

**Location**

/usr/discreet/sw/tools

**Related Utilities**

print\_config, swr\_get\_config, swr\_set\_config

## swr\_get\_config

*swr\_get\_config* is used to acquire the *swr2cfg* file from the specified disk. Without this utility, you would have to use *dvhtool*.

### Synopsis

```
swr_get_config <adapter> <target> <filename>
```

Part:	Description:
<adapter>	Specifies the adapter of the disk you wish to update.
<target>	Specifies the target of the disk that you wish to update.
<filename>	Specifies the name of the <i>swr2cfg</i> file.

### Examples

To get the configuration from the disk on Adapter 2, Target 4 and save it as *swrConfig*, you would use the following command:

```
swr_get_config 2 4 swrConfig
```

### Location

```
/usr/discreet/sw/tools
```

### Related Utilities

```
swr_print_config, swr_set_config
```

## swf-tweak

*swf-tweak* configures your local framestore settings from the command line.

**NOTE:** Your configuration should be tuned for optimal performance. We do not recommend changing any values using *swf-tweak*.

### Synopsis

```
swf-tweak [-i]
```

Part:	Description:
-i	Starts the interactive mode, which allows you to change settings using menu-based selections.

## Options

Option:	Description:
Debug Flag	Specifies the level of debug output tracing. The higher the number the more verbose the information becomes. The output is redirected to the console shell and the system log file (/var/adm/SYSLOG).
Sync Threshold	This setting specifies the number of filesystem operations completed before the filesystem synchronizes. A number that is too large will not allow the filesystem to be updated often enough to be consistent and a number too small may impact performance negatively.
Sync Period	If the Sync Threshold is not exceeded but the number of seconds from the last filesystem synchronization exceeds the Sync Period, the filesystem is automatically synchronized.
Sync Lock Time	The time interval in seconds between filesystem synchronizations.
Max Sync Lock Time	If the last time the filesystem synchronized exceeds the Maximum Sync Lock Time, a filesystem synchronization will be forced after the given number of seconds.
Retired Frame Expiration Time	Represents the number of seconds before Retired frames (frames that are marked retired, purgeable, etc.) are purged from the filesystem.
Unused Frame Expiration Time	Represents the number of seconds before Unused frames (frames that have been allocated but never used) are purged from the filesystem.

## Location

/usr/discreet/sw/tools

## Related Utility

swr-set-debug

**swr\_restore\_vh**

When the *heal* command is interrupted, the original system configuration should be restored automatically. However, there are cases where the *heal* utility will not recover from an interruption, and you will need to restore the original configuration manually. *swr\_restore\_vh* reads the configuration file stored on disk and puts it back in memory. Normally, this operation is done automatically at startup.

**Synopsis**

`swr_restore_vh`

**Location**

`/usr/discreet/sw/tools`

**Related Utilities**

`heal`, `mark_disk`

**mark\_disk**

*mark\_disk* is used to set drive status from the command line. Alternately, you can set drive status using the configuration tool.

**Synopsis**

`mark_disk -n <Adapter>,<Target> -s [DEAD|SICK|WELL]`

Part:	Description:
<code>-n &lt;Adapter&gt;,&lt;Target&gt;</code>	Defines which disk is updated to the new status.
<code>-s [DEAD SICK WELL]</code>	Sets the new state of the target.

**Example**

To mark the disk on Adapter 3, Target 4 “dead”, use the following command:

`mark_disk -n 3,4 -s DEAD`

**Location**

`/usr/discreet/sw/tools`

**Related Utility**

`config`



## swf-analfs

*swf-analfs* displays the number of fields in a superblock for each frame descriptor (in tabular form).

### Synopsis

```
swf-analfs <backup_filename> <filename.txt>
```

Part:	Description:
<backup_filename>	Specifies the source file. Whereas <i>swf-backupfs</i> takes the base filename, <i>swf-analfs</i> takes the true filename. Therefore, you must specify the filename including the partition (i.e. if you specified the filename Test in <i>swf-backup</i> , you will need to enter Test-0 to restore partition 0).
<filename.txt>	Redirects output to a text file.

### Example

The following command writes the result of **swf-analfs** to a text file:

```
swf-analfs Test-0 > analysis.txt
```

### Location

```
/usr/discreet/sw/tools
```

### Related Utilities

```
swf-backupfs, swf-restorefs
```

## swr-set-debug

*swr-set-debug* allows you to set the debug level of the Discreet Filesystem kernel driver according to your needs.

### Synopsis

```
swr-set-debug <level>
```

Part:	Description:
<level>	Defines the level: 0 for minimal error messages, 1 for all errors, 2 for debugging information, and 3 for everything.

**Location**

/usr/dscreet/sw/tools

**Related Utilities**

swf-tweak

**swr\_set\_config**

*swr\_set\_config* writes the *swr2cfg* file to the specified disk. Without this tool, you would have to use *dvhtool*. This command can be destructive.

**Synopsis**

*swr\_set\_config* <adapter> <target> <filename>

Part:	Description:
<adapter>	Specifies the disk's adapter.
<target>	Specifies the target disk.
<filename>	Specifies the file that will replace <i>swr2cfg</i> on the specified disk. This file must exist in the current directory.

**Example**

To configure the disk on Adapter 2, Target 4 based on *swrConfig*, use the following command:

```
swr_set_config 2 4 swrConfig
```

**Location**

/usr/dscreet/sw/tools

**Related Utilities**

*swr\_print\_config*, *swr\_get\_config*

**print\_config**

*print\_config* displays a list of disks within the framestore, and information about the partition. The output is in text format.

**NOTE:** Unlike the *swr\_print\_config* tool, disks are not shown as per the striping order. *print\_config* will not show a disk if it was not part of the inventory at boot time even though it is still part of the configuration.

**Synopsis**

```
print_config
```

**Location**

```
/usr/discreet/sw/tools
```

**Related Utilities**

```
swr_print_config
```

**swr-set-license**

*swr-set-license* installs a licence onto selected drives.

**Synopsis**

```
swr-set-license [<filename>]
```

Part:	Description:
<filename>	The filename of the licence. Defaults to <i>/etc/swr-license</i> .

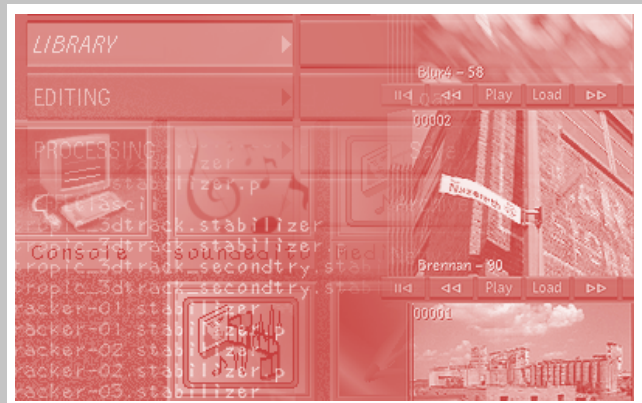
**Location**

```
/usr/discreet/sw/tools
```

**Related Utility**

```
disk_serials
```

[illegible]



## Section 2: Networking

*Learn how to*

*set up wire,*

*configure a*

*HIPPI network,*

*and use*

*mountstone.*



# 5

## Setting Up wire

*Hooking up*

*Improve workflow by sharing your clips with other Discreet workstations  
using optional **wire** networking.*

### Summary

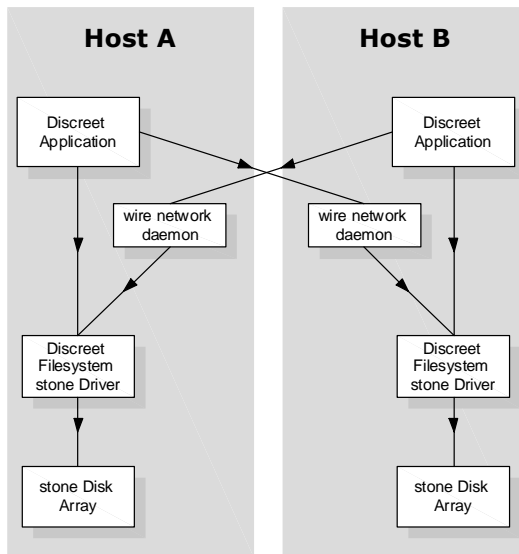
This chapter describes the features of the optional **wire** networking solution along with detailed configuration procedures.

In this chapter, you learn about:

- “About wire” on page 72
- “wire Licencing” on page 73
- “Setting Up wire” on page 74

## About wire

Use Discreet's **wire** networking to access Discreet Filesystem volumes on **stone** disk arrays located on other host machines. **wire** supports both the TCP/IP and HIPPI-FP network interfaces.



**wire** allows an application to transfer material to and from any clip library, in any software partition, on any host on the **wire** network. This is provided by a network daemon running on each local host. The use of a local daemon means that network access is independent of the main Discreet application. Applications running on other hosts can read and write material on the local framestore by communicating with the **wire** network daemon.

## Supported Network Protocols

The **wire** interface supports the following network protocols:

- TCP/IP over Ethernet (10BaseT, 100BaseT), Gigabit Ethernet, HIPPI, FDDI, Fibre Channel or ATM (OC3, OC12).
- HIPPI-FP (Frame Protocol or character mode) over HIPPI

The fastest interface is HIPPI. It uses a transfer protocol called HIPPI-FP to transfer images and sound between two hosts equipped with a HIPPI interface (either SGI or Essential). HIPPI-FP is a low-level protocol that bypasses the TCP/IP protocol layer and therefore provides a superior transfer rate when compared to TCP/IP Ethernet.



You can also use the TCP/IP protocol through the HIPPI interface. This lets you enjoy the features available only through the TCP/IP protocol (such as ftp and nfs) with the added speed provided by the HIPPI interface.

## Requirements

**wire** requires the following:

- **stone** disk array
- Discreet Filesystem
- Valid **wire** network daemon licence (server)

## Main Features

- High-speed throughput
- Flexible network access
- Compatible with all Discreet applications

## File Sharing

Since the clip library pointers reside on the host's system disk, file locking is implemented by NFS. This stops two persons from simultaneously writing to the same clip library.

## wire Licencing

You need a valid **wire** network daemon licence to run the **wire** network server (**wire** clients do not require a licence).

This licence is separate from your main application licence. To obtain a licence, submit the SGI System ID number for each computer on which you will install **wire** to Discreet Licencing. To get your System ID number, type **lmhostid** in a UNIX shell. This returns the following:

```
lmhostid - Copyright (C) 1989-1998 Globetrotter Software,
Inc.The FLEXlm host ID of this machine is "690e0f8a"
```

Where 690e0f8a is the System ID.

**To install the wire licence:**

1. Log in with root privileges.
2. Type:
 

```
cd /usr/local/flexlm/licenses
jot DL_license.dat
```
3. Enter the licence number (case sensitive) and save the file.

**NOTE:** After you install the licence you will need to reboot the machine.

Once the licence is installed, the `/usr/local/flexlm/licenses/DL_license.dat` file should include a statement similar to the following:

```
FEATURE wirel discreet_l 1.999 1-jan-0 200 5C7237F5AD005CEA8F82
ck=21
```

## Setting Up wire

Set up **wire** only after setting up the Discreet Filesystem on your local machine. You must perform the following procedures to configure **wire**:

- Verify that autofs is enabled.
- Edit the host configuration file (`sw_host_map`) on every machine on the **wire** network
- Edit the `/etc/exports` file on remote machines that you want to access.

For additional information, refer to Chapter 4, “Troubleshooting wire”.

- Make sure all the hosts you want to access are specified in the `/etc/hosts` file. A typical entry in the `/etc/hosts` file look like:

```
172.16.100.176 samoa
```

**NOTE:** NFS must be installed on your machines to run the **wire** network.

### Verify autofs Is Enabled

You can perform this procedure at any point when setting up the **wire** network.

1. In a UNIX shell, type:
 

```
chkconfig
```
2. Look for the `autofs` entry, and verify that it is set to on.

- 3. If this entry is set to off, type:

```
chkconfig autofs on
```

Verify NFS Is Enabled

- 1. In a UNIX shell, type

```
chkconfig
```

- 2. Look for the nfs entry, and verify that it is set to on.

- 3. If this entry is set to off, type

```
chkconfig nfs on
```

Edit the Host Configuration File

Edit this file on each host machine you want to include in the **wire** network.

