IBM Subsystem Device Driver Installation and User's Guide

Version 1 Release 2.0 July, 2000

IBM Subsystem Device Driver Installation and User's Guide

Version 1 Release 2.0 July, 2000

Note:
Before using this information and the product it supports, read the information in “Notices” on page 95.

Fourth Edition (July 2000)

This edition applies to Version 1 Release 2.0 of the IBM Subsystem Device Driver and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition also includes information that specifically applies to:

- AIX 4.2.1, 4.3.2, 4.3.3
- Windows NT Service Pack 3 or 4
- Windows 2000
- HP-UX 11.00
- Solaris 2.6
- Solaris 7

Order publications through your IBM representative or the IBM branch office serving your locality.

IBM welcomes your comments; send them to the following address:

International Business Machines
RCF Processing Department
G26/050
5600 Cottle Road
San Jose, CA 95193-0001
U.S.A.

You can also send your comments about this book electronically to:

- IBMLink™ from US and Canada: starpubs@us.ibm.com
- Internet: starpubs@us.ibm.com
- Fax (US): 1-800-426-6209

Contents

About this book

The IBM Subsystem Device Driver resides on either an AIX®, HP, Sun, Microsoft® Windows NT® or Microsoft® Windows 2000® host system that is attached to an IBM Enterprise Storage Server™. Using redundant paths between the server and the disk storage, it enhances performance and availability.

By balancing the input or output workload over multiple active paths, the Subsystem Device Driver:
- Permits applications to run without interruption when path errors occur
- Automatically balances the workload across paths
- Integrates transparently with applications

The Subsystem Device Driver supports the IBM Enterprise Storage Server.

This book includes the following information:
- Product overview
- Installation instructions
- Using the product

Summary of Changes for IBM Subsystem Device Driver Installation and User's Guide

The summary of changes informs you of changes to this book. Revision bars (|) in the left margin of the book indicate changes from the previous edition.

This book contains information previously presented in *Subsystem Device Driver Version 1 Release 1 April, 2000*.

The following sections summarize the changes to that information.

New information

This edition includes the following new information:

- A description of the Subsystem Device Driver's HACMP support in "Chapter 1. Introducing the IBM Subsystem Device Driver" on page 1.
- Pointer to the Subsystem Device Driver Web site.
- A section on installing fibre-channel device drivers and configuring fibre-channel devices in "Chapter 2. Installing and configuring the Subsystem Device Driver on an AIX host" on page 3.
- Graphics illustrating where the Subsystem Device Driver fits in the protocol stack for each of the supported host platforms.
- A graphic illustrating multipath connections between host server and ESS LUNs.
- A new chapter for HP hosts "Chapter 6. Installing and configuring the IBM Subsystem Device Driver on an HP host" on page 47.
- A new chapter for Sun hosts "Chapter 7. Installing and configuring IBM Subsystem Device Driver on a Sun host" on page 65.
- A new chapter for Windows 2000 hosts "Chapter 5. Installing and configuring the IBM Subsystem Device Driver on a Windows 2000 host" on page 41.

Chapter 1. Introducing the IBM Subsystem Device Driver

The IBM Subsystem Device Driver resides in the host server with the native disk device driver for IBM Enterprise Storage Server (ESS). It uses redundant connections between the host server and disk storage in an ESS to provide enhanced performance and data availability. These connections comprise many different components through which data flows during input and output processes. Redundancy and the ability to switch between these components provides many different paths for the data to travel. In the event of failure in one input-output path, it automatically switches to another input-output path. This automatic switching in the event of failure is called *failover*.

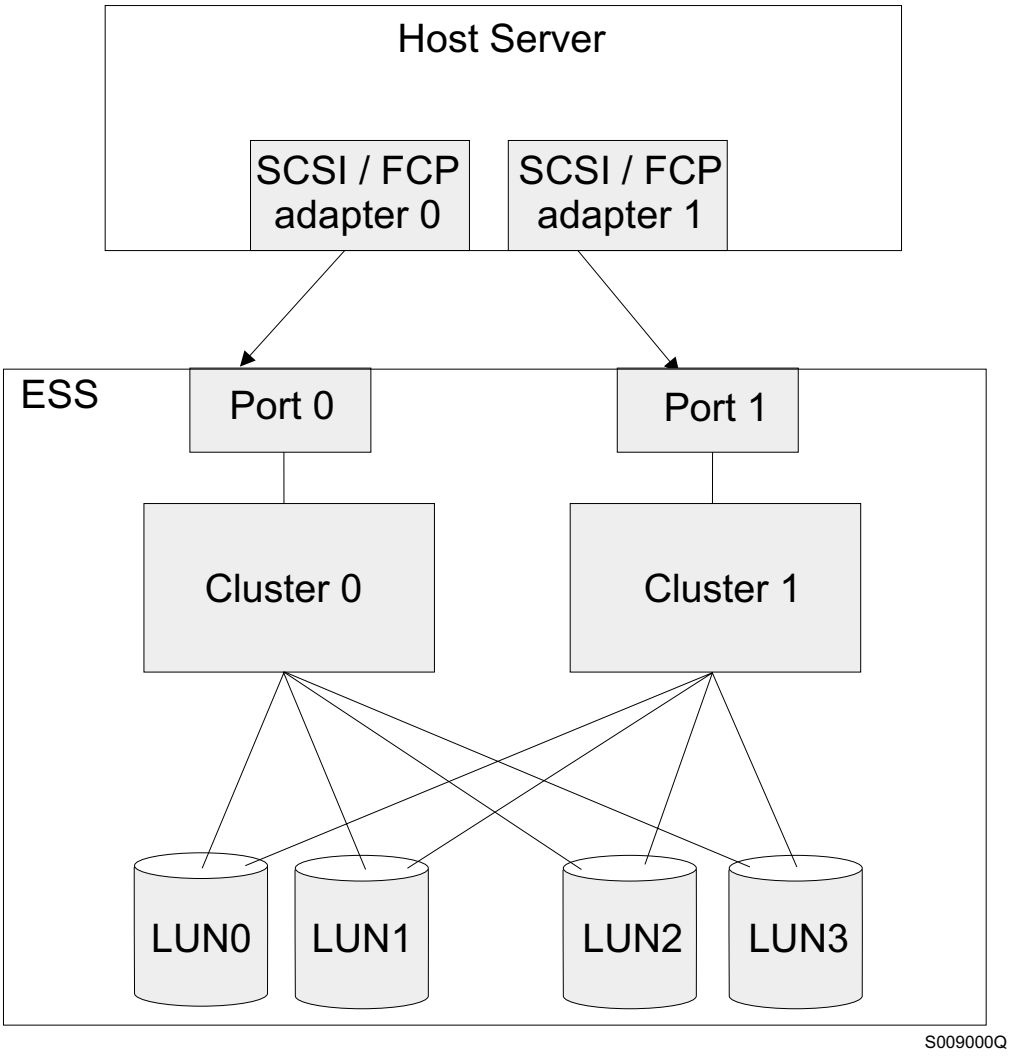


Figure 1. Multipath connections between host server and ESS LUNs

The IBM Subsystem Device Driver provides the following functions:

- Enhanced data availability
- Automatic path failover and recovery to an alternate path
- Dynamic load balancing of multiple paths
- Setting path selection policies (AIX only)

- Concurrent download of licensed internal code.

In most cases, host servers are configured with multiple host adapters with SCSI or fibre-channel connections to an ESS that, in turn, provides internal component redundancy. With dual clusters and multiple host interface adapters, the ESS provides more flexibility in the number of input-output (I/O) paths that are available. When there is a failure, the IBM Subsystem Device Driver reroutes I/O operations from the failed path to the remaining path. This prevents a bus adapter on the host server, an external SCSI cable, a fiber connector, a cluster on the ESS, or a host interface adapter on the ESS from disrupting data access.

In addition, multipath load balancing of data flow prevents a single path from becoming overloaded with I/O.

Concurrent download of licensed internal code is the capability to download and install licensed internal code on an ESS while applications continue to run. During the time when new licensed internal code is being installed in an ESS, the upper interface adapters inside the ESS may not respond to host I/O requests for approximately 30 seconds. The IBM Subsystem Device Driver makes such failures transparent to the host through its path selection and retry algorithms.

Note: For updated and additional information not included in this publication, see the README file on the IBM Subsystem Device Driver compact disk recordable (CDR) or visit the Subsystem Device Driver Web site www.ibm.com/storage/support/techsup/swtechsup.nsf/support/sddupdates

Chapter 2. Installing and configuring the Subsystem Device Driver on an AIX host

Notes:

1. You must have AIX 4.2.1, 4.3.2, or AIX 4.3.3 installed on your host server.
2. The Subsystem Device Driver supports 32-bit mode applications.
3. For updated and additional information not included in this manual, see the README file on the CDR or visit the Subsystem Device Driver Web site www.ibm.com/storage/support/techsup/swtechsup.nsf/support/sddupdates

This chapter provides instructions for installing and setting up the IBM Subsystem Device Driver on an AIX host system attached to a ESS. The *IBM Subsystem Device Driver/Data Path Optimizer on an ESS—Installation Procedures/Potential Gotchas* publication is a very helpful source of information. This is especially true if you have SP systems. This publication can be found on the web at www.ibm.com/storage/support/techsup/swtechsup.nsf/support/sddupdates

Understanding how Subsystem Device Driver works on an AIX host

Subsystem Device Driver failover system

The Subsystem Device Driver failover system is designed to provide recovery upon failure of a data path. If this situation occurs, the failover system selects an alternate data path, while minimizing any disruptions in operation. This failover process consists of:

- Detecting a failure
- Signaling to the AIX host that a failure has occurred
- Compensating for the failure by selecting an alternate data path.

The Subsystem Device Driver dynamically selects an alternate I/O data path when a software or hardware problem is detected.

Subsystem Device Driver

The Subsystem Device Driver resides above the AIX disk driver in the protocol stack. Each Subsystem Device Driver device represents a unique physical device on the storage subsystem. There can be up to 32 hdisk devices that represent up to 32 different paths to the same physical device.

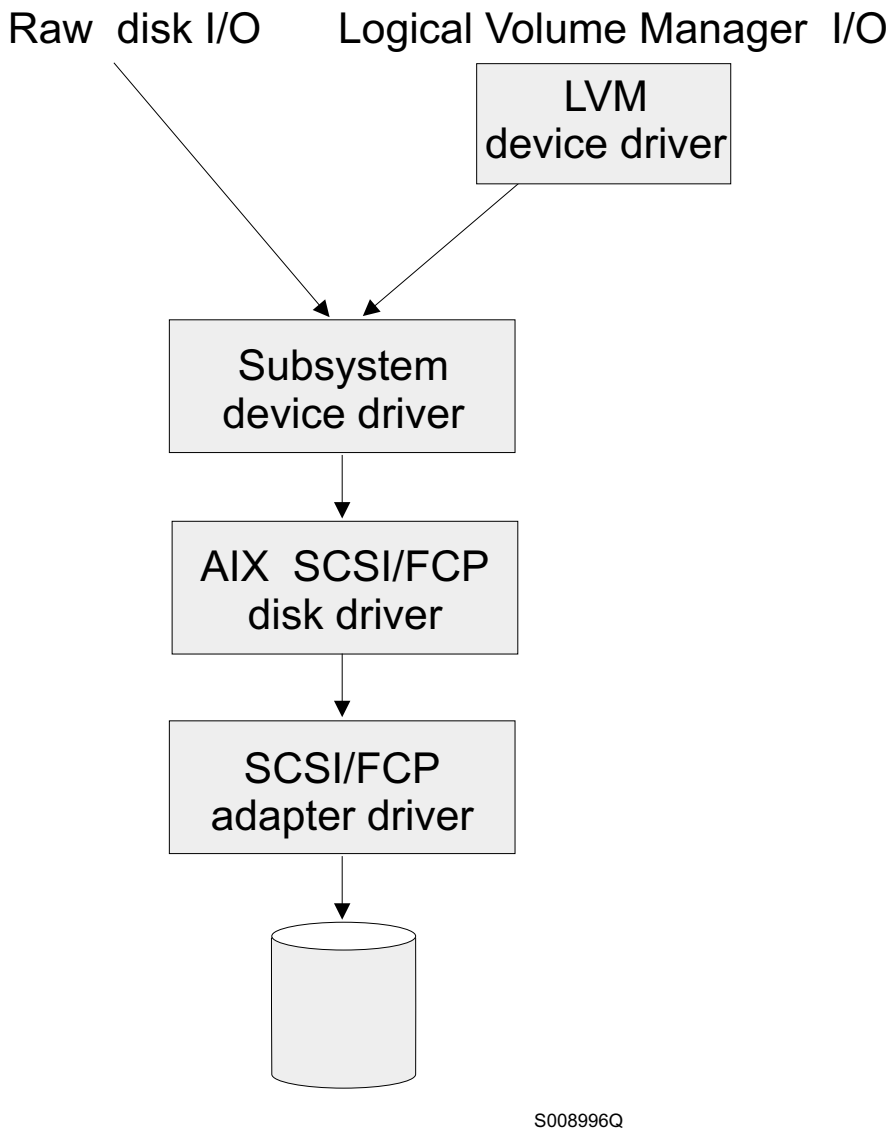


Figure 2. Where the Subsystem Device Driver fits in the protocol stack

Subsystem Device Driver devices behave almost like hdisk devices. Most operation on an hdisk device can be performed on the Subsystem Device Driver device, including commands such as **open**, **close**, **dd**, or **fsck**.

The Subsystem Device Driver acts as a pseudo device driver. I/Os sent to the Subsystem Device Driver are passed to the AIX disk driver after path selection. When an active path experiences a failure (such as a cable or controller failure), the Subsystem Device Driver dynamically switches to another path. The device driver dynamically balances the load, based on the workload of the adapter.

The Subsystem Device Driver also supports one SCSI adapter on the host system. With SCSI single-path access, concurrent download of licensed internal code is supported. However, the load balancing and failover features are not available.

Notes:

1. You cannot run the Subsystem Device Driver in a non-concurrent environment in which more than one host is attached to the same logical unit number (LUN) on a Enterprise Storage Server; for example, in a multi-host environment. This

restriction includes clustered hosts, such as RS/6000 servers running HACMP in non-concurrent mode, or RS/6000 SP servers running RVSD. However, concurrent multi-host environments are supported. See “High Availability Cluster Multi-Processing (HACMP)” on page 18 for more information on HACMP support.

Hardware and software requirements

To install the Subsystem Device Driver you must have AIX 4.2.1, 4.3.2, or 4.3.3 with the following fixes already installed on your system:

Table 1. AIX PTF Required Fixes

AIX 4.2.1	PTF IX62304	
	U451711	perfagent.tools 2.2.1.4
	U453402	bos.rte.libc 4.2.1.9
	U453481	bos.adt.prof 4.2.1.11
	U458416	bos.mp 4.2.1.15
	U458478	bos.rte.tty 4.2.1.14
	U458496	bos.up 4.2.1.15
	U458505	bos.net.tcp.client .2.1.19
	U462492	bos.rte.lvm 4.2.1.16
AIX 4.3.2	U461953	bos.rte.lvm 4.3.2.4

- The Enterprise Storage Server (ESS) devices must be configured for 2105.
- The IBM2105.rte ESS package must be installed.
- For SCSI support, ensure that the **bos.adt** package is installed on your system. The host system can be a uniprocessor or a multiprocessor system, such as SMP.
- For information on the SCSI adapters that can attach to your AIX host system, go to http://www.storage.ibm.com/hardsoft/products/ess/supserver.htm
- The maximum number of SCSI adapters that is supported is 32.
- You need a SCSI cable to connect each SCSI host adapter to an ESS port.
- For fibre-channel support, the AIX host system should be an IBM RS/6000® system with AIX Version 4.3.3. The AIX host system should have the fibre-channel device drivers installed along with APARS IY04636 and IY05369. Ensure that the **bos.adt** package is also installed on your system. The host system can be a uniprocessor or a multiprocessor system, such as SMP.
- For information on the fibre-channel adapters that can be used on your AIX host system go to http://www.storage.ibm.com/hardsoft/products/ess/supserver.htm
- The Emulex LP70000E adapter should be attached to its own PCI bus and the adapter should not be shared with other PCI adapters.

  Attention: If more than one adapter is attached to a PCI bus, both adapters' devices will be configured, but sometimes one adapter saturates the entire PCI bus and causes command timeouts.
- The RS/6000 Models S70, S7A, and S80 support the attachment of a maximum of four Emulex LP7000E adapters.
- The RS/6000 Models F50 and H50 support the attachment of a maximum of three Emulex LP7000E adapters.
- You need a fiber-optic cable to connect each fibre-channel adapter to a ESS port.

To install the Subsystem Device Driver and use the input-output load balancing and failover features, you need a minimum of two SCSI or fibre-channel adapters.

The Subsystem Device Driver also supports one SCSI adapter on the host system. With single-path access, concurrent download of licensed internal code is supported with SCSI devices. However, the load balancing and failover features are not available.

Notes:

1. A host server with a single-path fibre connection to an ESS is not supported.
2. A host server with SCSI connections and a single-path fibre connection to an ESS is not supported.

Configuring the ESS

Before you install the Subsystem Device Driver, configure your ESS for single-port or multiple-port access for each logical unit number (LUN). The Subsystem Device Driver requires that you provide a minimum of two independent paths that share the same logical unit to use the load balancing and failover features.

It is strongly recommended that the paths be on different clusters to achieve maximum availability.

For more information on configuring your IBM Enterprise Storage Server, see *IBM Enterprise Storage Server Introduction and Planning Guide*, GC26–7294.

Note: Ensure the IBM2105.rte install package is installed.

Installing fibre-channel device drivers and configuring fibre-channel devices

Notes:

1. For fibre-channel support, the AIX host system should be an IBM RS/6000 system with AIX Version 4.3.3. The AIX host system should have the fibre-channel device drivers installed, along with APARS IY04636 and IY05369.
2. AIX fibre-channel device drivers are developed by IBM for the Emulex LP7000E adapter.

This section contains the procedures for installing fibre-channel device drivers and configuring fibre-channel devices. These procedures include:

1. Installing the AIX fibre-channel device drivers
2. Installing the Emulex adapter firmware (sf222x1)
3. Configuring fibre-channel attached devices.

This section also contains procedures for:

- Removing fibre-channel attached devices
- Deinstalling fibre-channel device drivers.

Installing the AIX fibre-channel device drivers

Perform the following steps to install the AIX fibre-channel device drivers:

1. Install the fibre-channel device drivers from the AIX V4.3.3 compact disk recordable (CDR). The fibre-channel device drivers include the following filesets:

   devices.pci.df1000f7
   adapter device driver for Emulex LP7000E FCS adapter

   devices.fcp.disk
   FCP disk driver

devices.common.IBM.fc
FCP SCSI protocol driver

2. Check to see if APARS IY04636 and IY05369 are installed by issuing the **instfix -i | grep IY04636** and **instfix -i | grep IY05369** commands. If the APARS are listed, that means that they are installed. If they are installed, go to "Configuring fibre-channel attached devices", otherwise the APARS must be installed.
3. Install APARS IY04636 and IY05369.

Configuring fibre-channel attached devices

The newly installed devices must be configured before they can be used. There are two ways to configure these devices. You can:

- Use the **cfgmgr** command.
- Use the **reboot -q** command to reboot the system.

After the system reboots, use the **lsdev -Cc disk** command to check the ESS fibre-channel protocol (FCP) disk configuration. If the FCP devices are configured correctly, go to "Determining the Emulex adapter firmware level (sf222x1)" to determine if the proper firmware level is installed.

Determining the Emulex adapter firmware level (sf222x1)

You are required to install new adapter firmware only if the current adapter firmware is not at the sf222x1 level. Perform the following steps to download the Emulex adapter firmware:

1. First, determine the firmware level that is currently installed. Issue the **lscfg -vl fcsN** command. The adapter's vital product data is displayed.
2. Look at the **ZB** field. The **ZB** field should look something like this:

   ```
   (ZB).............S2F2.22X1
   ```

   To determine the firmware level, ignore the second character in the **ZB** field. In the example the firmware level is sf222x1.
3. If the adapter firmware level is at the sf222x1 level, there is no need to upgrade; otherwise, the firmware level must be upgraded. To upgrade the firmware level, go to "Upgrading the Emulex adapter firmware level".

Upgrading the Emulex adapter firmware level

Upgrading the firmware level consists of downloading the firmware (microcode) from your AIX host system to the adapter. Before this can be done, however, the fibre-channel attached devices must be configured. After the devices are configured, you are ready to download the firmware from the AIX host system to the FCS adapter. Perform the following steps to download the firmware:

1. Verify that the correct level of firmware is installed on your AIX host system. Locate the file called df1000f7.113.222.222.222.501 or df1000f7.113.222.222.222.502. It should be in the /etc/microcode directory. This file was copied into the /etc/microcode directory during the installation of the fibre-channel device drivers.
2. From the AIX command prompt, type `diag` and press Enter.
3. Select the **Task Selection** option.
4. Select the **Download Microcode** option.

5. Select all the fibre-channel adapters you want to download firmware to. Press F7. The Download window is displayed with one of the selected adapters highlighted. Press Enter to continue.
6. Type the filename for the firmware that is contained in the /etc/microcode directory and press Enter; or use the Tab key to toggle to **Latest**.
7. Follow the instructions that are displayed to download the firmware, one adapter at a time.
8. After the download is complete issue the **lscfg -v -l fcsN** command to verify the firmware level on each fibre-channel adapter.

Removing fibre-channel attached devices

To remove all fibre-channel attached devices, you must issue the **rmdev -dl fcsN -R** command for each installed FCP adapter. For example, if you have two installed FCP adapters (adapter 0 and adapter 1), you must issue two separate commands: **rmdev -dl fcs0 -R** and **rmdev -dl fcs1 -R**.

Note: *N* is the FCP adapter number.

Deinstalling fibre-channel device drivers

There are two methods for deinstalling all of your fibre-channel device drivers. You can:

- Use the **smitty deinstall** command.
- Manually deinstall the drivers using the **installp** command.

Perform the following steps to use the **smitty deinstall** command:

- Type **smitty deinstall** at the AIX command prompt. The Remove Installed Software panel is displayed.
- Press F4. All of the software that is installed is displayed.
- Select the filesets of the fibre-channel device driver you want to deinstall. Press Enter. The selected filename is displayed in the **Software Name Field** of the Remove Installed Software panel.
- Use the Tab key to toggle to **No** in the **PREVIEW Only?** field. Press Enter. The deinstall process begins.

Perform the following steps to use the **installp** command from the AIX command line::

1. Type **installp -ug devices.pci.df1000f7** and press Enter.
2. Type **installp -ug devices.common.IBM.fc** and press Enter.
3. Type **installp -ug devices.fcp.disk** and press Enter.

Installing the Subsystem Device Driver on an AIX host system

Note: The following procedures assume that the Subsystem Device Driver will be used to access all of your single and multipath devices.