To edit the host configuration file:

- 1. Make sure you are logged in as root.
- 2. Obtain the host name for each machine on the **wire** network. Type:

```
hostname
```

- 3. Obtain the IP address of each machine on the **wire** network. Type:

```
ping -c 1 <hostname>
```

- 4. Go to the directory by typing:

```
cd /usr/discreet/sw/cfg
```

- 5. Type:

```
jot sw_host_map
```

- 6. In the [HOSTS] section, include a line for each host on the **wire** network. Use the following syntax:

```
HOST=<hostname> HADDR=<hostIPAddress> ID=<uniqueID>
```

Where:	Is:
<hostname>	The name of the host machine. You can verify the host name by typing <b>hostname</b> in a UNIX shell.

Where:	Is:
<hostIPAddress>	The IP address of the host machine. You can find this information in the <i>/etc/hosts</i> file. Look for the line that matches the host name or check with your systems administrator.
<uniqueID>	The Discreet Filesystem unique identifier. This ID is stamped on all disks when you configure your framestore. If you change this ID at a later date, you must recreate the Discreet Filesystem (erasing the contents of the array).

To edit a specific file, go to each host machine you want to include in the **wire** network, and open the configuration (*sw\_host\_map*) file. Copy and paste the [HOSTS] default line of that specific machine into the file you are editing. The [HOSTS] section should be identical on each of the hosts.

7. In the [INTERFACES] section, type a line for each host on the **wire** network. When typing a line for the local host, include all the interface protocols supported by the local host. When typing a line for a remote host, include only one interface protocol. Use the following syntax:

```
HOST=<hostname>
PROT=<protocol> IADDR=<interface address> DEV=<device#>
```

Where:	Is:
<hostname>	The name of the host machine. Make sure that this is the primary name of the host machine.
<protocol>	Either HIPPI or TCP. Include all interface protocols supported by the local host. Include only one interface protocol for each remote host.
<interface address>	The IP or HIPPI address of the host machine. If you have selected the TCP/IP protocol with the PROT=TCP token, the interface address is the IP address for the host. You can find the IP address in the network configuration file <i>/etc/hosts</i> .  If you have selected the HIPPI protocol with the PROT=HIPPI token, the interface address is the I-Field address for the HIPPI interface (for example, 0x03000001). Please refer to your HIPPI network configuration documents for more information on how to determine the I-Field value.
<device#>	Set the DEV token to 1.

8. When you have finished typing all your information, save the file and exit.

- Restart the server by typing:

```
/usr/discreet/sw/sw_restart
```

**NOTE:** If you are running mountstone, you must restart the server manually. Type:

```
/usr/discreet/MountStone/bin/dlmountstone start
```

## Edit the /etc/exports File

Edit the Exports file (*/etc/exports*) on the remote machines you want to access.

**To edit the exports file:**

- Log in as root to the remote host you want to access.
- Type:

```
jot /etc/exports
```

- Add a line at the end of the Exports file using the syntax presented below. The entry must be a single line:

```
/usr/discreet/clip -rw,access=<host1>:<host2>:<host3>,anon=0
```

For example, if you want host machines *carrot* and *rutabaga* to access the clip libraries on the host machine *zucchini*, you would type the following on *zucchini*:

```
/usr/discreet/clip -rw,access=carrot:rutabaga,anon=0
```

**NOTE:** You can add as many hosts as you want to one file. You must type a colon (:) after each host name, except for the last one.

- Save the file and exit.
- Enable the changes you have just made by typing:

```
/usr/etc/exportfs -a
```

**NOTE:** To access a remote machine's exported clip library, make sure the autofs daemon is on, type: **chkconfig autofs on**.

[illegible]



# Setting Up a HIPPI Network

*Put your files in a commune*

*Configure and test your HIPPI network interface card and connect to the wire network.*

## Summary

In this chapter, you learn about:

- “HIPPI Components” on page 80
- “Configuring SGI HIPPI Cards” on page 81
- “Configuring Essential HIPPI Cards” on page 84
- “Testing a HIPPI Network” on page 86
- “Discreet HIPPI Utilities” on page 97

## About Setting Up a HIPPI Network

Your system should already have the required hardware installed and have all the necessary files updated. Use this chapter to:

- Verify that your HIPPI network is configured correctly.
- Verify that you have all the necessary components installed.

If you find that your system is not set up properly, contact your Discreet representative.

Depending on your platform and configuration, the procedure for setting up your HIPPI network will vary. The following chart describes the steps you need to follow and points you to the relevant information for your configuration.

To:	Refer to:
Check for required HIPPI components	"HIPPI Components" on page 80
Configure the HIPPI card	For SGI HIPPI cards (Onyx/Onyx2): "Configuring SGI HIPPI Cards" on page 81 For Essential HIPPI cards (OCTANE, O2): "Configuring Essential HIPPI Cards" on page 84
Verify the files	For SGI HIPPI cards: "Verifying Files for SGI HIPPI Cards" on page 81 For Essential HIPPI cards: "Verifying Files For Essential HIPPI Cards" on page 85
Test the HIPPI network	"Testing a HIPPI Network" on page 86

## HIPPI Components

Depending on the platform and configuration you are using, the components you will require for setting up your HIPPI network will vary. Use the following table to determine what components you need:

Platform:	Necessary component:
Onyx	SGI HIPPI PCA (HIO board and VME board)
Onyx2	SGI XIO Serial HIPPI card or; SGI HIPPI PCA or; Essential PCI Serial HIPPI card (using a PCI card cage)
OCTANE	Essential PCI Serial HIPPI card (using a PCI card cage)
O2	Essential PCI Serial HIPPI card



**WARNING:** Only SGI-certified service technicians can install hardware in an Onyx or Onyx2 system. If such an installation is performed by a non-certified person, the warranty may become void.



## Configuring SGI HIPPI Cards

For HIPPI transfers over the **wire** network, Discreet applications use a fast transfer protocol called HIPPI-FP to transfer images and audio between two hosts. HIPPI-FP (sometimes referred to as HIPPI raw or character mode) is a low-level protocol that bypasses the TCP/IP protocol layer, giving a faster transfer rate than TCP/IP.

SGI HIPPI cards work in both HIPPI-FP mode and in TCP/IP mode. Configuring TCP/IP on the HIPPI interface allows you to use tools like *ftp* or *ping* through the HIPPI interface. The following files should be modified for use with SGI HIPPI cards:

- `/etc/hosts`
- `/usr/etc/hippi.imap`
- `/etc/config/netif.options`
- `/etc/config/ifconfig-2.options`

### Drivers for SGI HIPPI Cards

To complete the installation of the SGI HIPPI card, you must install the correct drivers. You can find the IRIX drivers for the SGI HIPPI cards at:

<http://www.sgi.com>

The current version of the IRIX HIPPI driver for IRIX 6.5.x is 3.3.1. You must be running a driver at least as recent as that version.

### Verifying Files for SGI HIPPI Cards

Make sure all necessary files have been updated. If the entries listed are missing or incorrect, contact Discreet Customer Support.

#### **/etc/hosts**

In the *hosts* file, the names of hosts included in the HIPPI network and their internet addresses need to be provided.

**NOTE:** The HIPPI network must be on a different sub-net from the ethernet (or other) network.

#### **To set up the *hosts* file:**

Make sure the following line exists for each host in the HIPPI network:

```
<ipaddress> <hippi hostname>
```

where `<ipaddress>` is the IP address of the host, and `<hippi hostname>` is the name of the interface for the host.

For example, to add the hippy host “hippi-russia” with an IP address of 172.16.50.115, this line should appear:

```
172.16.50.115  hippy-russia
```

The names of all HIPPI hosts and their Ethernet (TCP/IP) addresses need to be provided. As mentioned earlier, it is important that the TCP/IP through HIPPI address be on a different sub-net from the Ethernet (or other) network as in the following example:

```
# Default IP address for a new IRIS. It should be changed
# immediately to the address appropriate for your network.
# (The '192.0.2' network number is the officially blessed
# 'test' network.)

173.12.24.110 potatoe
173.12.24.111 cucumber
173.12.24.112 onion
# TCP over the HIPPI cards
173.12.25.110 potatoe-h
173.12.25.111 cucumber-h
173.12.25.112 onion-h
# This entry must be present or the system will not work.
127.0.0.1  localhost
```

**/usr/etc/hippi.imap**

The file */usr/etc/hippi.imap* defines the link between a host name and the HIPPI I-field address. All hosts on the HIPPI network must be defined in this file. The IP address should correspond to the address found in */etc/hosts*. The I-field address should correspond to the address of the port on the switch to which the host is connected.

**To set up the */usr/etc/hippi.imap* file:**

- 1. Verify that the following line exists for each host in the HIPPI network:

```
<ipaddress> <ifield> [<ULA>]
```

Where:	Is:
<ipaddress>	The IP Address of the host interface.
<ifield>	The unique identification address of the HIPPI host within your local network. The last digits of the I-Field should always match the switch slot ID to which the card is connected. If no switch is used (point to point) , assign the value 01 and 02 to each host. The I-Field count starts at zero (0).
<ULA>	The ULA is the hardware address of the NIC and is also called the MAC address (optional).

For example, if the host interface hippy-russia is at I-field 2, the following line should appear:

```
172.16.50.115 0x03000002
```

### **/etc/config/netif.options**

The HIPPI interface needs to be enabled in the */etc/config/netif.options* file. Only the local interface should be defined in *netif.options*. For example:

```
if2name=hip0
```

```
if2addr=hippy-russia
```

### **/etc/config/ifconfig-2.options**

This file sets the netmask for the HIPPI interface. You will need to create this file to set the netmask for this TCP/IP HIPPI protocol.

A netmask is a 32-bit mask used to determine the number of bytes representing the network portion and the host portion of the IP address. It is also called a subnet mask, which is used to create subnets. For example, if an IP address is 192.100.10.4, and the netmask is 0xfffff00, the network part of the IP address is 192.100.10, and the host part of the IP address is 4.

1. To create the file, type:

```
cp /etc/config/ifconfig-1.options /etc/config/ifconfig-2.options
```

2. Open the file in a text editor. The following line must be present:

```
netmask 0xfffff00 up
```

The *up* variable forces *ifconfig* to mark the interface as active.

3. To activate *ifconfig* without rebooting, type **ifconfig hip0 up** at the prompt (root privileges required).

## Configuring Essential HIPPI Cards

Essential PCI HIPPI cards use a high speed transfer protocol called HIPPI-FP to transfer images and audio between two hosts. HIPPI-FP (sometimes referred to as HIPPI character mode) is a low level protocol that bypass TCP/IP protocol layer, and provides a much better transfer rate than TCP/IP.

**NOTE:** When using Essential PCI HIPPI cards, you must be running Stone and Wire ProductSW\_073 or greater to be able to use HIPPI-FP wire transfers. Otherwise, upgrade your Stone and Wire version if the Discreet application installation procedures have not already done so.

**To find out which version of Stone and Wire you have installed:**

1. Type:

```
/usr/discreet/sw/sw_ping
```

The following text appears:

```
Hello from /usr/discreet/sw/sw_ping
```

```
Using stonefs 1.05 Release; ProductSW_073
```

## Drivers for Essential HIPPI Cards

To complete the installation of the Essential HIPPI card, you need to install the correct drivers. You can find drivers for Essential PCI HIPPI cards at:

**Essential HIPPI version 1.0.6 for Irix6.5.x on octane/onyx2:**

[ftp.ods.com/esscom/release/fkpds/H\\_IP27IP30\\_6.5\\_1.0.6.tar](ftp.ods.com/esscom/release/fkpds/H_IP27IP30_6.5_1.0.6.tar)

**Essential HIPPI version 1.0.4 for Irix6.5.x on O2:**

[ftp.ods.com/esscom/release/flsdo/H\\_IP32\\_6.5\\_1.0.4.tar](ftp.ods.com/esscom/release/flsdo/H_IP32_6.5_1.0.4.tar)

You can also obtain information directly from Discreet at the following URL:

<http://www.discreet.com/support/>

## Verifying Files For Essential HIPPI Cards

Make sure all necessary files have been updated. If the following entries are missing or incorrect, contact Discreet Customer Support.

### **/etc/hosts**

When using in HIPPI-FP (or character) mode, no changes are necessary in the */etc/hosts* file. For working with the Essential HIPPI card in TCP/IP mode, refer to the *readme.txt* file that you downloaded with the driver.

### **/etc/config/netif.options**

The HIPPI interface needs to be enabled in the */etc/config/netif.options* file. Only the local interface should be defined in *netif.options*. For example:

```
if2name=ess0
if2addr=hippi-russia
```

### **/etc/config/ifconfig-2.options**

This file sets the netmask for the HIPPI interface. You will need to create this file to set the netmask for this TCP/IP HIPPI protocol.

A netmask is a 32-bit mask used to determine the number of bytes representing the network portion and the host portion of the IP address. It is also called a subnet mask. For example, if an IP address is 192.100.10.4, and the netmask is 0xfffff00, the network part of the IP address is 192.100.10, and the host part of the IP address is 4.

1. To create the file, type:
 

```
cp /etc/config/ifconfig-1.options /etc/config/ifconfig-2.options
```
2. Open the file in a text editor. The following line must be present:
 

```
netmask 0xfffff00 up
```

The *up* variable forces *ifconfig* to mark the interface as active.
3. To activate *ifconfig* without rebooting, type **ifconfig ess0 up** at the prompt (root privileges required).

### **/etc/init.d/Essnic.pre (for HIPPI-FP Transfer)**

Modify the */usr/Essential/hippi/bin/essarp* command line to specify the HIPPI switch slot to which the card is connected. If you are connecting two machines without a switch (point to point connection), use 01 for one machine and 02 for the other.

For example, if the NIC is plugged into slot 2 of the HIPPI switch, the *essarp* command line in the *Essnic.pre* file should look like the following:

```
/usr/Essential/hippi/bin/essarp -l 02
```

The Network Interface card must be put in “long” mode.

Add the following two lines in the *Essnic.pre* file to enable long mode:

```
/usr/Essential/hippi/bin/esshippi off
/usr/Essential/hippi/bin/esshippi on long
```

### **/etc/init.d/Essnic.post (For TCP/IP Through HIPPI Transfer)**

This file defines all hosts on the TCP/HIPPI network including the ULA address for each host. The ULA is a unique identifier found on each NIC. To obtain the ULA address for a network interface card, type:

```
essarp -h
```

Get the ULA for each host on the HIPPI network and modify each line according to the hardware configuration.

```
#Note that both these hosts have a ULA of all 0's, This is very
#common on hosts that do not support ARP. If possible, please use
#the real ULA
#COMMAND LINE          HOST          ULA          SWITCHID
/usr/Essential/hippi/bin/essarp -sonyx-h0:a0:88:1:0:b301
/usr/Essential/hippi/bin/essarp -soctane-h0:a0:32:0:0:d102
```

For each host, modify the *Essnic.post* file and run the following command to finish the Essential PCI HIPPI card configuration:

```
/usr/Essential/hippi/scripts/esssetup.sh
```

This script tunes the card and updates its Runcode (EPROM).

## **Testing a HIPPI Network**

It is recommended that you test the HIPPI network to make sure all components are installed and working properly. You can test your HIPPI network using the following tests:

- Internal loopback tests
- Local loopback tests
- Switch loopback tests
- Point to Point tests

- HIPPI Network tests
- **wire** network tests

These tests can be performed for both SGI XIO HIPPI cards and Essential PCI HIPPI cards, however, the procedures are slightly different. Procedures for both Network Interface Cards are described below.

## Internal Loopback Tests

Use an internal loopback test to verify the interface card functionality. You can put the card into internal loopback mode and then use the `hiptest` program to perform the test.

### To use an internal loopback test for SGI HIPPI XIO:

**NOTE:** When manipulating the SGI driver with `hipcntl` commands, make sure that the `sw_server` process is not running on the machine. Otherwise, the `hipcntl` commands will not respond properly. To shut down the `sw_server`, as root, run `/usr/discreet/sw/sw_stop`

1. Type `/usr/etc/hipcntl shutdown`
2. Type `/usr/etc/hipcntl loopback`
3. Type `/usr/etc/hipcntl startup`
4. Type `/usr/etc/hiptest`

Lines similar to the following appear:

```
hiptest: /dev/hippi0:
sending 100 packets, size range [16..2097160], to I-field
0x01000001
hiptest(DST): received 64
hiptest(DST): received 100
hiptest: /dev/hippi0: Successfully transferred 100 HIPPI
packets.
```

5. To go back to network mode, type:
 

```
/usr/etc/hipcntl shutdown
/usr/etc/hipcntl startup
```

### To use an internal loopback test for Essential HIPPI cards:

1. Disable the interface by typing `ifconfig ess0 down`
2. Change to the `/bin` directory by typing `cd /usr/Essential/hippi/bin`

3. Disable *esshippi* by typing **esshippi off**
4. Enable the loopback test by typing **esshippi on loopback**
5. Run the *blast* command by typing **blast -I0x030000003 -m 10 -l 1000000**

Where *-I0x030000003* is the I-field value defined in */etc/init.d/Esenic.pre*. *blast* yields output similar to the following:

```
irix 34# blast -I0x030000002 -m 10 -l 100000
blast(ess0): 10 packets, length 100000, I=0x030000002, ulp=0x82
blast(ess0): 0.10 Meg, 0.00 s (0.00 MB/s)
blast(ess0): 0.10 Meg, 0.00 s (0.00 MB/s)
blast(ess0): 0.10 Meg, 0.01 s (9.54 MB/s)
blast(ess0): 0.10 Meg, 0.00 s (0.00 MB/s)
blast(ess0): 0.10 Meg, 0.00 s (0.00 MB/s)
blast(ess0): 0.10 Meg, 0.00 s (0.00 MB/s)
blast(ess0): 0.10 Meg, 0.01 s (9.54 MB/s)
blast(ess0): 0.10 Meg, 0.00 s (0.00 MB/s)
blast(ess0): 0.10 Meg, 0.00 s (0.00 MB/s)
blast(ess0): 0.10 Meg, 0.00 s (0.00 MB/s)
```

If you're not getting this output, make sure the card is visible using *hinvt*.

6. Disable *esshippi* by typing **esshippi off**
7. Turn on the network by typing **esshippi on network**
8. Enable the interface by typing **ifconfig ess0 up**

## Local Loopback Tests

Use a local loopback test to validate the local network interface card and a single external cable.

### SGL XIO Serial HIPPI Card Tests

#### To use a local loopback test:

1. Use a cable to connect the external HIPPI SRC output to the DST output.
2. Type:  
**/usr/etc/hiptest**

Lines similar to the following appear:



```

hiptest: /dev/hippi0:
sending 100 packets, size range [16..2097160], to I-field
0x01000001
hiptest(DST): received 64
hiptest(DST): received 100
hiptest: /dev/hippi0: Successfully transferred 100 HIPPI
packets.

```

### Essential HIPPI Card Tests

Use blast to run loopback tests by blasting 100 packets of 1000000 bytes in length. You should see the yellow light on the Network Interface Card light up. For the following examples, an I-Field address of 0x03000003 is used.

#### To use a local loopback test:

1. Use a cable to connect the external HIPPI SRC output to the DST output.
2. Type:

```
/usr/Essential/hippi/bin/blast -I<ifield> -m 100 -l 1000000
```

where <ifield> is the I-Field Address. For example,

```
/usr/Essential/hippi/bin/blast -I0x03000003 -m 100 -l 1000000
```

The yellow light on the Network Interface Card should light up. If you get an error, you may have a hardware problem. One possibility is that you might have a faulty external HIPPI cable. Repeat the test with a different external HIPPI cable.

### Switch Loopback Tests

Use a switch loopback test to validate the switch port and configuration as well as cabling and the local network interface card.

#### Before You Begin

- Verify the physical connection by checking the status lights on the back of the switch. The LEDs on the switch are good indicators of functionality. The middle two lights on the parallel switch interface card should be on for each connection. For serial switch interface cards, the top two lights on the card should be on.
- If the lights are off, the cable is either swapped or damaged, or the interface card in the host is faulty.

### Connecting the Local Host to a Configured Port on the Switch

The following test will verify if packets can go through the switch and back to the sender host. Cables should be normally connected to the switch.

1. From a shell, type:

```
/usr/discreet/sw/debug/sw_sink -n 100
```

2. From a different shell, type:

```
/usr/discreet/sw/debug/sw_blast -I0x03000002 -m 10 -l
1000000
```

Where 02 is the switch ID for this machine.

3. On the receiving shell (where you launched *sw\_sink*), you should get:

```
./sw_sink
sink(ESS)(/dev/hippi/h0): receiving into 2097152 byte buffers.
One process ulp=130
```

If you get an error in the switch loopback test, you may have a hardware or switch configuration problem. Check the following:

- Verify that the I-field specified is correct.
- Verify the switch configuration to insure that the port is configured correctly.
- The host interface card or cables may be faulty. Perform the local loopback test with both cables.
- The switch port or interface card may be faulty. Try another port or card on the switch, modifying the I-field used for the test as required.

### Point to Point Tests

Use point to point tests to exercise the hardware in a local loopback mode or between two hosts connected directly or through a switch.

#### HIPPI FP Transfer (Between Essential Cards)

*blast* and *sink* are a low-level HIPPI test program pair designed to exercise the HIPPI interface at the lowest level. *blast* sends packets to a destination, and *sink* receives packets. These test routines can be found in */usr/Essential/hippi/bin* for the Essential PCI HIPPI NIC. By default, the blast program sends 500 2MB packets. The following steps describe the output of these programs under normal circumstances. You can also use *sw\_blast* or *sw\_sink* to perform this test.

**NOTE:** Sink for the Essential HIPPI card is not verbose. You can only rely on the messages described in the switch loopback test to see if the point to point tests work. If you are using the Essential NIC card, you can look at the lights on the NICs and on the switch to see if packets are being sent to the correct destination.

**To use blast:**

1. Change to the directory that contains the blast program. Type:

```
./blast -I <ifield>
```

where <ifield> is the I-Field Address. For example,

```
./blast -I0x03000002
```

Lines similar to the following appear:

```
blast(/dev/hippi0): sending 500 packets of length 2097152 to
I=0x03000002 (1000 meg total)
```

**To use the sink program:**

1. To change to the directory that contains the sink program, type:

```
./sink
```

Lines similar to the following appear:

```
/dev/hippi0 : receiving into 2097152 size buffers
64 packets received by child
128 packets received by parent
192 packets received by child
256 packets received by parent
320 packets received by child
384 packets received by parent
448 packets received by child
```

**To use a point to point test:**

1. On one host, execute the sink program.
2. On the other host, execute the blast program.

If blast returns an error, or sink does not log any packets, you may have a configuration or hardware problem. Check the following:

- Verify that the I-field specified is correct.
- The host interface cards or cables may be bad. Perform the local or switch loopback tests on both hosts.

### HIPPI FP Transfer (Between SGI Drivers)

1. On the receiving machine, type:

```
/usr/discreet/sw/debug/sw_sink
```

2. From the sending machine, type:

```
/usr/discreet/sw/debug/sw_blast -I0x03000002
```

Where 02 is the ID switch for the receiving machine.

On the sending machine shell you should get output similar to the following:

```
blast(SGI)(/dev/hippi0): packets=500 length=1048576 total
megs=500 ULP=130
blast(/dev/hippi0)(1): 500.00 Megs in 5.82 s Transfer
rate:(85.91 MB/s)
```

**NOTE:** For more information about *sw\_blast* and *sw\_sink*, see “Discreet HIPPI Utilities” on page 97.

### HIPPI FP Transfer (Between SGI and Essential Drivers)

To test HIPPI FP transfer between two different network cards (i.e. SGI and Essential), use *sw\_blast* and *sw\_sink*.

1. On the receiving machine, type:

```
/usr/discreet/sw/debug/sw_sink
```

2. From the sending machine, type:

```
/usr/discreet/sw/debug/sw_blast -I0x03000002
```

Where 02 is the ID switch for the receiving machine. You should obtain output similar to the following on the sending machine:

```
./sw_blast -I0x03000002

blast(ESS)(/dev/hippi/h0): packets=500 length=1048576 total
megs=500 ULP=130

.....
.....
.....
```

```
.....
.....
```

```
blast(/dev/hippi/h0)(1): 500.00 Megs in 6.28 s  Transfer
rate:(79.62 MB/s)
```

On the receiving machine you should get the following output confirming that the send process is executed.

```
lybia 7# ./sw_sink

sink(SGI)(/dev/hippi0): receiving into 2097152 byte buffers.
One process  ulp=130
```

```
.....
.....
.....
.....
.....
```

**NOTE:** For more information about *sw\_blast* and *sw\_sink*, see “Discreet HIPPI Utilities” on page 97.

## Network Tests

### netstat -nr

The *netstat* command symbolically displays the contents of various network-related data structures including the *ess0* (Essential) or *hip0* (SGI) interfaces. If these interfaces do not appear, verify the contents of the */etc/config/netif.options* file.

```
Internet:
DestinationGatewayNetmaskFlagsRefs      Use      Interface
Default      172.12.20.2   UGS 3      210      ef0
127.0.0.1     127.0.0.1     UH 19      32       lo0
192.12.20.1127.0.0.1   UGHS 3      4        lo0
192.12.100 193.12.100.20xxxxffff00U 0      0        ess0
```

### ifconfig *ess0* or *hip0*

This command assigns an address to a network interface and sets its parameters. *Ifconfig* is invoked at boot time from */etc/init.d/network* and defines the network address of the interfaces present on a given machine. This command can also be used to change an interface address or other operating parameters once the system is running (see man pages for details).