You must have root access to install the Subsystem Device Driver and you must have AIX system administrator knowledge.

The following table lists the major files that are part of the Subsystem Device Driver install package.

Table 2. Major files included in the Subsystem Device Driver install package

Files	Description
defdpo	Define method of Subsystem Device Driver pseudo parent dpo
cfgdpo	Configure method of Subsystem Device Driver pseudo parent dpo
define_vp	Define method of Subsystem Device Driver vpath devices
cfgvpath	Configure methods of Subsystem Device Driver vpath devices
vpathdd	Subsystem Device Driver
hd2vp	Subsystem Device Driver conversion script to convert ESS hdisk device volume group to Subsystem Device Drive vpath device volume group
vp2hd	Subsystem Device Driver conversion script to convert Subsystem Device Driver vpath device volume group to ESS hdisk device volume group
datapath	Subsystem Device Driver driver console command tool
lsvpcfg	Subsystem Device Driver driver query configuration status command
mkvg4vp	Command to create Subsystem Device Driver volume group
extendvg4vp	Command to extend Subsystem Device Driver devices to Subsystem Device Driver volume group
dpovgfix	Command to fix a Subsystem Device Driver volume group with vpath and hdisk physical volumes mixed

To install the Subsystem Device Driver use the System Management and Interface Tool (SMIT). The Subsystem Device Driver is released as an install image. The fileset name is **dpo.ibmssd.rte.nnn**. Here *nnn* represents the AIX version level (4.2.1, 4.3.2). For example, the fileset name for the AIX 4.3.2 level is **dpo.ibmssd.rte.432**.

Perform the following SMIT steps to install the Subsystem Device Driver package on your system.

Note: Throughout this procedure, /dev/cd0 is used for the CDR drive address. This might be different in your environment.

1. Log in as the root user.
2. Load the CDR into the CD-ROM drive.
3. From your desktop window, type **smitty install_update** and press Enter to go directly to the installation panels. The Install and Update Software menu is displayed.
4. Highlight **Install and Update from LATEST Available Software** and press Enter. The screen is displayed.
5. Press F4 to display the INPUT Device/Directory for Software panel.

6. Select the CDR drive that you are using for the installation; for example, /dev/cd0, and press Enter.
7. Press Enter again. The Install and Update from LATEST Available Software panel is displayed.
8. Highlight **Software to Install** and press F4. The SOFTWARE to Install panel is displayed.
9. Select the correct dpo.ibmssd installation package for your operating system and press Enter. The Install and Update from LATEST Available Software panel is displayed with the name of the software you selected to install.

   Note: The fileset name for the Subsystem Device Driver is **dpo.ibmssd.rte.421** or **dpo.ibmssd.rte.432**. The AIX version level is included as part of the fileset name (421 or 432).
10. Check the default option settings to ensure that they are what you need.
11. Press Enter to install. SMIT responds with the following message:

```
ARE YOU SURE??
Continuing may delete information you may want to keep.
This is your last chance to stop before continuing.
```

12. Press Enter to continue. The installation process can take several minutes to complete.
13. When the installation is complete, press F10 to exit from SMIT. Remove the CDR.

Note: You can verify that the Subsystem Device Driver has been successfully installed by issuing the **lslpp -l dpo.ibmssd.rte.421** or **lslpp -l dpo.ibmssd.rte.432** command. For example, if you have successfully installed the **dpo.ibmssd.rte.432** package, the output from the **lslpp -l dpo.ibmssd.rte.432** command looks like this:

```
Fileset                      Level  State      Description
  ----------------------------------------------------------------------------
Path: /usr/lib/objrepos
  dpo.ibmssd.rte.432         1.2.0.0  COMMITTED  IBM Subsystem Device Driver
                                                 runtime for AIX V432

Path: /etc/objrepos
  dpo.ibmssd.rte.432         1.2.0.0  COMMITTED  IBM Subsystem Device Driver
                                                 runtime for AIX V432
```

Configuring the Subsystem Device Driver for an AIX host system

Before you configure the Subsystem Device Driver, ensure the ESS is on and the ESS hdisks are configured correctly on the AIX host system, and that the **dpo.ibmssd.rte** software is installed on the AIX host system. When multiple paths to a ESS device are configured on storage subsystems, make sure all paths (hdisks) are configured in the `Available` condition on the AIX host before the Subsystem Device Driver is configured. Otherwise, some Subsystem Device Driver devices will lose multiple-path capability. Use the **lsdev -Cc disk | grep 2105** command to check the ESS device configuration. If you have already created some ESS volume groups, you have to vary off (deactivate) all active volume groups with ESS subsystem disks, using the **varyoffvg** (LVM) command.

Note: Before you vary off a volume group, unmount all file systems of that volume group that are mounted. If some ESS devices (hdisks) are used as physical volumes of an active volume group, and there are file systems of that

volume group being mounted, then you must unmount all file systems, and vary off (deactivate) all active volume groups with ESS Subsystem Device Driver disks.

You can configure the Subsystem Device Driver on an AIX host system using SMIT.

Perform the following steps to configure the Subsystem Device Driver using SMIT:

1. Type **smitty device** from your desktop window. The Devices menu is displayed.
2. Highlight **Data Path Device** and press Enter. The Data Path Device panel is displayed.
3. Highlight **Define and Configure All Data Path Devices** and press Enter. The configuration process begins.
4. Check the Subsystem Device Driver configuration status. See "Displaying the ESS vpath device configuration" on page 21.
5. Enter the **varyonvg** command to vary on all deactivated ESS volume groups.
6. Mount the file systems for all volume groups that were previously unmounted.

Checking the Subsystem Device Driver configuration

To check the Subsystem Device Driver configuration use the SMIT Display Device Configuration panel or the **lsvpcfg** console command. In either case you see output similar to this:

```
vpath0 (Avail pv vpathvg) 018FA067 = hdisk1 (Avail )
vpath1 (Avail ) 019FA067 = hdisk2 (Avail )
vpath2 (Avail ) 01AFA067 = hdisk3 (Avail )
vpath3 (Avail ) 01BFA067 = hdisk4 (Avail ) hdisk27 (Avail )
vpath4 (Avail ) 01CFA067 = hdisk5 (Avail ) hdisk28 (Avail )
vpath5 (Avail ) 01DFA067 = hdisk6 (Avail ) hdisk29 (Avail )
vpath6 (Avail ) 01EFA067 = hdisk7 (Avail ) hdisk30 (Avail )
vpath7 (Avail ) 01FFA067 = hdisk8 (Avail ) hdisk31 (Avail )
vpath8 (Avail ) 020FA067 = hdisk9 (Avail ) hdisk32 (Avail )
vpath9 (Avail pv vpathvg) 02BFA067 = hdisk20 (Avail ) hdisk44 (Avail )
vpath10 (Avail pv vpathvg) 02CFA067 = hdisk21 (Avail ) hdisk45 (Avail )
vpath11 (Avail pv vpathvg) 02DFA067 = hdisk22 (Avail ) hdisk46 (Avail )
vpath12 (Avail pv vpathvg) 02EFA067 = hdisk23 (Avail ) hdisk47 (Avail )
vpath13 (Avail pv vpathvg) 02FFA067 = hdisk24 (Avail ) hdisk48 (Avail )
```

The output shows:

- The name of each pseudo device (for example, vpath13)
- The `Defined` or `Available` condition of a pseudo device
- Whether or not the pseudo device is defined to AIX as a physical volume (*pv* flag)
- The name of the volume group the device belongs to (for example, vpathvg)
- The unit serial number of the ESS LUN (for example, 02FFA067)
- The names of the AIX disk devices making up the pseudo device and their configuration and physical volume status.

Changing the path-selection policy for a Subsystem Device Driver device

The IBM Subsystem Device Driver supports path-selection policies to increase the performance of multipath-configured ESSs, and to make path failures transparent to applications. The following path-selection policies are supported:

load balancing (lb)
The path to use for an I/O is chosen by estimating the load on the adapter

that each path is attached to. The load is a function of number of I/Os currently in flight. If multiple paths have the same load, a path is chosen at random from those paths.

round robin (rr)
The path to use for each I/O is chosen at random from those paths not used for the last I/O. If a device has only two paths, the Subsystem Device Driver alternates between the two.

failover only (fo)
All I/Os for the device are sent to the same (preferred) path until the path fails because of I/O errors. Then an alternate path is chosen for subsequent I/Os.

The path-selection policy is set at the Subsystem Device Driver device level. The default path-selection policy for a Subsystem Device Driver device is load balancing. You can change the policy for a Subsystem Device Driver device with the **chdev** command.

Before changing the path-selection policy, determine the active attributes for the Subsystem Device Driver device. Type the **lsattr -El vpathN** command and press Enter. *N* represents the vpath-number, N=[0,1,2,...]. The output should look similar to this:

```
pvid         0004379001b90b3f0000000000000000 Data Path Optimizer Parent False
policy       df                               Scheduling Policy          True
active_hdisk hdisk1/30C12028                  Active hdisk               False
active_hdisk hdisk5/30C12028
```

The path-selection policy is the only attribute of a Subsystem Device Driver device that can be changed. The valid policies are *rr, lb*, *fo*, and *df*. Here are the explanations for these policies:

rr round robin

fo failover only

lb load balancing

df load balancing (default policy)

Attention: By changing a Subsystem Device Driver device's attribute, the **chdev** command unconfigures and then reconfigures the device. You must ensure the device is not in use if you are going to change its attribute. Otherwise, the command fails.

Use the following command to change the Subsystem Device Driver path-selection policy:

```
chdev -l vpathN -a policy=[rr/fo/lb/df]
```

Adding paths to a Subsystem Device Driver device that is part of a volume group

To activate additional paths to a Subsystem Device Driver device, the related Subsystem Device Driver devices must be unconfigured and then reconfigured. The Subsystem Device Driver conversion scripts should be run to enable the necessary Subsystem Device Driver associations and links between the Subsystem Device Driver vpath (pseudo) devices and the ESS hdisk devices.

Note: Ensure that logical volume sharing is enabled at the ESS for all applicable devices. Logical volume sharing is enabled using the ESS Specialist. See *IBM Enterprise Storage Server Web Interface User's Guide for ESS Specialist and ESS Copy Services*, SC26-7346, for information about enabling volume sharing.

Perform the following steps to activate additional paths to a Subsystem Device Driver device:

1. Identify the volume groups containing the Subsystem Device Driver devices that you want to add additional paths to. Use the command:

   ```
   lspv
   ```

2. Check if all the physical volumes belonging to that Subsystem Device Driver volume group are Subsystem Device Driver devices (vpathNs). If they are not, you need to fix the problem by typing the **dpovgfix vg-name** command and pressing Enter.

   Attention: This volume group problem needs to be fixed before proceeding to step 3.

   vg-name represents the volume group.

3. Identify the associated file systems for the selected volume group. Use the command:

   ```
   lsvgfs vg-name
   ```

4. Identify the associated mounted file systems for the selected volume group. Use the command:

   ```
   mount
   ```

5. Unmount the file systems of the selected volume group listed in step 3. Use the command:

   ```
   umount mounted-filesystem
   ```

6. Run the **vp2hd** volume group conversion script to convert the volume group from Subsystem Device Driver devices to ESS hdisk devices. Use the following command to run the script:

   ```
   vp2hd  vg-name
   ```

   When the conversion script completes, the volume group is in the `Active` condition (varied on).

7. Vary off the selected volume group in preparation for Subsystem Device Driver reconfiguration. Type the following command to do this:

   ```
   varyoffvg  vg-name
   ```

8. Run the AIX configuration manager **cfgmgr** to recognize all new hdisk devices. You can do this in one of two ways:

   - Run the **cfgmgr** command *n* times, where *n* represents the number of paths for the Subsystem Device Driver. (See Note on page 27 for an explanation of why **cfgmgr** should be run *n* times)
   - Run the **cfgmgr -l [scsiN/fcsN]** command for each relevant SCSI or FCS adapter.

   Note: Ensure that all logical drives on the ESS are seen as hdisks before continuing.

9. Unconfigure affected Subsystem Device Driver devices to the `Defined` condition, using the **rmdev -l vpathN** command. *N* represents the vpath-number to set to the `Defined` condition N=[0,1,2,...]. This command allows you to unconfigure only Subsystem Device Driver devices for which you are adding paths.

Note: Use the **rmdev -l dpo -R** command if you need to unconfigure *all* Subsystem Device Driver devices. The Subsystem Device Driver volume groups must be inactive before unconfiguring. This command attempts to unconfigure all Subsystem Device Driver devices recursively.

10. Use **System Management Interface Tool (SMIT)** to reconfigure Subsystem Device Driver devices.
    a. Enter **smitty device** from your desktop window. The Devices menu is displayed.
    b. Highlight **Data Path Devices** and press Enter. The Data Path Devices menu is displayed.
    c. Highlight **Define and Configure All Data Path Devices** and press Enter. SMIT executes a script to define and configure all Subsystem Device Driver devices that are in the `Defined` condition.

    Alternatively, you can run the **mkdev -l vpathN** command for each Subsystem Device Driver device.
11. Verify your datapath configuration using either SMIT or the command-line interface:
    a. Enter **Smitty device** from your desktop window. The Devices menu is displayed.
    b. Highlight **Data Path Devices** and press Enter. The Data Path Devices menu is displayed.
    c. Highlight **Display Data Path Device Configuration** and press Enter.

    Alternatively, you can run the **lsvpcfg** command to display Subsystem Device Driver configuration status.

    Note: Subsystem Device Driver devices should show two or more hdisks associated with each Subsystem Device Driver device when failover protection is required.
12. Vary on the volume groups selected in step 3 on page 13. Type the following command:

    ```
    varyonvg  vg-name
    ```
13. Run the **hd2vp** script to convert the volume group from ESS hdisk devices back to Subsystem Device Driver vpath devices. Type the following command:

    ```
    hd2vp vg-name
    ```
14. Mount all file systems for the volume groups that were previously unmounted.

Verify the Subsystem Device Driver configuration

Perform the following steps to verify configuration of the Subsystem Device Driver on an AIX host system:

1. Enter **smitty device** from your desktop window. The Devices menu is displayed.
2. Highlight **Data Path Device** and press Enter. The Data Path Device panel is displayed.
3. Highlight **Display Data Path Device Configuration** and press Enter. A list is displayed of the condition (either `Defined` or `Available`) of all Subsystem Device Driver pseudo devices, in addition to the multiple paths of each device. You see output similar to this:

```
vpath0 (Avail pv vpathvg) 018FA067 = hdisk1 (Avail )
vpath1 (Avail ) 019FA067 = hdisk2 (Avail )
vpath2 (Avail ) 01AFA067 = hdisk3 (Avail )
vpath3 (Avail ) 01BFA067 = hdisk4 (Avail ) hdisk27 (Avail )
vpath4 (Avail ) 01CFA067 = hdisk5 (Avail ) hdisk28 (Avail )
vpath5 (Avail ) 01DFA067 = hdisk6 (Avail ) hdisk29 (Avail )
vpath6 (Avail ) 01EFA067 = hdisk7 (Avail ) hdisk30 (Avail )
vpath7 (Avail ) 01FFA067 = hdisk8 (Avail ) hdisk31 (Avail )
vpath8 (Avail ) 020FA067 = hdisk9 (Avail ) hdisk32 (Avail )
vpath9 (Avail pv vpathvg) 02BFA067 = hdisk20 (Avail ) hdisk44 (Avail )
vpath10 (Avail pv vpathvg) 02CFA067 = hdisk21 (Avail ) hdisk45 (Avail )
vpath11 (Avail pv vpathvg) 02DFA067 = hdisk22 (Avail ) hdisk46 (Avail )
vpath12 (Avail pv vpathvg) 02EFA067 = hdisk23 (Avail ) hdisk47 (Avail )
vpath13 (Avail pv vpathvg) 02FFA067 = hdisk24 (Avail ) hdisk48 (Avail )
```

If any device is listed as `Defined`, the configuration was not successful. Check the configuration procedure again. See “Configuring the Subsystem Device Driver for an AIX host system” on page 10 for information about the procedure.

Perform the following steps to verify that multiple paths are configured for *each* adapter connected to a ESS port:

1. Enter **smitty device** from your desktop window. The Devices menu is displayed.
2. Highlight **Data Path Device** and press Enter. The Data Path Device panel is displayed.
3. Highlight **Display Data Path Device Adapter Status** and press Enter. All attached paths for each adapter are displayed.

Unconfiguring Subsystem Device Driver devices

Before you unconfigure the Subsystem Device Driver devices, all the file systems belonging to Subsystem Device Driver volume groups must be unmounted. Then, run the **vp2hd** conversion script to convert the volume group from Subsystem Device Driver devices (**vpathN**) to ESS subsystem devices (hdisks).

Using the System Management and Interface Tool (SMIT), you can unconfigure the Subsystem Device Driver devices in two ways. Either you can unconfigure ***without deleting*** the device information from the Object Database Management (ODM) database, or you can ***delete*** device information from the ODM database. If you unconfigure ***without*** deleting the device information, the device remains in the `Defined` condition. Using either **SMIT** or the **mkdev -l vpathN** command, you can return the device to the `Available` condition.

If you delete the device information from the ODM database, that device is removed from the system. To return it, follow the procedure described in “Configuring the Subsystem Device Driver for an AIX host system” on page 10.

Perform the following steps to unconfigure the Subsystem Device Driver devices:

1. Enter **smitty device** from your desktop window. The Devices menu is displayed.
2. Highlight **Devices** and press Enter. The Devices menu is displayed.
3. Highlight **Data Path Device** and press Enter. The Data Path Device panel is displayed.
4. Highlight **Remove a Data Path Device** and press Enter. A list of all Subsystem Device Driver devices and their condition (either `Defined` or `Available`) is displayed.
5. Select the device that you want to unconfigure. Select whether or not you want to delete the device information from the ODM database.

6. Press Enter. The device is unconfigured to the condition that you selected.
7. To unconfigure more Subsystem Device Driver devices you have to repeat Steps 4-6 for each Subsystem Device Driver device.

Notes:

1. The fast-path command to unconfigure *all* Subsystem Device Driver devices from the `Available` to the `Defined` condition is: **rmdev -1 dpo -R**.
2. The fast-path command to remove *all* Subsystem Device Driver devices from your system is: **rmdev -dl dpo -R**.

Removing the Subsystem Device Driver from an AIX host system

Before you remove the Subsystem Device Driver package from your AIX host system, all the Subsystem Device Driver devices must be removed from your host system. The fast-path **rmdev -dl dpo -R** command removes all the Subsystem Device Driver devices from your system. After all Subsystem Device Driver devices are removed, perform the following steps to remove the Subsystem Device Driver.

1. Type **smitty deinstall** from your desktop window to go directly to the Remove Installed Software panel.
2. Type **dpo.ibmssd.rte.421** or **dpo.ibmssd.rte.432** in the **SOFTWARE name** field and press Enter.
3. Press the Tab key in the **PREVIEW Only?** field to toggle between Yes and No. Select **No** to remove the software package from your AIX host system.

   Note: If you select **Yes**, the process stops at this point and previews what you are removing. The results of your pre-check are displayed without removing the software. If the condition for any Subsystem Device Driver device is either `Available` or `Defined`, the process fails.
4. Select **No** for the remaining fields on this panel.
5. Press Enter. SMIT responds with the following message:

   ```
   ARE YOU SURE??
   Continuing may delete information you may want to keep.
   This is your last chance to stop before continuing.
   ```
6. Press Enter to begin the process. This might take a few minutes.
7. When the process is complete, the Subsystem Device Driver software package is removed from your system.

If you are upgrading Subsystem Device Driver, go to “Installing the Subsystem Device Driver on an AIX host system” on page 8.

Upgrading the Subsystem Device Driver

Note: If you attempt to install Subsystem Device Driver *over* an existing version of Subsystem Device Driver or IBM Data Path Optimizer (DPO), the installation fails.

To upgrade the Subsystem Device Driver to a newer version, all the Subsystem Device Driver devices must be removed, and the existing Subsystem Device Driver must be uninstalled. If your application program accesses Subsystem Device Driver devices through AIX LVM, then you have to use Subsystem Device Driver’s conversion tools to convert all physical volumes of the Subsystem Device Driver volume groups into ESS hdisks before removing the Subsystem Device Driver. After installing and configuring the newer version of the Subsystem Device Driver, you

need to convert these physical volumes back from ESS hdisk devices to Subsystem Device Driver vpath devices. Perform the following steps to upgrade the Subsystem Device Driver:

1. Remove any .toc files generated during previous installs of Subsystem Device Driver or DPO. Enter the following command to delete any .toc file found in the **/usr/sys/inst.images** directory:

   ```
   rm .toc
   ```

   Ensure that this file is removed because it contains information about the previous version of Subsystem Device Driver or DPO.
2. Run the **lspv** command to find out all the Subsystem Device Driver volume groups.
3. Run the **lsvgfs** command for each Subsystem Device Driver volume group, to find out its mounted file systems.

   ```
   lsvgfs  vg_name
   ```
4. Run the **umount** command to umount all file systems belonging to Subsystem Device Driver volume groups.

   ```
   umount  filesystem_name
   ```
5. Run the **vp2hd** script to convert the volume group from Subsystem Device Driver devices to ESS hdisk devices.
6. Run the **varyoffvg** command to vary off the volume groups.

   ```
   varyoffvg  vg_name
   ```
7. Remove all Subsystem Device Driver devices.

   ```
   rmdev  -dl dpo -R
   ```
8. Use the **smitty** command to deinstall the Subsystem Device Driver. Type **smitty deinstall** and press Enter. The uninstall process begins. Complete the uninstall process. See “Removing the Subsystem Device Driver from an AIX host system” on page 16 for a step-by-step procedure on uninstalling the Subsystem Device Driver.
9. Use the **smitty** command to install the newer version of Subsystem Device Driver from the CDR. Type **smitty install** and press Enter. The installation process begins. Complete the installation process. See “Installing the Subsystem Device Driver on an AIX host system” on page 8 for a step-by-step procedure on installing the Subsystem Device Driver.
10. Use the **smitty device** command to configure all the Subsystem Device Driver devices to the `Available` condition. See “Configuring the Subsystem Device Driver for an AIX host system” on page 10 for a step-by-step procedure for configuring devices.
11. Run the **lsvpcfg** command to verify the Subsystem Device Driver configuration.

    ```
    lsvpcfg
    ```
12. Run the **varyonvg** command for each volume group that was previously varied offline.

    ```
    varyonvg  vg_name
    ```
13. Run the **hd2vp** script for each Subsystem Device Driver volume group, to convert the physical volumes from ESS hdisk devices back to Subsystem Device Driver vpath devices.

    ```
    hd2vp  vg_name
    ```
14. Run the **lspv** command to verify that all physical volumes of the Subsystem Device Driver volume groups are Subsystem Device Driver vpath devices.

Attention: If a Subsystem Device Driver volume group's physical volumes are mixed with hdisk devices and vpath devices, you must run the **dpovgfix** utility to fix this problem. Otherwise, the Subsystem Device Driver will not function properly. Use the **dpovgfix vg_name** command to fix this problem.