The output of *ifconfig ess0* should be similar to the following:

```
# ifconfig ess0
ess0: flags=4061<UP,NOTRAILERS,RUNNING,DRVLOCK>
      inet 193.169.25.74 netmask 0xffffffff00
chad 2# ifconfig hip0
hip0: flags=5041<UP,RUNNING,CKSUM,DRVLOCK>
      inet 172.55.50.113 netmask 0xffff0000
```

Where *UP* confirms that the interface is running. If the interface is not running, type **ifconfig ess0 up** to start the interface.

*RUNNING* should also be listed; otherwise, the interface is unusable.

### osview -i 1

This command displays the system's status second-by-second. It can be used to see which protocol is used to send packets when a transfer begins.

```
1: system
2: cpu
3: memory
4: network
5: other
0: all
```

### Ping

Use ping to verify the IP interface locally or between two hosts.

#### To use a network test:

1. Ping the local HIPPI interface.
2. Ping the remote HIPPI interface.

**NOTE:** Network tests are not applicable for Essential PCI HIPPI cards if the TCP/IP services are not configured.

### esshippi status (Essential Only)

This command provides driver settings. The following elements should be present if the physical connections are recognized and valid:

- ACCEPT ON LONG
- RUNCODE
- LINK

For example, *esshippi status* should generate output similar to the following:

```
irix 3# esshippi status
FLAGS: ACCEPT ON LONG HIPPI-FP NETWORK SWITCHED (RUNCODE_ON
LINK_ON) ess0
SRC connections : 0
SRC packets : 0
SRC rejects : 0
DST packets : 0
DST rcv on bad ulp: 0
DST hippi-le drop: 0
DST data errors: 0
DST sequence err: 0
DST sdic lost: 0
```

To reset the HIPPI card, type:

```
esshippi restart
```

### **network tests:esshippi version (Essential Only)**

Provides the version of the Essential HIPPI drivers. For a machine running Essential HIPPI driver version 1.0.4, *esshippi version* should yield the following:

```
irix3# esshippi version
Driver version 1.0.4
ess0: RunCode version 2.0.47 NIC 120f-0001 rev 02
```

See “Drivers for Essential HIPPI Cards” on page 84

### **hipcntl shutdown (SGI Only)**

This tool controls HIPPI adapters. *hipcntl shutdown* forces a hardware reset of the HIPPI adapter and puts the adapter in a non-operational state.

1. Log in as super user.
2. Shut down *sw\_server*. Type:
 

```
/usr/discreet/sw/sw_stop
```

 or
 

```
killall sw_server
```
3. Then, type:
 

```
hipcntl shutdown
```
4. Start *sw\_server*. Type:
 

```
/usr/discreet/sw/sw_start
```

**hipcntl startup (SGI Only)**

Enables a HIPPI adapter. At the prompt, type:

```
hipcntl startup
```

**hipcntl status (SGI Only)**

Returns status information for a given HIPPI adapter. At the prompt, type:

```
hipcntl status
```

The command returns output similar to the following:

Origin HIPPI Serial XIO Adapter

FLAGS: ACCEPTING DST.LNK\_RDY DST.FSYNC DST.OH8SYNC DST.SIG\_DET

SRC connections: 0

SRC packets : 0

SRC rejects : 0

SRC xmit retry: 0

SRC glink reset: 3

SRC glink lost: 0

SRC time outs: 0

SRC connects lost: 0

SRC parity errs: 0

SRC number bytes sent: 0

DST connections: 0

DST packets : 0

DST rcv on bad ulp: 0

DST hippi-le drop: 0

DST llrc : 0

DST parity : 0

DST frame/state err: 0

DST flag err : 0

DST illegal burst: 0

DST link rdy lost in pkt: 0

DST null connections: 0

DST ready errors: 0



DST bad packet starts: 0

DST number bytes received: 0

Flag	Description
LOOPBACK	The board is in internal loopback mode, so no packet will go through the external fiber optic cables.
DST.FSYNC	Internal 'goodness' metric (required for fiber optic link).
DST.LNK_RDY	Interface is synchronized with opposite end at the flag level (required for fiber optic link).
DST.OH8SYNC	Adapter detects lights on the interface (required for fiber optic link).
DST.SIG_DET	Interface is synchronized with opposite end at the symbol level (required for fiber optic link).
ACCEPTING	Interface is configured to accept packets (required for fiber optic link).

## Discreet HIPPI Utilities

### sw\_blast

*sw\_blast* is a low level application that checks the HIPPI FP (raw) connectivity for SGI and Essential configurations. We recommend that the remote host run *sw\_sink*, however, this is not a requirement.

Before executing this application, stop *sw\_server* using the *sw\_stop script*.

If *sw\_blast* does not execute correctly, problems with *sw\_server* are likely.

### Synopsis

```
sw_blast [-D <device>] -I <iaddr> [-l <length>] [-m <packets>]
[-n <loops>] [-u <ulp>]
```

Part	Description
-D <device>	Specifies the HIPPI device to test. By default, the first HIPPI device located is used.

Part	Description
-l <iaddr>	Required parameter that specifies the Interface Address of the machine that you are attempting to communicate with (i.e. 0x03000009).
-l <length>	The length of the packet to send; the number specified must be divisible by 8 (i.e. 1048576).
-m <packets>	Number of writes to the remote host per loop.
-n <loops>	Number of loops.
-u <ulp>	HIPPI port running <i>sw_sink</i> .

### Example

To blast the machine connected to slot 2 on the HIPPI switch, use the following command:

```
sw_blast -I0x03000002
```

For a more thorough test, send 2048 byte packets for 20 loops:

```
sw_blast -I0x03000002 -l 2048 -n 20
```

### Location

/usr/discreet/sw/debug

### Related Utilities

*blast*, *sink*, *sw\_sink*, *sw\_stop*

### sw\_sink

*sw\_sink* is a low level application that checks the HIPPI FP (raw) connectivity for SGI and Essential configurations. *sw\_sink* should run on the server being pinged with *sw\_blast*.

Before executing this application, stop *sw\_server* using the *sw\_stop* script.

If *sw\_sink* does not execute correctly, problems with *sw\_server* are likely.

### Synopsis

```
sw_sink [-D <device>] [-l <length>] [-n <packets>] [-u <ulp>]
```

Options

Part	Description
-D <device>	Specifies the HIPPI device to test. By default, the first HIPPI device located is used.
-l <length>	Length of the buffers to receive; this number must be divisible by 8 (i.e. 1048576).
-n <packets>	Number of packets to listen for. Once this number is reached, the application exits (indicating success).
-u <ulp>	HIPPI port that you wish to have <i>sw_sink</i> listen on.

Example

Typically, *sw\_sink* is used without parameters.

Location

`/usr/discreet/sw/debug`

Related Utilities

*blast, sink, sw\_blast, sw\_stop*

[illegible]

# Troubleshooting wire

*If it ain't broke...*

*This chapter explains how to identify and correct **wire** problems. It also describes the **wire** network utilities.*

## Summary

In this chapter, you learn about:

- “Basic wire Network Tests” on page 101
- “Problems and Solutions” on page 106
- “wire Utilities” on page 107

## Basic wire Network Tests

This section describes the most common networks tests that can help you make sure your **wire** network is operating correctly or pinpoint potential network problems. Some commands are described in more detail in the section entitled “wire Utilities” on page 107.

### Ping Test

Try to ping your local host from a client machine. If this works, ping all other machines that should be accessible through **wire**.

1. At the prompt, type:  
`ping <machine_name>`

If the ping fails, try using the machine's IP address (172.16.100.23) instead of its name. If this is successful, verify how the machine negotiates the name on the network. We recommend setting the machine's identification to first look at the local setup file, then validate on the network. On IRIX 6.5.x, the following change should be made to */etc/nsswitch.conf*:

```
hosts:  files nis dns
```

## sw\_ping

The basic functionality of the **wire** network can be tested using the *sw\_ping* program found in */usr/discreet/sw*.

The *sw\_ping* test program can be used to test the interface between the local host and a **wire** network daemon (*sw\_server*) running on a remote host.

### To use *sw\_ping*:

1. To go the sw folder, type:

```
cd /usr/discreet/sw
```

2. Type:

```
./sw_ping [-host <hostname>] [-loop <n>] [-size <pktsize>]
[-read] [-write]
```

<b>Where:</b>	<b>Is:</b>
<hostname>	The name of the host to ping
<n>	The number of times to execute this test
<pktsize>	The size of the read/write buffer (in bytes)

For example, type

```
./sw_ping -host russia
```

*sw\_ping* should yield the following:

```
MB/sec: 0.000000
Minimum MB/sec: 0.000000
Maximum MB/sec: 0.000000
Average MB/sec: 0.000000
Test program complete
```

If a *sw\_ping* from machine 1 to machine 2 is successful, the server on machine 1 is fine and the *sw\_host\_map* on machine 1 is also fine.

If a *sw\_ping* from machine 1 to machine 2 is unsuccessful, look at the *sw\_host\_map* on both machines and the server on machine 2.

3. After making changes, always stop and restart the network using the **sw\_stop** and **sw\_start** commands on both machines.

If the problem persists, verify that *sw\_server* is running (see “Verify That *sw\_server* is Running” on page 103).

**NOTE:** See “*sw\_ping*” on page 109 for more information on the *sw\_ping* command.

## Verify That *sw\_server* is Running

If a machine does not respond to *sw\_ping*, verify that the **wire** network daemon *sw\_server* is running on the remote host. The *network* daemon must be running on the server machines in order for the server to respond to *sw\_ping* commands sent by client machines.

1. In a shell type:

```
ps -ef | grep sw
```

This command must return at least seven process/lines containing */usr/discreet/sw/sw\_server*.

2. If less than seven processes are running, try to stop the server by typing:

```
/usr/discreet/sw/sw_stop
```

```
/usr/discreet/sw/sw_server &
```

3. If error messages appear, verify the licence information on the remote host or check the *sw\_host\_map* files of each machine for typographic or protocol errors. Also, review all hardware connections.

**NOTE:** The *sw\_server* command is linked to its shell. When you kill the shell, you also kill the *sw\_server* process. This can be avoided by typing *sw\_start* instead of *sw\_server*. The *sw\_start* script is not linked to the shell where the process is running. However, *sw\_start* does not provide error messages for licensing problems.

4. Repeat this test for all remote hosts.

## NFS Subsystems

Make sure you have the *nfs*, *nfs3* subsystems and all the required *nfs* patches that come with the software distribution.

To check that you have the proper NFS subsystems to run **wire**, type:

```
versions | grep nfs
```

## Network Performance

Use the `sw_ping` command to test network performance. For more significant results, run the test 100 times (-l 100).

1. At the prompt, type:

```
/usr/discreet/sw/sw_ping -h<host_hostname> -read -write -  
size 1000000 -loop 100
```

Part:	Description:
-host <hostname>	Is the name of the host to ping
-loop <n>	Executes this test n times
-size* <pktsize>	Reads/writes a buffer of pktsize bytes
-read	Reads a buffer from the remote host
-write	Writes a buffer to the remote host (non destructive)

Example for a NTSC (720 x 486 x 3) frame format, sent 100 times in read and write mode to host batman:

```
/usr/discreet/sw/sw_ping -h batman -r -w -s 1049760 -l 100
```

**NOTE:** See “sw\_ping” on page 109 for more information on the `sw_ping` command.

## Verify Clip Library Access

Verify that the local host has write permissions to the clip library directory of the remote host. If `/etc/exports` is properly set, the following test should be successful.

1. Try to access the clip library directory of the remote host in UNIX (as super-user). Type:

```
cd /hosts/<remote_machine>/usr/discreet/clip
```

2. If an error message appears on the client machine, check to see if automount is on by typing:

```
chkconfig -s
```

3. If automount is not on, enable it by typing:

```
chkconfig automount on
```

If automount is on and access error messages appear, check the `/etc/exports` file on the server.