Concurrent download of licensed internal code

Concurrent download of licensed internal code is the capability to download and install licensed internal code on an ESS while applications continue to run. This capability is supported for single-path (SCSI only) and multiple-path (SCSI or FCP) access to an ESS.

Note: During the download of licensed internal code, the AIX error log might overflow and excessive system paging space could be consumed. When the system paging space drops too low it could cause your AIX system to hang. To avoid this problem, you can:

- Save the existing error report by using the following command:

  ```
  > errpt > file.save
  ```

- Delete the error log from the errlog buffer by typing this command:

  ```
  > errclear 0
  ```

- Enlarge the system paging space by using the SMIT tool.
- Stop the AIX error log daemon during the concurrent download, and resume the error log daemon after the concurrent download.

  Note: To stop the AIX error log daemon, type **/usr/lib/errstop** and press Enter. To start the AIX error log daemon, type **/usr/lib/errdemon** and press Enter.

High Availability Cluster Multi-Processing (HACMP)

You can run the Subsystem Device Driver in a *concurrent* multi-host environment in which more than one host is attached to the same logical unit number (LUN) on an Enterprise Storage Server. For example, RS/6000 servers running HACMP in a *concurrent* mode are supported.

HACMP provides a reliable way for clustered IBM RS/6000 servers that share disk resources to recover from server and disk failures. In an HACMP environment, each RS/6000 server in a cluster, is a node. Each node has access to shared disk resources that are accessed by other nodes. The Subsystem Device Driver supports RS/6000 servers connected to shared disks through either SCSI adapters and drives or fibre adapters and drives. The Subsystem Device Driver requires APAR IY07392 to support HACMP running in concurrent mode.

HACMP transfers ownership of shared disks and other resources when there is a failure based on how you define the relationship among nodes in a cluster. Basically, HACMP supports two modes of operation:

non-concurrent
: Only one node in a cluster is actively accessing shared disk resources while other nodes are idle.

concurrent
: More than one node in a cluster are actively accessing shared disk resources.

Even though the Subsystem Device Driver supports HACMP, certain combinations of features are not supported. The following table lists those combinations:

Table 3. HACMP and supported features

Feature	RS/6000 node in HACMP concurrent mode
ESS concurrent code load	Yes
Subsystem Device Driver load balancing	Yes
SCSI	Yes
FCP (fibre)	Yes
Single-path fibre	No
SCSI and single-path fibre (mixed environment)	No

Note: HACMP is not supported on all models of the ESS. For information about the ESS models that support HACMP, go to the Web site http://www.storage.ibm.com/hardsoft/products/ess/supserver.htm#6

Chapter 3. Using the IBM Subsystem Device Driver on an AIX host system

After you configure the Subsystem Device Driver, it creates Subsystem Device Driver devices (vpath devices) for ESS logical units (LUNs). ESS LUNs are accessible through the connection between the AIX host server SCSI or FCP adapter and the ESS ports. The AIX disk driver creates the original or ESS devices (hdisks). Therefore, with Subsystem Device Driver, an application now has two ways in which to access ESS devices.

To use the load balancing and failover features of the Subsystem Device Driver and access ESS devices, your application must use the Subsystem Device Driver vpath devices rather than the ESS hdisk devices.

Two types of applications use ESS disk storage. One type of application, such as Oracle or DB2®, accesses ESS devices directly through the Subsystem Device Driver vpath device (raw device). The other type of application accesses ESS devices through AIX logical volume management (LVM). For this type of application, you must create a volume group with the Subsystem Device Driver vpath devices.

Note: The Subsystem Device Driver does not support the following:

- Any application that depends on a Reserve/Release device
- More than 600 vpaths.

  The maximum number of vpaths the Subsystem Device Driver supports is 600. The maximum number of vpaths supported with AIX 4.2.1 might be less than 600 when there is one logical volume and volume group per vpath. This additional limit also applies to AIX 4.3.2 systems when the original install was performed with AIX 4.2.1 and then upgraded to AIX 4.3.2.

Providing load balancing and failover protection

Subsystem Device Driver provides load balancing and failover protection on AIX for applications and the LVM when ESS vpath devices are used. These devices must have a minimum of two paths to a physical logical unit (LUN) for failover protection to exist.

Displaying the ESS vpath device configuration

To provide failover protection, an ESS vpath device must include a minimum of two paths. Both the Subsystem Device Driver vpath device and the ESS hdisk devices must all be in the `Available` condition. In the following example, vpath0, vpath1, and vpath2 all have a single path and, therefore, will not provide failover protection because there is no alternate path to the ESS LUN. The other Subsystem Device Driver vpath devices have two paths and, therefore, can provide failover protection.

To display which ESS vpath devices are available to provide failover protection, use either the Display Data Path Device Configuration SMIT panel, or run the **lsvpcfg** command. Perform the following steps to use SMIT:

1. Enter **smitty device** from your desktop window. The Devices menu is displayed.
2. Select **Data Path Devices** and press Enter. The Data Path Devices menu is displayed.
3. Select **Display Data Path Device Configuration** and press Enter.

You will see output similar to this:

```
vpath0 (Avail pv vpathvg) 018FA067 = hdisk1 (Avail )
vpath1 (Avail ) 019FA067= hdisk2 (Avail )
vpath2 (Avail ) 01AFA067 = hdisk3 (Avail )
vpath3 (Avail ) 01BFA067 = hdisk4 (Avail ) hdisk27 (Avail )
vpath4 (Avail ) 01CFA067 = hdisk5 (Avail ) hdisk28 (Avail )
vpath5 (Avail ) 01DFA067 = hdisk6 (Avail ) hdisk29 (Avail )
vpath6 (Avail ) 01EFA067 = hdisk7 (Avail ) hdisk30 (Avail )
vpath7 (Avail ) 01FFA067 = hdisk8 (Avail ) hdisk31 (Avail )
vpath8 (Avail ) 020FA067 = hdisk9 (Avail ) hdisk32 (Avail )
vpath9 (Avail pv vpathvg) 02BFA067 = hdisk20 (Avail ) hdisk44 (Avail )
vpath10 (Avail pv vpathvg) 02CFA067 = hdisk21 (Avail ) hdisk45 (Avail )
vpath11 (Avail pv vpathvg) 02DFA067 = hdisk22 (Avail ) hdisk46 (Avail )
vpath12 (Avail pv vpathvg) 02EFA067 = hdisk23 (Avail ) hdisk47 (Avail )
vpath13 (Avail pv vpathvg) 02FFA067 = hdisk24 (Avail ) hdisk48 (Avail )
```

Figure 3. Output from the Display Data Path Device Configuration SMIT panel

The following information is displayed:

- The name of each Subsystem Device Driver vpath device; for example, vpath1.
- The configuration condition of the Subsystem Device Driver vpath device. It is either `Defined` or `Available`. There is no failover protection if only one path is in the `Available` condition. At least two paths to each Subsystem Device Driver vpath device must be in the `Available` condition to have failover protection.

  Note: Vpath0, vpath1, and vpath2 have only a single path and therefore will not provide failover protection. The other ESS vpath devices each have two paths and thus can provide failover protection. The requirement for failover protection is that the ESS vpath device, and at least two hdisk devices making it up, must be in the `Available` condition.

  Attention: The configuration condition also indicates whether or not the Subsystem Device Driver vpath device is defined to AIX as a physical volume (*pv* flag). If *pv* is displayed for *both* Subsystem Device Driver vpath devices and ESS hdisk devices, you might not have failover protection. Run the **dpovgfix** command to fix this problem.
- The name of the volume group the device belongs to (for example, vpathvg)
- The unit serial number of the ESS LUN; for example, 019FA067
- The names of the AIX disk devices that comprise the Subsystem Device Driver vpath devices, their configuration condition, and the physical volume status.

It is also possible to use the **datapath** command to display information about an Subsystem Device Driver vpath device. This command displays the number of paths that the device has. For example, the **datapath query device 10** command might produce this output:

```
DEV#:  10  DEVICE NAME: vpath10   TYPE: 2105B09   SERIAL: 02CFA067
===========================================================================
Path#      Adapter/Hard Disk   State     Mode    Select        Errors
    0          scsi6/hdisk21    OPEN   NORMAL        44             0
    1          scsi5/hdisk45    OPEN   NORMAL        43             0
```

This shows that device vpath10 has two paths and both are operational.

Configuring a volume group for failover protection

You can create a volume group with Subsystem Device Driver vpath devices using the Logical Volume Groups SMIT panel. The Subsystem Device Driver vpath devices included in the volume group should be chosen from those that can provide failover protection. It is possible to create a volume group that has only a single path (see Figure 3 on page 22) and then add paths later by reconfiguring the ESS. (See “Adding paths to a Subsystem Device Driver device that is part of a volume group” on page 12 for information on adding paths to a Subsystem Device Driver device.) With a physical volume with only a single path, failover protection will not be provided to that Subsystem Device Driver volume group.

To create a new volume group with Subsystem Device Driver vpaths, follow these steps:

1. Enter **SMIT** from your desktop window. The System Management Interface Tool is displayed.
2. Select **System Storage Management (Physical & Logical Storage)** and press Enter. The System Storage Management (Physical & Logical Storage) panel is displayed.
3. Select **Logical Volume Manager** and press Enter. The Volume Group menu is displayed.
4. Select **Volume Group** and press Enter. The Add Volume Group with Data Path Devices panel is displayed.
5. Select **Add Volume Group with Data Path Devices** and press Enter.

   Note: Press F4 while highlighting the **PHYSICAL VOLUME names** field to list all the available Subsystem Device Driver vpaths.

All the functions that apply to a regular volume group also apply to a Subsystem Device Driver volume group. Use SMIT to create a logical volume group (mirrored, striping, or compressed), or a file system (mirrored, striping, or compressed) on a Subsystem Device Driver volume group.

Once you create the volume group, AIX creates the Subsystem Device Driver vpath device as a physical volume (*pv*). In Figure 3 on page 22, vpath9 through vpath13 are included in a volume group and they become physical volumes. To list all the physical volumes known to AIX, use the **lspv** command. Any ESS vpath devices that were created into physical volumes are included in the output. The output should look similar to this:

```
hdisk0          0001926922c706b2    rootvg
hdisk1          none                None
...
hdisk10         none                None
hdisk11         00000000e7f5c88a    None
...
hdisk48         none                None
hdisk49         00000000e7f5c88a    None
vpath0          00019269aa5bc858    None
vpath1          none                None
vpath2          none                None
vpath3          none                None
vpath4          none                None
vpath5          none                None
vpath6          none                None
vpath7          none                None
vpath8          none                None
vpath9          00019269aa5bbadd    vpathvg
vpath10         00019269aa5bc4dc    vpathvg
vpath11         00019269aa5bc670    vpathvg
vpath12         000192697f9fd2d3    vpathvg
vpath13         000192697f9fde04    vpathvg
```

To display which devices comprise a volume group, enter the **lsvg -p vg-name** command. For example, the **lsvg -p vpathvg** command might produce the following output:

```
PV_NAME           PV STATE    TOTAL PPs   FREE PPs    FREE DISTRIBUTION
vpath9            active      29          4           00..00..00..00..04
vpath10           active      29          4           00..00..00..00..04
vpath11           active      29          4           00..00..00..00..04
vpath12           active      29          4           00..00..00..00..04
vpath13           active      29          28          06..05..05..06..06
```

This indicates that the **vpathvg** volume group uses physical volumes vpath9 through vpath13.

How failover protection can be lost

AIX can only create volume groups from disk (or pseudo) devices that are physical volumes. If a volume group is created using a device that is not a physical volume, AIX makes it a physical volume as part of the procedure of creating the volume group. A physical volume has a physical volume identifier (*pvid*) written on its sector 0 and also has a *pvid* attribute attached to the device attributes in the CuAT ODM. The **lspv** command can be used to list all the physical volumes known to AIX. Here is sample output from this command:

```
hdisk0          0001926922c706b2    rootvg
hdisk1          none                None
...
hdisk10         none                None
hdisk11         00000000e7f5c88a    None
...
hdisk48         none                None
hdisk49         00000000e7f5c88a    None
vpath0          00019269aa5bc858    None
vpath1          none                None
vpath2          none                None
vpath3          none                None
vpath4          none                None
vpath5          none                None
vpath6          none                None
vpath7          none                None
vpath8          none                None
vpath9          00019269aa5bbadd    vpathvg
vpath10         00019269aa5bc4dc    vpathvg
vpath11         00019269aa5bc670    vpathvg
vpath12         000192697f9fd2d3    vpathvg
vpath13         000192697f9fde04    vpathvg
```

In some cases, access to data is not lost, but failover protection might not be present. Failover protection can be lost in several ways:

- Loss of a device path

  Due to hardware errors, Subsystem Device Driver may remove one or more paths to a vpath pseudo device. Once the point is reached where only one path remains, failover protection for the pseudo devices no longer exists. The **datapath query device** command can be used to show the status of paths to a pseudo device. Any path whose condition is shown as `Dead` is no longer be used for I/O.

- Creating a volume group from single-path vpath (pseudo) devices

  A volume group created using any single-path pseudo devices does not have failover protection because there is no alternate path to the ESS LUN.

- Side effects of running the disk change method

  It is possible to modify attributes for an hdisk device by running the **chdev** command. The **chdev** command invokes the hdisk configuration method to make the requested change. In addition, the hdisk configuration method sets the *pvid* attribute for an hdisk if it determines that the hdisk has a *pvid* written on sector 0 of the LUN. This will cause the vpath pseudo device and one or more of its hdisks to have the same *pvid* attribute in the ODM. If the volume group containing the vpath pseudo device is now activated, the LVM will use the first device it finds in the ODM with the desired *pvid* to activate the volume group.

  As an example, if you issue the **lsvpcfg** command, the following output is displayed:

```
vpath0 (Avail pv vpathvg) 018FA067 = hdisk1 (Avail )
vpath1 (Avail ) 019FA067 = hdisk2 (Avail )
vpath2 (Avail ) 01AFA067 = hdisk3 (Avail )
vpath3 (Avail ) 01BFA067 = hdisk4 (Avail ) hdisk27 (Avail )
vpath4 (Avail ) 01CFA067 = hdisk5 (Avail ) hdisk28 (Avail )
vpath5 (Avail ) 01DFA067 = hdisk6 (Avail ) hdisk29 (Avail )
vpath6 (Avail ) 01EFA067 = hdisk7 (Avail ) hdisk30 (Avail )
vpath7 (Avail ) 01FFA067 = hdisk8 (Avail ) hdisk31 (Avail )
vpath8 (Avail ) 020FA067 = hdisk9 (Avail ) hdisk32 (Avail )
vpath9 (Avail pv vpathvg) 02BFA067 = hdisk20 (Avail ) hdisk44 (Avail )
vpath10 (Avail pv vpathvg) 02CFA067 = hdisk21 (Avail ) hdisk45 (Avail )
vpath11 (Avail pv vpathvg) 02DFA067 = hdisk22 (Avail ) hdisk46 (Avail )
vpath12 (Avail pv vpathvg) 02EFA067 = hdisk23 (Avail ) hdisk47 (Avail )
vpath13 (Avail pv vpathvg) 02FFA067 = hdisk24 (Avail ) hdisk48 (Avail )
```

An example of a **chdev** command that could also set the *pvid* attribute for an hdisk is:

```
chdev -l hdisk46 -a queue_depth=30
```

The output of the **lsvpcfg** command will look similar to this:

```
vpath0 (Avail pv vpathvg) 018FA067 = hdisk1 (Avail )
vpath1 (Avail ) 019FA067 = hdisk2 (Avail )
vpath2 (Avail ) 01AFA067 = hdisk3 (Avail )
vpath3 (Avail ) 01BFA067 = hdisk4 (Avail ) hdisk27 (Avail )
vpath4 (Avail ) 01CFA067 = hdisk5 (Avail ) hdisk28 (Avail )
vpath5 (Avail ) 01DFA067 = hdisk6 (Avail ) hdisk29 (Avail )
vpath6 (Avail ) 01EFA067 = hdisk7 (Avail ) hdisk30 (Avail )
vpath7 (Avail ) 01FFA067 = hdisk8 (Avail ) hdisk31 (Avail )
vpath8 (Avail ) 020FA067 = hdisk9 (Avail ) hdisk32 (Avail )
vpath9 (Avail pv vpathvg) 02BFA067 = hdisk20 (Avail ) hdisk44 (Avail )
vpath10 (Avail pv vpathvg) 02CFA067 = hdisk21 (Avail ) hdisk45 (Avail )
vpath11 (Avail pv vpathvg) 02DFA067 = hdisk22 (Avail ) hdisk46 (Avail pv vpathvg)
vpath12 (Avail pv vpathvg) 02EFA067 = hdisk23 (Avail ) hdisk47 (Avail )
vpath13 (Avail pv vpathvg) 02FFA067 = hdisk24 (Avail ) hdisk48 (Avail )
```

The output of the **lsvpcfg** command will show that vpath11 contains hdisk22 and hdisk46. However, hdisk46 is the only one with the *pv* attribute set. Again using the **lsvg -p vpathvg** command, you might see something like this:

```
vpathvg:
PV_NAME           PV STATE    TOTAL PPs   FREE PPs    FREE DISTRIBUTION
vpath10           active      29          4           00..00..00..00..04
hdisk46           active      29          4           00..00..00..00..04
vpath12           active      29          4           00..00..00..00..04
vpath13           active      29          28          06..05..05..06..06
```

Notice that now device vpath11 has been replaced by hdisk46. That is because hdisk46 is one of the hdisk devices included in vpath11 and it has a *pvid* attribute in the ODM. In this case the LVM has used hdisk46 instead of vpath11 when it activated volume group vpathvg. The volume group is now in a ″mixed mode″ of operation, partially using vpath pseudo devices and partially using hdisk devices. Failover protection is effectively disabled for this physical volume of the vpathvg volume group.

Note: The way to fix this problem is to run the **dpovgfix vg-name** command after running the **chdev** command.

- Running **mksysb restore**

  If a system is restored from a mksysb restore file or tape, the vpath pseudo device *pvid* attribute is not set. All logical volumes made up of vpath pseudo

devices use hdisk devices instead of vpath devices. This can be rectified by using the **hd2vp** shell script to convert the volume group back to using vpath devices.

- Manually deleting devices and running config manager **(cfgmgr)**

  Assume for example that vpath3 is made up of hdisk4 and hdisk27 and that vpath3 is currently a physical volume. If the vpath3, hdisk4, and hdisk27 devices are all deleted by using the **rmdev** command and then **cfgmgr** is invoked at the command line, only one path of the original vpath3 is configured by AIX. A sequence that would produce this situation would be:

  ```
  rmdev -dl vpath3
  rmdev -dl hdisk4
  rmdev -dl hdisk27
  cfgmgr
  ```

  The **datapath query device** command displays the vpath3 configuration status.

  Next, all paths to the vpath must be restored. You can restore the paths in one of the following ways:

  – Run **cfgmgr** once for each installed SCSI or fibre adapter.
  – Run **cfgmgr** *n* times, where *n* represents the number of paths per Subsystem Device Driver device.

    Note: Running the AIX Configuration Manager (**cfgmgr**) *n* times for n-path configurations of ESS devices is not always required. It depends on whether the ESS device has been used as a physical volume of a volume group or not. If it has, it is necessary to run **cfgmgr** *n* times for n-path configuration. Since the ESS device has been used as a physical volume of a volume group before, it has a PVID value written on its sector 0. When the first SCSI or fibre adapter is configured by **cfgmgr**, the AIX disk driver configuration method creates a *pvid* attribute in the AIX ODM database with the *pvid* value it read from the device. It then creates a logical name (hdiskN), and puts the hdiskN in the `Defined` condition. When the second adapter is configured, the AIX disk driver configuration method reads the *pvid* from the same device again, and searches the ODM database to see if there is already a device with the same *pvid* in the ODM. If there is a match, and that hdiskN is in a `Defined` condition, the AIX disk driver configuration method does not create another hdisk logical name for the same device. That is why you only see one set of hdisks getting configured the first time **cfgmgr** runs. When **cfgmgr** runs for second time, the first set of hdisks are in the `Available` condition, so a new set of hdisks are `Defined` and configured to the `Available` condition. This is how the process works. That is why you have to run **cfgmgr** *n* times to get *n* paths configured. If the ESS device has never belonged to a volume group, that means there is no *pvid* written on its sector 0. In that case, you only need to run **cfgmgr** once to get all multiple paths configured. This is how the AIX disk driver configuration method works.

  If you run the **cfgmgr** command instead of rebooting the system after all the ESS hdisk devices are restored, you must unconfigure *all* Subsystem Device Driver devices to the `Defined` condition. Then reconfigure the Subsystem Device Driver devices to the `Available` condition in order to restore all paths to the Subsystem Device Driver (vpath) devices.

  An example of unconfiguring a Subsystem Device Driver device to the `Defined` condition is:

```
rmdev -l vpathN
```

An example of unconfiguring *all* Subsystem Device Driver device to the `Defined` condition is:

```
rmdev -l dpo -R
```

An example of configuring a vpath device to the `Available` condition is:

```
mkdev -l vpathN
```

An example of configuring all vpath device to the `Available` condition is:

```
smitty device
```

- Recovering from mixed volume groups

  The user can run the **dpovgfix** shell script to recover a mixed volume group. The syntax is **dpovgfix vg-name**. The script tries to find a pseudo device corresponding to each hdisk in the volume group and replaces the hdisk with the vpath pseudo device. In order for the shell script to be executed, all mounted file systems of this volume group have to be unmounted. After successful completion of the **dpovgfix** shell script, mount the file systems again.

Extending an existing Subsystem Device Driver volume group

You can extend a volume group with Subsystem Device Driver vpath devices using the Logical Volume Groups, SMIT panel. The Subsystem Device Driver vpath devices to be added to the volume group should be chosen from those that can provide failover protection. It is possible to add an Subsystem Device Driver vpath device to a Subsystem Device Driver volume group that has only a single path (vpath0 in Figure 3 on page 22) and then add paths later by reconfiguring the ESS. With a single path, failover protection is not provided. (See “Adding paths to a Subsystem Device Driver device that is part of a volume group” on page 12 for information on adding paths to a Subsystem Device Driver device.)

To extend a volume group with Subsystem Device Driver devices, follow these steps:

1. Enter **SMIT** from your desktop window. The System Management Interface Tool is displayed.
2. Select **System Storage Management (Physical & Logical Storage)** and press Enter. The System Storage Management (Physical & Logical Storage) panel is displayed.
3. Select **Logical Volume Manager** and press Enter. The Volume Group menu is displayed.
4. Select **Volume Group** and press Enter. The Add Volume Group with Data Path Devices panel is displayed.
5. Select **Add Volume Group with Data Path Devices** and press Enter.
6. Type in the volume group name and physical volume name and press Enter. You can also use the F4 key to list all the available Subsystem Device Driver devices, and select the devices you want to add to the volume group.