4. If you have access to `/hosts/<remote_machine>/usr/discreet/clip`, make sure you have root privileges to read and write to this directory. Type:

```
mkfile 2k /hosts/<remote_machine>/usr/discreet/clip/
deleteme
```

If error messages appear, verify the `/etc/exports` file on the server machine. Read and write permission must be given to the hosts. The appropriate line should look similar to one of the following:

```
/usr/discreet/clip -rw,access=
/usr/discreet/clip -rw
```

For testing purposes, we recommend using exports without specifying the list machine access. If a conflict occurs in the name resolution, the export directory line to any user (`/usr/discreet/clip -rw`) bypasses this issue. Since network security can be affected by these settings, make sure to restore the original network settings once testing is complete.

If no errors are present, perform a final test by starting your Discreet application and transferring clips between a client machine and the server host. Transfer clips in both directions.

## Testing TCP Data Transfer

TTCP times the transmission and reception of data between two system using the UDP or TCP protocols (see man pages for details). This command lets you ascertain that TCP is functioning correctly.

This example sends 100 buffers of 1MB, and sets the `NO_DELAY` option on the TCP socket to simulate **wire**:

1. On the receiving side, type:

```
ttcp -r -s -l 1000000 -n 100
```

2. On the sending side, type:

```
ttcp -t -s -l 1000000 -n 100 -D <remote_hostname #1>
```

At the end of the test, the receiving side should provide performance statistics similar to the following:

```
ttcp -r -s -l 1000000 -n 100
ttcp-r: buflen=1000000, nbuf=100, align=16384/0, port=5001 tcp
ttcp-r: socket
ttcp-r: accept from 172.16.100.180
ttcp-r: 100000000 bytes in 113.69 real seconds = 858.93 KB/sec
```

```
+++
ttcp-r: 24874 I/O calls, msec/call = 4.68, calls/sec = 218.78
ttcp-r: 0.0user 2.7sys 1:53real 2% 556maxrss 0+0pf
24377+2003csw
```

3. In a different shell, run **netstat -C 1** to see where the packets are going to and which interface is used (TCP or HIPPI). More traffic should be apparent on the ef-0 interface. If the ess0 or hip0 interface is used instead, something may be wrong with the Ethernet interface.

## Problems and Solutions

### Debugging wire on multiple subnets

Scenario: 1 Fire and 2 Infernos on a Fiber Channel network. 1 Flint and 1 Flame running on 100BaseT.

When you are running on two different networks, you must pay close attention to the [INTERFACES] section in the host map files. Each host entry for the local machine may have more than one entry (i.e Fire and Infernos) that identifies which interface the sw\_server will allow connections over. The remote machines should have only one entry indicating over which interface the client (inferno, sw\_ping) will talk to the remote server.

Assuming that your fiber network is on subnet 21, and your ethernet interface is on 16 for the purposes of this example, the Fire host map file should contain the following:

```
[ INTERFACES ]
HOST=fire
PROT=TCP IADDR=172.16.100.1 DEV=1
PROT=TCP IADDR=172.21.100.1 DEV=1
HOST=inferno1
PROT=TCP IADDR=172.21.100.2 DEV=1
HOST=inferno2
PROT=TCP IADDR=172.21.100.3 DEV=1
HOST=flint
PROT=TCP IADDR=172.16.100.4 DEV=1
HOST=flame
PROT=TCP IADDR=172.16.100.5 DEV=1
```

The sw\_server on Fire should allow connections over both the ethernet and fiber network. Applications on the Fire host will talk to the inferno hosts over the fiber network, and the flint/flame hosts will use the ethernet network.

The `sw_host_map` files on `inferno1` and `inferno2` should look similar to the `Fire` example, i.e on `inferno1` you should see:

```
[ INTERFACES ]
HOST=fire
PROT=TCP IADDR=172.21.100.1 DEV=1
HOST=inferno1
PROT=TCP IADDR=172.16.100.2 DEV=1
PROT=TCP IADDR=172.21.100.2 DEV=1
HOST=inferno2
PROT=TCP IADDR=172.21.100.3 DEV=1
HOST=flint
PROT=TCP IADDR=172.16.100.4 DEV=1

HOST=flame
PROT=TCP IADDR=172.16.100.5 DEV=1
```

On the `flint` and `flame` you should see:

```
[ INTERFACES ]
HOST=fire
PROT=TCP IADDR=172.16.100.1 DEV=1
HOST=inferno1
PROT=TCP IADDR=172.16.100.2 DEV=1
HOST=inferno2
PROT=TCP IADDR=172.16.100.3 DEV=1
HOST=flint
PROT=TCP IADDR=172.16.100.4 DEV=1
HOST=flame
PROT=TCP IADDR=172.16.100.5 DEV=1
```

**NOTE:** If you make changes to the `sw_host_map` file you will need to restart the `sw_server` on that machine (`/usr/sw/sw_restart`).

## wire Utilities

### **sw\_lcheck**

Checks for a valid **wire** licence.

### **Synopsis**

```
sw_lcheck
```

### **Location**

```
/usr/discreet/sw
```

**sw\_server**

This is the **wire** server, and should only be started via the *sw\_start* script.

**Synopsis**

```
sw_server
```

**Environment**

For debug information, set the *SW\_NETWORK\_DEBUG* environment label. This variable displays simple messaging when information is sent or received. For information regarding processes, you can also set the environment label *SW\_DEBUG\_PID*. This variable displays process information at startup time.

**Location**

```
/usr/discreet/sw
```

**Related Utilities**

```
sw_start
```

**sw-start**

This scripts mounts the Discreet Filesystem and starts the **wire** server.

**NOTE:** Unlike *sw\_stop*, *sw\_start* does NOT start mountstone (if installed).

**NOTE:** The **wire** server will only operate if a valid **wire** licence is present.

**Synopsis**

```
sw_start
```

**Location**

```
/usr/discreet/sw
```

**Related Utilities**

```
sw_restart, sw_stop
```

## sw\_ping

Tests the *cfg/sw\_host\_map* file and communication methods to foreign hosts.

### Synopsis

```
sw_ping [-host <hostname>] [-size <bytes>] [-read] [-write] [-
loop <count>]
```

Part:	Description:
-host <hostname>	Specifies the hostname of the machine you wish to ping. If none is specified, the system will attempt to ping all systems defined in the <i>sw_host_map</i> file. The command exits when an attempt to ping fails.
-size <bytes>	Defines the size of each packet that will be sent. The default is 0.
-read	Tests the readability of data from the remote host.
-write	Tests the writeability of data to the remote host. This is not a destructive function.
-loop	Defines the number of times to ping the remote host. The default is 1.
-listsize <messages>	Specifies the number of commands to send per attempt.
-buffer	For the listsize option, -buffer will not use Buffers to send data, reducing performance.
-verify	The application sends a packet to the server, and the server is suppose to send the same packet back. This option will ensure that the packet sent back is the same as that sent.

### Environment

For debug information, you can set the *SW\_NETWORK\_DEBUG* environment label. This variable provides simple messaging when information is sent or received.

### Examples

The following example will ping the host named "pogo" by writing packets of 1100000 bytes in length using a loop value of 200 (200 iterations):

```
sw_ping -host pogo -write -read -size 1100000 -loop 200
```

If you receive errors, verify the *sw\_host\_map* configuration on both machines. Both files should include information about the remote host. You should also verify that *sw\_server* is running on “pogo”.

### Location

/usr/discreet/sw

### Related Utilities

sw\_blast, sw\_host\_map, sw\_server

### sw\_stop

A script that stops the **wire** server and mountstone (if installed).

**NOTE:** Unlike *sw\_start*, *sw\_stop* does NOT unmount Discreet Filesystems. If required, use *sw-unmount* to unmount the Discreet Filesystem before using *sw\_stop*.

### Synopsis

sw\_stop

### Location

/usr/discreet/sw

### Related Utilities

sw\_restart, sw\_start

### sw\_killall

This application is identical to *killall* and should only be used in conjunction with *sw\_stop*.

### Synopsis

sw\_killall <command>

Part:	Description:
<command>	Defines the executable to be removed from memory.

**Location**

/usr/dscreet/sw

**Related Utilities**

killall, sw\_stop

**sw\_restart**

This script stops and starts the Discreet Filesystem using *sw\_stop* and *sw\_start*. Since this kills *sw\_server*, you may wish to ensure that your coworkers are not connected to your machine using **wire**.

**NOTE:** This stops mountstone but does not start it up again.

**Synopsis**

sw\_restart

**Location**

/usr/dscreet/sw

**Related Utilities**

sw\_start, sw\_stop

**make\_host\_map**

This script creates a very simple *sw\_host\_map* configuration based on the local host. This is used when installing the Discreet Filesystem without parameters.

**NOTE:** You may wish to redirect the output of this script to a file. The program uses the last number sequence of the machine's IP address (205.236.124.143) as a unique Filesystem ID.

**Synopsis**

make\_host\_map [-f] [-h <ifield\_address>]

Part:	Description:
-f	Creates a complete configuration including a header and the configuration information for the localhost. You can then easily add other hosts.

Part:	Description:
-h <ifield_address>	Similar to -f but for use with a HIPPI connection ( <i>make_host_map</i> adds a HIPPI entry to the Interfaces sections).

### Example

To overwrite an existing configuration with a basic configuration, use the following command:

```
make_host_map -h 0x03000001 > /usr/discreet/sw/cfg/  
sw_host_map
```

### Location

/usr/discreet/sw/tools

### Related Utility

sw\_host\_map

## sw\_hostupdate

A maintenance system designed for large networks that use **wire** network technology. With *sw\_hostupdate*, you edit a single file and then update all *sw\_host\_map* files using the *sw\_hostupdate* update on all machines. The system includes other modes for use with a variety of commands.

### Synopsis

```
sw_hostupdate [-s <server>] [-a] [-v] [import] [add <host>]  
[remove <host>] [list] [set <host> HIPPI | TCP]
```

Part:	Description:
-a	When used before update, the system automatically adds all entries that are not currently in the <i>sw_host_map</i> and ensures that this information is correct.
-s <server>	Specifies the update server. The default matches the first line of <i>/usr/discreet/sw/cfg/sw_update_server</i> and the <i>SW_UPDATE_SERVER</i> environment label. This switch overrides both values.
-v	Enables Verbose mode.



Part:	Description:
add <host>	Adds the host <host> to the <i>sw_host_map</i> file as found on the server. An error occurs if the host is not found.
import	Generates an <i>sw_host_map</i> file based one the server's <i>sw_host_map</i> (using the localhost's information).
list	Lists all hosts currently configured on the local machine followed by all hosts that appear in the server's configuration (excluding those that are not on the localhost).
remove <host>	Removes the host <host> from the <i>sw_host_map</i> file. If the host does not exist, an error occurs.
set <host> [HIPPI   TCP]	Sets the network connection for host <host> to HIPPI or TCP/IP. This can be undone by update.
update	Assesses a configuration to ensure that <i>sw_host_map</i> is up-to-date. It does not automatically add entries if none exist. When update follows the -a option, it adds all entries that are not in the target <i>sw_host_map</i> file. It should be noted that it will undo any "set" statements.

### Example

To import the *sw\_host\_map* from the host "netstore", use:

```
sw_hostupdate -s netstore import
```

To update an *sw\_host\_map* to match the latest *sw\_host\_map* on the host netstore, use:

```
sw_hostupdate -s netstore update
```

To set the host TestHIP to use TCP/IP instead of HIPPI, use:

```
sw_hostupdate -s netstore set TestHIP TCP
```

You can also specify multiple commands per command line. The commands then are executed from left to right. Therefore, if you wish to import a hostmap from the server netstore, then add the machine TestWorld from the database on worldstore, and then set it to use TCP/IP instead of HIPPI, you would use:

```
sw_hostupdate -s netstore import -s worldstore add TestWorld  
set TestWorld TCP
```

## Notes

You will need to export the `/usr/discreet/sw/cfg` directory on your server. To do this, edit the `/etc/exports` file and add an entry similar to your `/usr/discreet/clip` entry. After this, execute `exportfs -va` to activate the configuration. Finally, you will need to create an `sw_master` file (see “sw\_master” on page 115).

To save time, you may wish to create a script similar to the following, replacing "TestHIP TestHOP TestIP" with the hosts on your network:

```
#!/bin/sh
#
# SWUpdate.sh Revision 1.0
#

for Name in TestHIP TestHOP TestIP
do
  rsh $Name /usr/discreet/sw/tools/sw_hostupdate $1
  $2 $3 $4 $5 $6 $7 $8 $9 $10 $11 $12
  rsh $Name /usr/discreet/sw/sw_restart
done
```

Once created, use `SWUpdate.sh -s [server] update`, replacing the commands as necessary. Should you have problems with `rsh`, see the `hosts.equiv` or `rsh` man pages.

## Environment

You can specify the `SW_UPDATE_SERVER` environment label instead of using the `-s` command line option (specifying the `-s` command line option overrides this environment label). Unlike other utilities, this application does not support the `SW_HOST_MAP` environment label.