If you use a script file to extend an existing Subsystem Device Driver volume group, you must modify your script file and replace the **extendvg** command with the **extendvg4vp** command.

Subsystem Device Driver utility programs

The Subsystem Device Driver provides two conversion scripts, **hd2vp** and **vp2hd**. The **hd2vp** script converts a volume group from original ESS hdisks into Subsystem

Device Driver vpaths, and the **vp2hd** script converts a volume group from Subsystem Device Driver vpaths into ESS hdisks. Use the **vp2hd** program when you want to configure your applications back to original ESS hdisks, or when you want to remove the Subsystem Device Driver from your AIX host system.

Note: You must convert all your applications and volume groups to the original ESS hdisk device special files before removing the Subsystem Device Driver.

The syntax for these conversion programs are:

```
hd2vp  vgname
vp2hd  vgname
```

These two conversion programs require that a volume group contain either *all* original ESS hdisks, or *all* Subsystem Device Driver vpaths. The program fails if a volume group contains both kinds of device special files. To avoid this problem, always use SMIT to create a volume group of Subsystem Device Driver devices.

Note: Performing AIX system management operations on adapters and ESS hdisk devices might cause original ESS hdisks to be contained within a Subsystem Device Driver volume group. This happens when a Subsystem Device Driver volume group is inactivated (varied off), and certain AIX commands to the hdisk put the *pvid* attribute of hdisk back into the ODM database. The following is an example of a command that does this:

```
chdev -l hdiskN -a queue_depth=30
```

There are basically two problem cases to be concerned with:

Case 1: If this disk is an active hdisk of a vpath that belongs to a Subsystem Device Driver volume group, and you run the **varyonvg** command to activate this Subsystem Device Driver volume group, LVM might pick up the hdisk device rather than the vpath device. The result is that a Subsystem Device Driver volume group partially uses Subsystem Device Driver vpath devices, and partially uses ESS hdisk devices.

Case 2: If this problem occurs when you use the **`lsvg -p vgname`** command to check the physical volumes of a volume group, run the **dpovgfix** script tool to fix the problem. The syntax is **dpovgfix vg-name**.

Using ESS devices directly

If your application used ESS hdisk device special files directly before installing the Subsystem Device Driver, follow these steps after you install the Subsystem Device Driver to convert it to using the Subsystem Device Driver vpath device special files.

Note: You can also enter the **lsvpcfg** command directly rather than using SMIT.

1. Type **SMIT** from your desktop window. The System Management Interface Tool is displayed.
2. Select **Devices** and press Enter. The Devices menu is displayed.
3. Select **Data Path Devices** and press Enter. The Data Path Devices panel is displayed.
4. Select **Display Data Path Device Configuration**. The system displays all Subsystem Device Driver vpaths with their attached multiple paths (hdisks).
5. Search the list of hdisks to locate the hdisks your application is using.

6. Replace each hdisk with its corresponding Subsystem Device Driver vpath device.

   Note: Depending upon your application, the manner in which you replace these files is different. If this is a new application, use the Subsystem Device Driver vpath rather than hdisk to use the Subsystem Device Driver load balancing and failover features.

Using ESS devices through AIX LVM

Attention: You must use the SMIT panels for this procedure. Do not use the **mkvg** command directly. Otherwise, the path failover capability could be lost.

If your application accesses ESS devices through the Logical Volume Manager (LVM), determine the volume group that it uses before you convert volume groups. Then, follow these steps to convert the volume group from the original ESS device hdisks to the Subsystem Device Driver vpaths.

1. Determine the file systems or logical volumes that your application accesses.
2. Enter **SMIT** from your desktop window. The System Management Interface Tool is displayed.
3. Select **System Storage Management (Physical & Logical Storage)** and press Enter. The System Storage Management (Physical & Logical Storage) panel is displayed.
4. Select **Logical Volume Manager** and press Enter. The Logical Volume Manager menu is displayed.
5. Select **Logical Volume** and press Enter. The Logical Volume panel is displayed.
6. Select **List All Logical Volumes by Volume Group** to determine the logical volumes that belong to this volume group and their logical volume mount points.
7. Press **Enter**. The logical volumes are listed by volume group.

   To determine file systems, follow these steps:

   a. Enter **SMIT** from your desktop window. The System Management Interface Tool is displayed.
   b. Select **System Storage Management (Physical & Logical Storage)** and press Enter. The System Storage Management (Physical & Logical Storage) panel is displayed.
   c. Select **Logical Volume Manager** and press Enter. The Logical Volume Manager menu is displayed.
   d. Select **File Systems** and press Enter. The File Systems panel is displayed.
   e. Select **List All File Systems** to locate all file systems that have the same mount points as the logical volumes.
   f. Press **Enter**. The file systems are listed.
   g. Note the file system name of that volume group and their mount points, if they are mounted.
   h. Unmount these file systems.
8. Enter the following to convert the volume group from the original ESS hdisks to Subsystem Device Driver vpaths:

   ```
   hd2vp vgname
   ```
9. When conversion completes, mount all file systems that you previously unmounted.

When the conversion completes, your application now accesses ESS physical LUNs through Subsystem Device Driver vpath devices. This provides load balancing and failover protection for your application.

Note: For a new application that will use an existing volume group created with ESS original device hdisks, follow the previous steps to unmount all file systems that belong to this volume group and then convert it to a volume group with the Subsystem Device Driver vpath devices.

Migrating a non-Subsystem Device Driver volume group to an ESS Subsystem Device Driver multipath volume group in concurrent mode

Before you migrate your non-Subsystem Device Driver volume group to a Subsystem Device Driver volume group, there are a number of things that you should check. You need to make sure that the following things have been done:

- The AIX Subsystem Device Driver is installed and configured. To see if Subsystem Device Driver is installed, issue the command:

  ```
  lslpp -l dpo.ibmssd.rte.432
  ```

  An example of output from the **lslpp** command is:

  ```
  Fileset                      Level  State      Description
    ----------------------------------------------------------------------------
  Path: /usr/lib/objrepos
    dpo.ibmssd.rte.432         1.2.0.0  COMMITTED  IBM Subsystem Device Driver
                                                   runtime for AIX V432

  Path: /etc/objrepos
    dpo.ibmssd.rte.432         1.2.0.0  COMMITTED  IBM Subsystem Device Driver
                                                   runtime for AIX V432
  ```

- The ESS subsystem devices you want to migrate to have multiple paths configured per LUN. To check the status of your Subsystem Device Driver configuration, use the System Management Interface Tool (SMIT) or issue the **lsvpcfg** command from the command line. To use SMIT:
  - Type **SMIT** and press Enter from your desktop window. The System Management Interface Tool Menu is displayed.
  - Select **Devices** and press Enter. The Devices menu is displayed.
  - Select **Data Path Device** and press Enter. The Data Path Device panel is displayed.
  - Select **Display Data Path Device Configuration** and press Enter. A list is displayed of the pseudo devices and whether there are multiple paths configured for the devices.
- Make sure the Subsystem Device Driver vpath devices you are going to migrate to do not belong to any other volume group, and that the corresponding physical device (ESS LUN) does not have a *pvid* written on it. Use the **lsvpcfg** command output to check the Subsystem Device Driver vpath devices that you are going to use for migration. Make sure there is no *pv* displayed for this vpath and its paths (hdisks). If a LUN has never belonged to any volume group, then there is no *pvid* written on it. In case there is a *pvid* written on the LUN and the LUN does not belong to any volume group, then the user needs to clear the *pvid* from the LUN, before using it to migrate a volume group. The commands to clear the *pvid* are:

  ```
  chdev -l  hdiskN  -a  pv=clear
  chdev -l  vpathN  -a  pv=clear
  ```

Attention: Exercise care when clearing a *pvid* from a device with this command. Issuing this command to a device, which DOES belong to an existing volume group, can cause system failures.

You should complete the following steps to migrate a non-Subsystem Device Driver volume group to a multipath Subsystem Device Driver volume group in concurrent mode:

1. Add new Subsystem Device Device Driver vpath devices to an existing non-Subsystem Device Device Driver volume group:
   a. Type **SMIT** and press Enter from your desktop window. The System Management Interface Tool Menu is displayed.
   b. Select **System Storage Management (Physical & Logical)** and press Enter. The System Storage Management (Physical & Logical) menu is displayed.
   c. Select **Logical Volume Manager** and press Enter. The Logical Volume Manager panel is displayed.
   d. Select **Volume Group** and press Enter. The Volume Group panel is displayed.
   e. Select **Add a Data Path Volume to a Volume Group** and press Enter.
   f. Type the volume group name and physical volume name and press Enter. You can also use the F4 key to list all the available Subsystem Device Driver vpath devices, and the F7 key to select the physical volumes you want to add.
2. Mirror logical volumes from the original volume to a Subsystem Device Driver ESS volume. Use the command:

   ```
   smitty mklvcopy
   ```

   Use the new Subsystem Device Driver vpath devices for copying all logical volumes. Do not forget to include JFS log volumes.

   Note: The command **smitty mklvcopy** copies one logical volume at a time. A fast-path command to mirror *all* the logical volumes on a volume group is **mirrorvg**.
3. Synchronize logical volumes (LVs) or force synchronization. Use the following command to synchronize all the volumes:

   ```
   smitty syncvg
   ```

   There are two choices on the smitty menu:
   a. Synchronize by Logical Volume
   b. Synchronize by Physical Volume

   The fast way to synchronize logical volumes is to use the **Synchronize by Physical Volume** option.
4. Break the mirror and delete the original LVs. Use the following command to remove the original copy of the logical volumes from all original non-Subsystem Device Driver physical volumes:

   ```
   smitty  rmlvcopy
   ```
5. Remove the original non-Subsystem Device Driver devices from the volume group. Use the command:

   ```
   smitty reducevg
   ```

   The Remove a Physical Volume panel is displayed. Remove all non-Subsystem Device Driver devices.

Notes:

1. A non-Subsystem Device Driver volume group can consist of non-ESS or ESS hdisk devices.
2. There is no failover protection without multiple paths configured per LUN.

Example of migrating an existing non-Subsystem Device Driver volume group to Subsystem Device Driver vpath devices in concurrent mode

This procedure shows how to migrate an existing AIX volume group to the use of Subsystem Device Driver vpath (pseudo) devices, having multipath capability, without taking the volume group out of service. In this example we start with a volume group vg1, made up of one ESS device, hdisk13.

In order to perform the migration, you must have vpath devices available that are greater than or equal to the size of each of the hdisks making up the volume group. In this example, we have a pseudo device, vpath12, with two paths hdisk14 and hdisk30, that we will migrate the volume group to.

1. Add the vpath device to the volume group as an `Available` volume:
   a. Type **SMIT** and press Enter from your desktop window. The System Management Interface Tool Menu is displayed.
   b. Select **System Storage Management (Physical & Logical)** and press Enter. The System Storage Management (Physical & Logical) menu is displayed.
   c. Select **Logical Volume Manager** and press Enter. The Logical Volume Manager panel is displayed.
   d. Select **Volume Group** and press Enter. The Volume Group panel is displayed.
   e. Select **Add a Data Path Volume to a Volume Group** and press Enter.
   f. Type **vg1** in the **Volume Group Name** field. Type **vpath12** in the **Physical Volume Name** field. Press Enter.

      You can also enter the command:

      ```
      extendvg4vp -f vg1 vpath12
      ```
2. Mirror logical volumes from the original volume to the new Subsystem Device Driver vpath volume:
   a. Type **SMIT** and press Enter from your desktop window. The System Management Interface Tool Menu is displayed.
   b. Select **System Storage Management (Physical & Logical)** and press Enter. The System Storage Management (Physical & Logical) menu is displayed.
   c. Select **Logical Volume Manager** and press Enter. The Logical Volume Manager panel is displayed.
   d. Select **Volume Group** and press Enter. The Volume Group panel is displayed.
   e. Select **Mirror a Volume Group** and press Enter. The Mirror a Volume Group panel is displayed.
   f. Type a volume group name. Type a physical volume name. Press Enter.