## Files

Instead of using the command line `-s` option or using the `SW_UPDATE_SERVER` environment label, you could enter the update server name in `/usr/discreet/sw/cfg/sw_update_server`. This file is read before the `SW_UPDATE_SERVER` environment label and the `-s` command line option.

## Location

`/usr/discreet/sw/tools`

## Related Utilities

`sw_host_map`, `sw_master`

## sw\_master

*sw\_master* is a configuration file located on the server that is similar to *sw\_host\_map* used with *sw\_hostupdate*. It contains all the information specific to the *sw\_host\_map* for a given network.

Lines that start with the "#" character are ignored. Each line of the configuration file contains four fields that can be separated by spaces or tabs.

In the first column, enter the Hostname for the system. This is comparable to the text following the HOST= token in the [HOSTS] section of the *sw\_host\_map*.

In the second column, enter the system ID. This is the same information as follows the ID= token in the [HOSTS] section of the *sw\_host\_map*.

In the third column, enter the IField Address for the HIPPI network. If the given system is not on the HIPPI Network, simply enter "-" and go to the fourth column.

The fourth column is used to specify more complex HIPPI configurations. If the system is not on a HIPPI Network, enter "0". Otherwise, all machines that are on the same HIPPI network should have the same value. This option is only useful for networks that have more than one non-interlinked HIPPI switch and is ignored for TCP networks.

## Example

In the following example, the machines "TestBox" and "TestHop" have both HIPPI and TCP connections, and "TestIP" has only a TCP/IP connection:

```
# Sample /usr/discreet/sw/cfg/sw_master file
#
# Host:      ID:      IField:      SubNet:
# -----
TestBox      213      0x03000008      1
TestHop      78       0x03000009      1
TestIP       15       -      0
```

## Location

/usr/discreet/sw/cfg

## Related Utilities

*sw\_host\_map*, *sw\_hostupdate*

[illegible]

# Installing and Configuring mountstone

*Mount them!*

*mountstone allows third-party applications access to **stone** disk arrays by creating a Discreet Filesystem partition and presenting it as an NFS-mountable UNIX filesystem. It is a way to access unused space on your **stone** array as normal UNIX disk storage.*

## Summary

This chapter contains the following sections:

- “Installing mountstone” on page 121
- “Configuring a mountstone Server” on page 123
- “Verifying the mountstone Installation” on page 124
- “Configuring a mountstone Client” on page 125
- “Starting and Stopping mountstone” on page 127
- “Troubleshooting mountstone” on page 128

## About mountstone

mountstone allows client machines (which do not have a locally installed **stone** disk array) to attach to mountstone servers (machines with local **stone** disk arrays). You can access the framestore both locally and remotely and use mountstone concurrently with other Discreet applications, such as **flame**® and **inferno**®. mountstone can only be used with Discreet **stone** disk arrays.

By using SAMBA or other commercially available NFS servers, you can connect PCs running Windows 95/NT to a mountstone server. For more information, see Chapter 9, “Accessing mountstone from a PC.”

**NOTE:** mountstone 1.1.2 does not let you directly access application clip libraries.

The install checklist (page 120) lists the required items and tasks you need to perform to properly install mountstone. It is recommended that you archive the contents of your framestore before installing mountstone. You must have root privileges to install and start mountstone.

mountstone Requirements

mountstone 1.1.2 requires:

- NFS (available from SGI)
- Discreet Filesystem (available from Discreet)
- **stone** disk array (any series)

mountstone Architecture

The mountstone architecture is similar to the client-server architecture of the NFS filesystem. mountstone is comprised of the following components:

Component:	Function:
Mount Client program	This program is triggered on the client side whenever a “mount” request is made for a Discreet Filesystem. The <i>dfs</i> (Discreet filesystem type flag) identifies the filesystem being mounted as one managed by mountstone.
Mount Server daemon	The Mount Server daemon runs on every host with a <b>stone</b> accessible through mountstone. The daemon fields requests from local or remote hosts. It does not conflict with the standard system mount server daemon because it operates on its own communications port. The daemon does not use the system <i>/etc/exports</i> file to validate mount requests.
Server daemon	The server daemon processes NFS requests directed at a mounted Discreet filesystem. The server fields NFS requests by accessing the <b>stone</b> via the mountstone database. The daemon does not conflict with the standard system NFS server daemon because it listens for NFS requests on its own communications port.

Component:	Function:
Database server	The database server allows mountstone to manage the hierarchy of files and directories in its filesystem. The Discreet Filesystem stores material in a single-level hierarchy, which is insufficient to represent the multi-level hierarchies typical of UNIX filesystems. The database provides the necessary mechanism to represent the hierarchy of directories, leaving the <b>stone</b> free to store actual content of files. The database is stored on the host's system disk, separate from the contents of the <b>stone</b> array.

## Data Transfer

All input and output from a mounted **stone** occurs as a sequence of NFS requests from the client application to the server daemon. This places an upper bound on the I/O bandwidth, which is achieved using mountstone.

mountstone uses the Discreet Filesystem to create an invisible, dynamic partition. All data written through mountstone is tagged on the **stone** as mountstone data, which makes it possible to remove mountstone data without affecting data belonging to other Discreet applications. Likewise, initializing other Discreet products with the `-v` option will not affect mountstone data on a **stone** disk array.

**NOTE:** mountstone does not provide file conversion services. Therefore, users must use the Discreet application's import and export commands to read and write files to and from mountstone partitions.

## Request Processing

mountstone uses two daemons to handle requests from known ports.

- *rpc.dl\_mountd* starts when a request to mount a framestore comes in. This process mounts the framestore and dies. It runs in the foreground.
- *rpc.dl\_nfsd* handles requests for files on a mountstone partition. This process runs in the background.

## mountstone Software

mountstone 1.1.2 includes the following software components:

Component:	Function:
Discreet mountstone	This is the main product, which must be installed on machines with locally connected <b>stone</b> disk arrays. This product acts as a server and allows you to export a mountstone filesystem to other client machines on the network.
Discreet mountstone Client	Use to connect client machines to a mountstone server. This is the only component you must install on a mountstone client.
Discreet mountstone Database	A required component of a mountstone server. If you are installing mountstone for the first time, you must install this product (not required for mountstone clients).
Discreet Samba	Connects PCs running Windows 95 and Windows NT to a mountstone server. This is not a required component for either the server or client setups. See Chapter 6, “Accessing mountstone from a PC” for more information.

## Install Checklist

Use the following checklist to verify that you have the required items or have performed the necessary tasks to properly install and use mountstone:

Item/Task	✓	Description	Reference
mountstone 1.1.2 software		mountstone is included on the main application’s installation CD.	Discreet Customer Support (see “Getting More Help” on page 5 of this guide).
Verify NFS installation		NFS must be installed on your machine to run mountstone. The version of NFS installed depends on the IRIX version you are running.	Discreet product installation guide and product release notes.
Verify NFS patches installation		Make sure you have the correct NFS patch installed on your machine. The patch number differs, depending on the version of IRIX you are running.	Discreet product installation guide and product release notes.



Item/Task	✓	Description	Reference
Configure your disk array to use the Discreet Filesystem		The Discreet Filesystem is required to use mountstone.	See Chapter 2, "Setting Up the Discreet Filesystem" on page 9 of this guide.
Archive your framestore		It is recommended that you archive the contents of your framestore before installing mountstone.	Discreet product user's guide.
Install mountstone		Install mountstone.	See "Installing mountstone" on page 121 of this guide.

## Installing mountstone

You install mountstone on all computers on the mountstone network: both servers and clients. However, you install specific components of the software, depending on whether the computer is a client, a server, or both.

### To install mountstone:

1. Make sure you are logged in as root or have root privileges.
2. Load the main Discreet application CD into the CD-ROM drive.
3. Go to the directory containing the mountstone software by typing:  
**cd /CDROM**
4. Make sure that the CD-ROM drive is properly mounted. In a UNIX shell, type:

```
ls -la
```

If the drive is mounted, the contents of the CD are listed. Among the contents of the CD, you should see a directory called *mountstone\_1.1.2*.

5. Start the Software Manager by typing:

```
swmgr
```

The Software Manager window appears.

**NOTE:** The message `WARNING: Starting up as Read Only` appears if you have not logged in as root or do not have root privileges.

6. In the pink field next to Available Software at the top of the screen, type:

`./`

This tells the Software Manager to look in the current directory for the mountstone distribution.

7. Click the button labelled Customize Installation.
8. To select products for installation, click the buttons next to the product names.
  - On a mountstone server, you must install Discreet mountstone and the mountstone database. By definition, a mountstone server must be connected to a **stone** disk array.
  - On a mountstone client, you only need to install the Discreet mountstone Client program.

**NOTE:** See “mountstone Software” on page 120 for more information on the components included with mountstone.

9. Click Start.

A message appears indicating that the installing/removing procedure has started.

**NOTE:** Press the Stop button to stop the automatic installation at any time. The log window at the bottom of the Software Manager screen indicates the current activity.

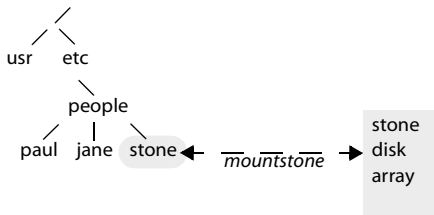
10. When you see the following message, Click OK.

Installations and removals were successful. You may continue with installations or quit now.

The installation procedure creates the directory */usr/discreet/MountStone*.

11. In the File menu of the window, choose Exit to complete the installation.

## Configuring a mountstone Server



Perform the following procedure to set up a mountstone server:

1. Create a mount directory using the *mkdir* command. This is where the **stone** will appear once mountstone is running.

**NOTE:** It is recommended that you create a new directory to function as your mount directory. Any files in an existing directory will be hidden once mountstone is running.

2. Use a text editor to add the next command on a single line in the */etc/fstab* file:

```
localhost:DL /<mount directory> dlfs
bg,retry=10,vers=3,hard,rsize=49152,wsiz=49152
```

Where	Is
localhost	The name of the machine to which the <b>stone</b> is connected
DL	The name of the local <b>stone</b> disk array
<mount directory>	The directory on your system disk where you will mount the <b>stone</b> disk array
dlfs	The type of filesystem (Discreet filesystem)
bg	The command that instructs the system to mount the program in the background
retry	The number of times to try starting mountstone before giving up
vers	The version of NFS (version 3 is recommended for mountstone 1.1.2)
hard	Instructs the program to continue retrying until the <i>retry</i> count is reached, instead of stopping after the first try
rsiz	The recommended read size (49152 for NFS 3)
wsiz	The recommended write size (49152 for NFS 3)

**NOTE:** The only configurable option is the mount directory. For maximum performance, the other options should be entered exactly as shown.

3. Save the file and close the editor.

This mounts your machine's local stone at *<mount\_dir>*, using NFS 3 and allows both read and write operations. It also runs the *rpc.dl\_mountd* daemon\* in the background if the first attempt to mount the filesystem is not successful.

4. Type:

```
chkconfig
```

This command shows the status of all configuration options. Make sure that *dlmountstone* is *on*. If not, type:

```
chkconfig dlmountstone on
```

5. Type:

```
cd /usr/discreet/MountStone/bin
```

6. To start mountstone, type:

```
dlmountstone start
```

## Verifying the mountstone Installation

We recommended that you perform some simple tests to verify that your mountstone is properly installed and running.

**To verify that mountstone is properly started:**

1. Change directory to the mount point by typing:

```
cd <mount directory>
```

2. Type:

```
df -k .
```

If mountstone is properly started, you should see a display similar to this example:

File system	Type	kbytes	use	avail	%use
localhost:DL	dlfs	#	#	#	#

The File system header should indicate that the local host is DL. The file type should be *dlfs* (Discreet filesystem).

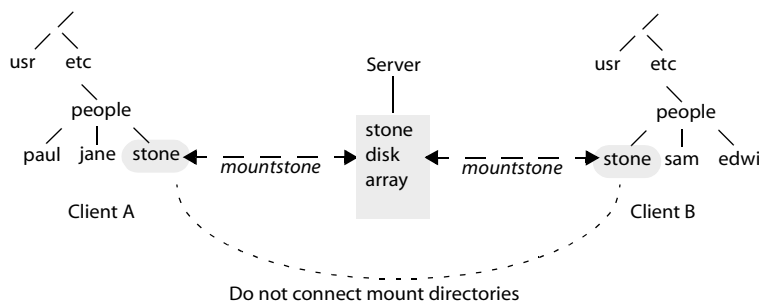
**NOTE:** # represent the values of your specific operating environment.

Once you verify that mountstone is properly started, you can perform some basic UNIX commands to verify that it is running.

For example:

1. Create a file called *deleteme* by typing:  
**mkfile 2k deleteme**
2. To verify that the file is in the correct location, type:  
**ls**
3. To delete the file, type:  
**rm deleteme**

## Configuring a mountstone Client



Perform the following procedures to mount and unmount a remote framestore.



**WARNING:** Do not mount an already mounted mountstone directory. Use the *DL* keyword as shown in the following examples.

### To mount a remote framestore:

1. Create a mount directory on the local machine using the *mkdir* command. This is where the **stone** will appear once mountstone is running.

**NOTE:** It is recommended that you create a new directory to function as your mount directory. Any files in an existing directory will be hidden once mountstone is running.

2. Install the client program (*mount\_dfls*) from the distribution using the software manager. For more information on installing the client program, see “Installing mountstone” on page 121.

- Ensuring that you enter it as one line, add the following to the file `/etc/fstab`:

```
<ms_server>:DL /<mount_directory> dlfs
bg,retry=10,vers=3,hard,rsize=49152,wsiz=49152
```

Where:	Is:
<ms_server>	The name of the mountstone server
DL	The <b>stone</b> disk array
<mount_directory>	The directory on your system disk where you will mount the <b>stone</b> disk array
dlfs	The type of filesystem (Discreet filesystem)
bg	The command that instructs the system to mount in the background
retry	The number of times to try starting mountstone before giving up
vers	The version of NFS (version 3 is recommended for mountstone 1.1.2)
hard	The command that instructs the program to continue retrying until the <i>retry</i> count is reached, instead of stopping after the first try
rsiz	The recommended read size (49152 for NFS 3)
wsiz	The recommended write size (49152 for NFS 3)

When you mount a remote **stone**, use the `:DL` keyword in the `/etc/fstab` file. For example, if the machine “aruba” has a **stone** that you wish to mount on the machine “valinor”, you would put the following in valinor’s `/etc/fstab`:

```
aruba:DL /aruba_stone dlfs
bg,retry=10,vers=3,hard,rsiz=49152, wsiz=49152
```

where `/aruba_stone` is a directory on valinor.



**WARNING:** Do not try to mount a server's mount directory on your local machine; using this setup will produce unpredictable results. Use the `:DL` keyword as described in the previous example.

- Save the file.
- Type:

```
mount -a -t dlfs
```

This forces the system to parse `/etc/fstab` and mounts the mountstone server on the client machine.

**To unmount all mounted mountstone servers, local and remote:**

As root, on the client machine, type:

```
umount -a -t dlfs
```

**To unmount a specific mountstone mounted directory:**

1. Make sure that you have root privileges.
2. Type:

```
umount <mountstone directory name>
```

For example, if you mounted aruba's **stone** on valinor in the `/aruba_stone` directory, to unmount the **stone**, type:

```
umount /aruba_stone
```

## Starting and Stopping mountstone

Use the following procedure if you need to start or stop mountstone manually:

**To start mountstone:**

1. Make sure that you have root privileges.
2. Type:

```
cd /usr/MountStone/bin
```

The following file is included in the `/bin` directory:

```
dlmountstone
```

3. To start mountstone, type:

```
dlmountstone start
```

**To stop mountstone:**

1. Make sure that you have root privileges.
2. Make sure that the mount directory is not occupied. If you attempt to unmount and the directory is occupied, you receive a resources busy message. To clear the directory, stop the offending application or cd out of the mount directory.
3. Type:

```
ps -flp `fuser -cq/<mount directory>`
```

If the directory is not occupied, you see the host name, and no other information. If the directory is occupied, you see the host name, the process owner, and the process.

4. Type:

```
/usr/discreet/MountStone/bin/dlmountstone stop
```

### **Preventing mountstone from starting at boot time**

To prevent mountstone from starting automatically at boot time, at the prompt type:

```
chkconfig dlmountstone off
```

## **Troubleshooting mountstone**

This section presents troubleshooting procedures.

### **mountstone data structure is still visible after replacing Volume 0**

In some cases, after re-initializing your Discreet Filesystem, mountstone may still erroneously show files and directories remaining. This indicates that mountstone's database has not been fully re-initialized. To correct this, proceed as follows:

1. Stop mountstone by typing:

```
/usr/discreet/MountStone/bin/dlmountstone stop
```

2. Change directories by typing:

```
cd . /usr/discreet/MountStone/lib/db/database/xfs_location
```

3. Remove all files in this directory by typing:

```
rm *
```

4. Type:

```
cd /usr/discreet/MountStone/bin/dlmountstone start
```





# Accessing mountstone from a PC

*Let's dance...*

*Explains how to connect to a mountstone partition using Samba.*

## Summary

In this chapter, you learn about:

- “About SAMBA for Windows” on page 129

## About SAMBA for Windows

The mountstone distribution contains the public domain SAMBA software. This software, which runs under IRIX 6.2, 6.3, 6.4 and 6.5, allows clients using the SMB (Server Message Block) protocol to communicate with NFS servers. The SMB protocol is used by all major Windows PC operating systems to allow applications to access remotely mounted drives. SAMBA allows PC applications to access mountstone without modification.

There are other commercially available packages that perform the same function, but SAMBA is distributed under the terms of the Free Software Foundation's GPL licence agreement. In accordance with this licence, the mountstone distribution contains the full source code for SAMBA.

## Installing SAMBA

SAMBA is not installed as part of the default mountstone installation.

### To install SAMBA:

1. Make sure you are logged in as root or have root privileges.
2. Load the mountstone 1.1.2 CD into the CD-ROM drive.
3. Go to the directory containing the mountstone software by typing:

```
cd /CDROM
```

4. Make sure that the CD-ROM drive is properly mounted. In a UNIX shell, type:

```
ls -la
```

If the drive is mounted, the contents of the CD are listed. Among the contents, you should see a file called *DLMountStone.sw*. If this file is listed, you are ready to install the software.

If this file is not listed, your CD-ROM may be improperly mounted.

5. Start the Software Manager by typing:  
**/usr/sbin/swmgr**
6. Select the CD-ROM as the source for the installation. A list of the CD's contents should appear.
7. Select the SAMBA subsystem for the installation. You can choose to install SAMBA source code, but it is not necessary to run SAMBA.
8. Begin the installation.
  - Once the installation is complete, SAMBA software will be in the directory */usr/discreet/MountStone/utls/samba*.
  - The default configuration file is located in */usr/discreet/MountStone/utls/samba/smb.conf*.
  - SAMBA man pages are found in the directory */usr/discreet/MountStone/utls/samba/man*.

## Running SAMBA

The mountstone server must be installed and running on the SGI workstation that you want to connect to.

### To start SAMBA manually:

1. Make sure that you are logged in as root or have root privileges.
2. Edit the file `/usr/discreet/MountStone/utils/samba/smb.conf`. In the "Mountstone" section, replace `path /usr/discreet/MountStone/mnt` with the mountstone mount directory. (i.e. If you mounted your stone at `/stone` then you would enter "path = /stone".)
3. Save the file, then exit.
4. At the prompt, type:  
`/usr/discreet/MountStone/utils/samba/bin/`
5. Type:  
`smbd`
6. Type:  
`nmbd`
7. From a Windows 95 or Windows NT machine on the same network as the mountstone server, press Start, select Find and choose "Computer".
8. Type the name of the mountstone server in the Named box. (i.e. If the name of the mountstone server is "aruba", type **aruba** in the "Named" field and press Find Now.)
9. If nmbd and smbd started properly and your mountstone server is running correctly, the name of the mountstone server appears. Double-click on the computer icon to access the files on the mountstone server.

**NOTE:** For a detailed description of Samba please refer to `/usr/discreet/MountStone/utils/samba/man/man5/smb.conf.5` (using `man -d <path>`).

**NOTE:** You must perform this procedure every time you reboot a machine.

### To start SAMBA automatically:

1. Use a text editor to add the following line to the `inetd.conf` file:  
`netbios-ssn stream tcp nowait root /usr/discreet/MountStone/`  
`utils/samba/bin/smbd smbd`
2. In the same file, type:  
`netbios-ns dgram udp wait root /usr/discreet/MountStone/`  
`utils/samba/bin/nmbd nmbd`

3. Use a text editor to add the following lines to the */etc/services* file by typing:

```
netbios-ssn 139/tcp
```

```
netbios-ns 137/udp
```

Once SAMBA is running on the SGI host, mountstone can be accessed from a Windows PC by mounting the `\\<sgi_host_name>\MountStone` network drive.

**To mount the network drive:**

1. From the Start menu, choose Find-Computer.
2. As the target of the search, type:  
`\\<sgi_host_name>\MountStone`

# index

---

## A

- active partitions 19
- Add partitioning mode 14
- adding a disk 14
- architecture
  - Discreet Filesystem 11
  - mountstone 118
- arrows
  - zoom 17
- audience 4

## B

- Bandwidth volume configuration parameter 16
- blast 91
- block size
  - standardizing across disks 38
- buttons
  - Configuration utility 15

## C

- Clear All partitioning mode 14
- Clear partitioning mode 14
- Configuration utility 13
  - buttons 15
  - configuring the disk array 20
  - Create FS (Filesystem) options 15
  - disk status 17
  - Partition Info 19
  - starting 13
  - troubleshooting 45, 46
  - volume configuration parameters 16
  - volume summary 17
  - zoom arrows 17

- configuring mountstone
  - client 125
  - server 123
- console
  - displaying detailed errors 44
- Create FS
  - No 15
  - Update 15
  - Wipe 15
  - Yes 15
- creating and configuring volumes 21
- Customer Support 5

## D

- Discreet Filesystem overview 12
- Discreet Filesystem utilities
  - disk\_serials 44
  - disk-summary 43
  - heal 43
  - stone-test 43
  - sw\_print\_config 43
  - swf-df 44
  - swf-frag 44
  - swf-fsck 44
  - sw-mount 42
  - swr\_print\_config 44
  - swr-set-config 42
  - swr-set-debug 42, 44
- Discreet storage overview 11
- disk
  - adding 14
  - address 19
  - bad disk error message 26
  - getting a new spare drive 35

- installing a spare disk 35
  - locating a bad disk 32
  - locating a spare disk 33
  - removing a bad disk 34
  - removing from the current volume 14
  - resizing 39
  - size 19
  - standardizing block sizes across disks 38
  - status 17
- disk array configuration 20
- disk\_serials 44
- disk-summary 43
- documentation
  - online PDFs 2
- drivers
  - Essential HIPPI 84
  - SGI HIPPI 81
- E
  - error message
    - bad disk 26
  - esshippi 87, 94, 95
  - Essnic.post 86
  - Essnic.pre 85
- F
  - fault-tolerance 10
  - file sharing 73
  - framestore organization 12
  - FS Inode Size volume configuration parameter 17
  - fx 43
- G
  - Groups volume configuration parameter 16
- H
  - heal 43
    - applying the new configuration 34
    - getting a new spare drive 35
    - installing a spare disk 35
    - locating a bad disk 32
    - locating a bad disk manually 29
    - locating a spare disk 33
    - removing a bad disk 34
    - running the heal utility 30
    - using a spare disk 27
    - verifying a spare disk 29
    - verifying the heal operation 31
    - without a spare disk 36
- help 13
- hinv 43
- hipcntl 87, 95, 96
- HIPPI
  - components 80
  - configuring Essential cards 84
  - configuring SGI cards 81
  - Discreet utilities 97
  - testing 86
- hiptest 87, 88
- host configuration file 20
- I
  - I/O driver 12
  - ifconfig 93
  - ifconfig-2.options 83, 85
  - installation
    - mountstone 121
    - overview 5
    - SAMBA 130
    - wire 73
  - intended audience 4
  - internal loopback tests 87
- L
  - Length volume configuration parameter 16
- licencing
  - wire 73
- local loopback tests 88
- loopback tests
  - internal 87
  - local 88
  - summary 86
  - switch 89
- M
  - make\_host\_map 111
  - mountstone
    - architecture 118
    - configuring a client 125
    - configuring a server 123
    - data transfer 119
    - install checklist 120

- installing 121
- request processing 119
- requirements 118
- software components 120
- starting and stopping 127
- troubleshooting 127, 128
- verifying an installation 124

multiple subnets 106

## N

- netif.options 83, 85
- netstat 93
- network protocols 72
- network tests
  - eshippi 94, 95
  - hipcntl 95, 96
  - ifconfig 93
  - netstat 93
  - osview 94
  - ping 94
- Next Id volume configuration parameter 16
- NFS subsystems 103
- notation conventions 4

## O

- Offset volume configuration parameter 16
- online documents 2
- osview 94
- overview
  - Discreet Filesystem 12
  - Discreet Storage 11
  - heal 25
  - HIPPI 79
  - mountstone 117
  - SAMBA 129
  - wire 72

## P

- parity 10, 12, 18
- Partition Info 18, 19
- partitioning mode
  - Add 14
  - Clear 14
  - Clear All 14
- PDF documents 2
- ping 94

- ping test 101
- point to point tests
  - Essential cards 90
  - SGI cards 92

## R

- RAID 10
- removing a bad disk from the array 34
- removing a disk from the current volume 14
- requirements
  - mountstone 118
  - wire 73
- resizing a disk 39

## S

- SAMBA 129
  - installing 130
  - mounting a network drive 132
  - starting automatically 131
  - starting manually 131
- serial number 19
- Set Dead 18
- Set Sick 18
- Set Well 18
- shutdown
  - accidental 47
- sink 91
- soft partitions 11, 12
- spare disk
  - installing 35
  - locating 33
- starting and stopping mountstone 127
- stone
  - installing a spare disk 35
  - locating a bad disk 32
  - locating a spare disk 33
  - removing a bad disk 34
- Stone + Wire 5
- stone-test 43
- storage concepts 10
- stripe 10, 11, 12
- sw\_blast 90, 92, 97
- sw\_host\_map 75
- sw\_hostupdate 112
- sw\_killall 110
- sw\_lcheck 107
- sw\_master 115

- sw\_ping 102, 109
- sw\_print\_config 43
- sw\_restart 111
- sw\_server 103
- sw\_sink 90, 92, 98
- sw\_stop 110
- swf-df 44
- swf-frag 44
- swf-fsck 44
- switch loopback tests 89
- sw-mount 42
- swr\_print\_config 44
- swr-set-debug 42
- sw-start 108

## T

- TCP data transfer 105
- testing a HIPPI network 86
- troubleshooting
  - Discreet Filesystem 45
  - mountstone 127, 128
  - wire 106
- troubleshooting tools
  - Configuration utility 42
  - disk\_serials 44
  - disk-summary 43
  - fx 43
  - heal 43
  - hinv 43
  - overview 42
  - stone-test 43
  - sw\_print\_config 43
  - swf-df 44
  - swf-frag 44
  - swf-fsck 44
  - sw-mount 42
  - swr\_print\_config 44
  - swr-set-config 42
  - swr-set-debug 42, 44
  - versions 44

## U

- UL 18
- using this guide 3

## V

- verbose error reporting
  - enabling 44
- versions 44
- volume configuration parameters 16
  - Bandwidth 16
  - FS Inode Size 17
  - Groups 16
  - Length 16
  - Next Id 16
  - Offset 16
- volume header
  - backing up during a heal 30
- volume summary 17
- volumes
  - creating & configuring 21
  - number of 21

## W

- wire
  - configuration 74
  - editing /etc/exports 77
  - editing sw\_host\_map 75
  - file sharing 73
  - licencing 73
  - performance test 104
  - problems and solutions 106
  - requirements 73
  - supported network protocols 72
  - troubleshooting 101, 106
  - verifying clip library access 104
- wire utilities
  - make\_host\_map 111
  - sw\_hostupdate 112
  - sw\_killall 110
  - sw\_lcheck 107
  - sw\_master 115
  - sw\_ping 109
  - sw\_restart 111
  - sw\_stop 110
  - sw-start 108

## Z

- zoom arrows 17



# Acknowledgments

---

## **Software Development Team:**

<b>Program Manager</b>	Annie Normandin
<b>Lead Engineer:</b>	Marc LaFleur
<b>Engineering Team:</b>	Le Huan Tran and Éric Thériault
<b>Product Testing:</b>	Éric Favreau

## **Support Specialists:**

<b>Team Leader:</b>	Dominique Bocquet
<b>Support team:</b>	Xavier Bourque, Pierre-Alexandre Vallée, Jacques Fortier, Marc Pageau and Patrick Baril

## **User's Guide Team:**

<b>Written and illustrated by:</b>	Louis Beaugard and Lisa MacDonald
<b>Edited by:</b>	Mylène Pepin
<b>Layout design:</b>	Brenda Barrie
<b>Section cover illustrations by:</b>	Alain Dubois



# **Discreet Filesystem and Networking\***

© Discreet Logic Inc., 1992-2000. All Rights Reserved.

This documentation contains proprietary information of Discreet Logic Inc. and its affiliates. No part of this documentation may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, recording or otherwise, without the prior written permission of Discreet Logic.

The End-User Software License Agreement which is included with the software program and this manual specifies the permitted and prohibited uses of the software program. Please read the terms and conditions of the Agreement before commencing use of the software program.

The information in this documentation was prepared by Discreet Logic with all reasonable care and is believed to be accurate. However, Discreet Logic and its affiliates shall not assume responsibility for losses or damages resulting from any omissions, inaccuracies, or errors contained herein. This documentation is subject to change, and revisions or new editions may be issued to incorporate such changes.

RESTRICTED RIGHTS LEGEND: The Software {and Documentation} is provided with RESTRICTED RIGHTS. Use, duplication or disclosure by the United States Government or any agency, department or instrumentality thereof is subject to the restrictions set forth in the Commercial Computer Software — Restricted Rights clause at FAR 52.227-19 or the Commercial Computer Software — Licensing clause at NASA FAR Supplement 1852.227-86, as applicable, or in successor provisions, including the limitations set forth in the End-User Software License Agreement included with the software program and this manual. Manufacturer is Discreet Logic Inc., 10 Duke Street, Montreal, Quebec, Canada, H3C 2L7.

**fire, flame, flint, inferno, stone,** and **wire** are registered trademarks and **Discreet** is a trademark of Autodesk Inc./ Discreet Logic Inc. in the USA and/or other countries. All other brand or product names are trademarks or registered trademarks of their respective owners.

Printed in Canada.

Title:	Discreet Filesystem and Networking Guide
Part No.	900-70271
Publication I.D.:	swm_082, mounstone 1.1.2
Date:	July 2000

# toc

## Table of Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
	Summary .....	1
	Welcome .....	1
	Documentation Set .....	2
	Documentation Available Online .....	2
	Using This Guide .....	3
	Intended Audience .....	4
	Notation Conventions .....	4
	Installation Overview.....	5
	About “Stone + Wire” .....	5
	Getting More Help .....	5
	 <b>Section 1: Discreet Filesystem</b>	 <b>7</b>
<b>2</b>	<b>Setting Up the Discreet Filesystem</b>	<b>9</b>
	Summary .....	9
	About Setting Up the Discreet Filesystem .....	9
	Storage Concepts .....	10
	RAID .....	10
	Soft Partitions .....	11
	Discreet Storage .....	11
	Discreet Filesystem .....	12
	Framestore Organization .....	12
	Configuration Utility.....	13
	Volume Configuration Parameters .....	16
	Volume Summary .....	17
	Zoom Arrows .....	17
	Hard Disk Area .....	17
	Configuring the Disk Array(s) .....	20
	Verify the Host Configuration File .....	20
	Determine the Number of Volumes You Need .....	21
	Create and Configure Volume(s) .....	21

<b>3</b>	<b>Healing a Bad Disk in a stone Disk Array</b>	<b>25</b>
	Summary .....	25
	About Heal .....	25
	Error Message in Discreet Applications .....	26
	Using the Correct Driver Version .....	26
	Using Heal .....	27
	Using heal with a Spare Disk in the Framestore Configuration .....	27
	Get Disk Information .....	27
	Locate a Bad Disk Manually .....	29
	Verify the Spare Disk .....	29
	Back Up the Volume Header Information .....	30
	Run the Heal Utility .....	30
	Verify the Heal Operation .....	31
	Physically Locate the Bad Disk .....	32
	Physically Locate the Spare Disk .....	33
	Remove the Bad Disk From the Array .....	34
	Apply the New Configuration .....	34
	Get a New Spare Drive .....	35
	Install the New Spare Drive .....	35
	Alternate Heal Procedure .....	36
	Standardizing Block Sizes Across Disks .....	38
	Resizing a Disk .....	39
 <b>4</b>	 <b>Troubleshooting the Discreet Filesystem</b>	 <b>41</b>
	Summary .....	41
	About Troubleshooting the Discreet Filesystem .....	41
	Basic Troubleshooting Tools .....	42
	Enable Verbose Error Reporting .....	44
	Problems and Solutions .....	45
	Disk/Controller Is Not Available in the Configuration Utility .....	45
	Configuration Utility Error: "Partition 7 does not exist" .....	46
	Configuration Utility Error: "Offset/Length out of range (X-+XXXXXXXX)" .....	46
	stone Disk Array Was Accidentally Shut Down .....	47
	Image Appears Striped Although No Drives Are Marked as Dead .....	47
	Cannot Access a Discreet Filesystem Volume .....	48
	VOLUMEMGT: WARNING: The disk array has 1 bad disk(s) .....	49
	stone Disk Is Marked as UL in the Configuration Utility .....	49
	Discreet Filesystem Utilities .....	50
	swf-purge .....	50
	sw-mount .....	51
	sw-unmount .....	52
	disk-summary .....	53
	swf-df .....	53

disk_serials .....	55
stone-test .....	55
swf-frag .....	57
heal .....	58
sw-fsck .....	59
swf-backupfs .....	59
swf-restorefs .....	60
swr_print_config .....	61
swr_get_config .....	62
swf-tweak .....	62
swr_restore_vh .....	64
mark_disk .....	64
swf-analys .....	65
swr-set-debug .....	65
swr_set_config .....	66
print_config .....	67
swr-set-license .....	67

## Section 2: Networking 69

### 5 Setting Up wire 71

Summary .....	71
About wire .....	72
Supported Network Protocols .....	72
Requirements .....	73
Main Features .....	73
File Sharing .....	73
wire Licencing .....	73
Setting Up wire .....	74
Verify autofs Is Enabled .....	74
Verify NFS Is Enabled .....	75
Edit the Host Configuration File .....	75
Edit the /etc/exports File .....	77

### 6 Setting Up a HIPPI Network 79

Summary .....	79
About Setting Up a HIPPI Network .....	79
HIPPI Components .....	80
Configuring SGI HIPPI Cards .....	81
Drivers for SGI HIPPI Cards .....	81
Verifying Files for SGI HIPPI Cards .....	81

Configuring Essential HIPPI Cards . . . . .	84
Drivers for Essential HIPPI Cards . . . . .	84
Verifying Files For Essential HIPPI Cards . . . . .	85
Testing a HIPPI Network . . . . .	86
Internal Loopback Tests . . . . .	87
Local Loopback Tests . . . . .	88
Switch Loopback Tests . . . . .	89
Point to Point Tests . . . . .	90
Network Tests . . . . .	93
Discreet HIPPI Utilities . . . . .	97
sw_blast . . . . .	97
sw_sink . . . . .	98

## 7 Troubleshooting wire 101

Summary . . . . .	101
Basic wire Network Tests . . . . .	101
Ping Test . . . . .	101
sw_ping . . . . .	102
Verify That sw_server is Running . . . . .	103
NFS Subsystems . . . . .	103
Network Performance . . . . .	104
Verify Clip Library Access . . . . .	104
Testing TCP Data Transfer . . . . .	105
Problems and Solutions . . . . .	106
Debugging wire on multiple subnets . . . . .	106
wire Utilities . . . . .	107
sw_lcheck . . . . .	107
sw_server . . . . .	108
sw-start . . . . .	108
sw_ping . . . . .	109
sw_stop . . . . .	110
sw_killall . . . . .	110
sw_restart . . . . .	111
make_host_map . . . . .	111
sw_hostupdate . . . . .	112
sw_master . . . . .	115

## 8 Installing and Configuring mountstone 117

Summary . . . . .	117
About mountstone . . . . .	117
mountstone Requirements . . . . .	118
mountstone Architecture . . . . .	118
Data Transfer . . . . .	119



Request Processing .....	119
mountstone Software.....	120
Install Checklist.....	120
Installing mountstone .....	121
Configuring a mountstone Server .....	123
Verifying the mountstone Installation .....	124
Configuring a mountstone Client .....	125
Starting and Stopping mountstone .....	127
Preventing mountstone from starting at boot time .....	128
Troubleshooting mountstone .....	128
mountstone data structure is still visible after replacing Volume 0 .....	128
 9 Accessing mountstone from a PC .....	 129
Summary .....	129
About SAMBA for Windows .....	129
Installing SAMBA .....	130
Running SAMBA .....	131
 <i>Index</i> .....	 133