      You can also enter the command:

      ```
      mirrorvg vg1 vpath12
      ```
3. Synchronize the logical volumes in the volume group:

a. Type **SMIT** and press Enter from your desktop window. The System Management Interface Tool Menu is displayed.
b. Select **System Storage Management (Physical & Logical)** and press Enter. The System Storage Management (Physical & Logical) menu is displayed.
c. Select **Logical Volume Manager** and press Enter. The Logical Volume Manager panel is displayed.
d. Select **Volume Group** and press Enter. The Volume Group panel is displayed.
e. Select **Synchronize LVM Mirrors** and press Enter. The Synchronize LVM Mirrors panel is displayed.
f. Select **Synchronize by Physical Volume**.

You can also enter the command:

```
syncvg -p hdisk13 vpath12
```

4. Delete copies from the original physical volume:
   a. Type **SMIT** and press Enter from your desktop window. The System Management Interface Tool Menu is displayed.
   b. Select **Logical Volumes** and press Enter. The Logical Volumes menu is displayed.
   c. Select **Set Characteristic of a Logical Volume** and press Enter. The Set Characteristic of a Logical Volume panel is displayed.
   d. Select **Remove Copy from a Logical Volume** and press Enter. The Remove Copy from a Logical Volume panel is displayed.

   You can also enter the command:

   ```
   rmlvcopy loglv01 1 hdisk13 or rmlvcopy lv01 1 hdisk13
   ```

5. Remove the old physical volume from the volume group:
   a. Type **SMIT** and press Enter from your desktop window. The System Management Interface Tool Menu is displayed.
   b. Select **Logical Volume manager** and press Enter. The Logical Volume manager menu is displayed.
   c. Select **Volume Groups** and press Enter. The Volume Groups panel is displayed.
   d. Select **Set Characteristics of a Volume Group** and press Enter. The Set Characteristics of a Volume Group panel is displayed.
   e. Select **Remove a Physical Volume from a Volume Group** and press Enter. The Remove a Physical Volume from a Volume Group panel is displayed.

   You can also enter the command:

   ```
   reducevg vg1 hdisk13
   ```

Using the Subsystem Device Driver AIX trace

The Subsystem Device Driver supports AIX trace functions. The trace ID for the Subsystem Device Driver is **0x2F8**. Trace ID **0x2F8** traces routine entry, exit, and error paths of the algorithm. To use it, manually turn on the trace function before the program starts to run, then turn off the trace function either after the program stops, or any time you need to read the trace report. To start the trace function, enter:

```
trace -a -j 0x2F8
```

To stop the trace function, enter:

```
trcstop
```

To read the report, enter:

```
trcrpt | pg
```

Note: To perform the AIX trace function, you must have the **bos.sysmgt.trace** fileset installed on your system.

Subsystem Device Driver error log messages

The Subsystem Device Driver logs error conditions into the AIX errlog system. To check if the Subsystem Device Driver generated an errlog message, enter this command:

```
errpt -a | grep VPATH
```

The following are the Subsystem Device Driver error log messages:

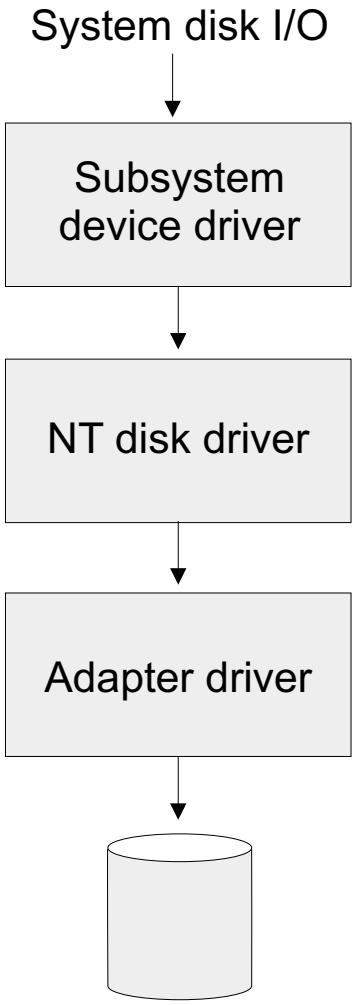
VPATH_XBUF_NOMEM
: Allocate kernel-pinned memory failed. When trying to open a Subsystem Device Driver vpath file, it tries to allocate kernel-pinned memory for later I/O transfer. If the system returns a null pointer to the caller, kernel-pinned memory is not available. The device File Open fails.

VPATH_DEVICE_OPEN
: The Subsystem Device Driver device file fails to open one of its paths (hdisks). A vpath device open is successful if at least one attached path opens. If all vpath device paths fail to open, then the vpath device open fails.

VPATH_DEVICE_OFFLINE
: A path of the Subsystem Device Driver vpath fails the I/O request after several attempts at retry, and its path status is set to **DD_PATH_DEAD**. This path is taken offline. Use the datapath console command program to set the offline path to online. For more information, see “Chapter 8. Using commands” on page 85.

VPATH_DEVICE_ONLINE
: The Subsystem Device Driver supports Dead path auto_failback and Dead path reclamation. A Dead path will be selected to send an I/O after 2000 I/O requests bypassed. If the I/O is successful, the Dead path is put Online, and its mode is put back to Normal; A Dead path is put Online, and its mode changes to Normal after it bypasses 50000 I/O requests on a good path.

Chapter 4. Installing and configuring IBM Subsystem Device Driver on a Windows NT host



S008997Q

Figure 4. Where the Subsystem Device Driver fits in the protocol stack

Notes:

1. If you attempt to install Subsystem Device Driver *over* an existing version of Subsystem Device Driver or Data Path Optimizer (DPO), the installation fails. You must uninstall any previous version of Subsystem Device Driver or DPO before installing this version of the Subsystem Device Driver.
2. You must have Windows NT 4.0 Service Pack 3 or higher installed on your system.
3. For updated and additional information not included in this manual, see the README file on the CDR or visit the Subsystem Device Driver Web site www.ibm.com/storage/support/techsup/swtechsup.nsf/support/sddupdates

4. You cannot run the Subsystem Device Driver in a non-concurrent environment in which more than one host is attached to the same LUN on a Enterprise Storage Server; for example, in a multihost environment. This restriction includes clustered hosts, such as servers running Windows NT clustering. However, concurrent multihost environments are supported.
5. The Subsystem Device Driver supports 32-bit mode applications.

This chapter provides instructions to install and set up the IBM Subsystem Device Driver on a Windows NT host system attached to an ESS.

Hardware and software requirements

To install the Subsystem Device Driver on your Windows NT host system, you must meet the following minimum hardware and software requirements:

- Configure the host system to the ESS as an IBM or Compaq server with Windows NT 4.0 or higher.
- The Windows NT host system should be an Intel-based system with Windows NT Version 4.0 Service Pack 3 or higher installed. The host system can be a uni-processor or a multi-processor system. NT clustering is not supported.
- To install the Subsystem Device Driver and use the input-output load balancing and failover features, you need a minimum of two SCSI or fibre-channel adapters. The maximum number of SCSI adapters that are supported is 32; the maximum number of fibre-channel adapters that are supported is 256.
- The Subsystem Device Driver also supports one SCSI adapter on the host system. With single-path access, concurrent download of licensed internal code is supported. However, the load balancing and failover features are not available.

  Note: A host system with a single-path fibre connection does not support concurrent firmware download.
- For information about the SCSI adapters or fibre-channel adapters that can attach to your Windows NT host system go to http://www.storage.ibm.com/hardsoft/products/ess/supserver.htm
- You need a SCSI cable to connect each SCSI host adapter to an ESS port.
- You need a fiber-optic cable to connect each fibre-channel adapter to a ESS port.

Configuring the ESS

Before you install the Subsystem Device Driver, configure your ESS for single-port or multiple-port access for each LUN. The Subsystem Device Driver requires a minimum of two independent paths that share the same LUN to use the load balancing and failover features.

It is strongly recommended that the paths be on different clusters to achieve maximum availability.

For information about configuring your ESS, see *IBM Enterprise Storage Server Introduction and Planning Guide*, GC26–7294.

Notes:

1. A host server with a single-path fibre connection to an ESS is not supported.
2. A host server with SCSI connections and a single-path fibre connection to an ESS is not supported.

Configuring SCSI adapters for the Subsystem Device Driver on Windows NT host systems

Attention: Failure to disable the BIOS of attached non-boot devices may cause your system to boot from an unexpected non-boot device.

Before you install and use the Subsystem Device Driver, you must configure your SCSI adapters. For SCSI adapters that attach boot devices, ensure that the BIOS for the adapter is *enabled*. For all other adapters that attach non-boot devices, ensure the BIOS for the adapter is *disabled*.

Note: When the adapter shares the SCSI bus with other adapters, the BIOS must be *disabled*.

Configuring fibre-channel adapters for the Subsystem Device Driver on Windows NT host systems

Unlike SCSI adapters, there are no special configuration requirements for fibre-channel adapters attached to Windows NT host systems.

Installing the Subsystem Device Driver on a Windows NT host system

Note: You must have root access and you must log on as an administrator user to install the Subsystem Device Driver.

To install all components, you must have 1 MB (MB equals approximately 1 000 000 bytes) of disk space available, and you must have Windows NT 4.0 Service Pack 3 or higher installed on your system.

Perform the following steps to install the Subsystem Device Driver filter and application programs on your system:

1. Log on as the administrator user.
2. Insert the Subsystem Device Driver installation CDR into the CD-ROM drive.
3. Start the Windows NT Explorer program.
4. Select the CD-ROM drive. A list of all the installed directories on the CDR is displayed.
5. Select the /winnt directory.
6. Run the **setup.exe** program. This starts the Installshield.
7. Click **Next**. The Software License agreement is displayed.
8. Click **Yes**. The User Information panel is displayed.
9. Type your name and your company name.
10. Click **Next**. The Choose Destination Location panel is displayed.
11. Click **Next**. The Setup panel is displayed.
12. Select the type of setup you prefer from the three setup choices described below. IBM recommends that you select **Typical**.

    Typical
    Selects all options.

    Compact
    Selects the minimum required options *only* (installation driver and readme file).

Custom
Select the options that you need.

13. Click **Next**. The Setup Complete panel is displayed.
14. Click **Finish**. The Subsystem Device Driver program prompts you to start your computer again.
15. Click **Yes** to start your computer again. When you log on again, you see a **Subsystem Device Driver Management** entry in your Program menu containing the following files:
    a. Data path management
    b. Subsystem Device Driver manual
    c. README file

Note: You can verify that the Subsystem Device Driver has been successfully installed by issuing the **datapath query device** command. If the command executes, the Subsystem Device Driver is installed.

Configuring the Subsystem Device Driver for a Windows NT host system

Note: You must log on as an administrator user to have access to the Windows NT disk administrator.

To activate the Subsystem Device Driver you need to restart your Windows NT system after it is installed. In fact, a restart is required to activate multipath support whenever a new file system or partition is added.

Adding paths to Subsystem Device Driver devices

Attention: Ensure that the Subsystem Device Driver is installed *before* you add additional paths to a device. Otherwise, the Windows NT server's ability to access existing data on that device could be lost.

Perform the following steps to activate additional paths to a vpath device:

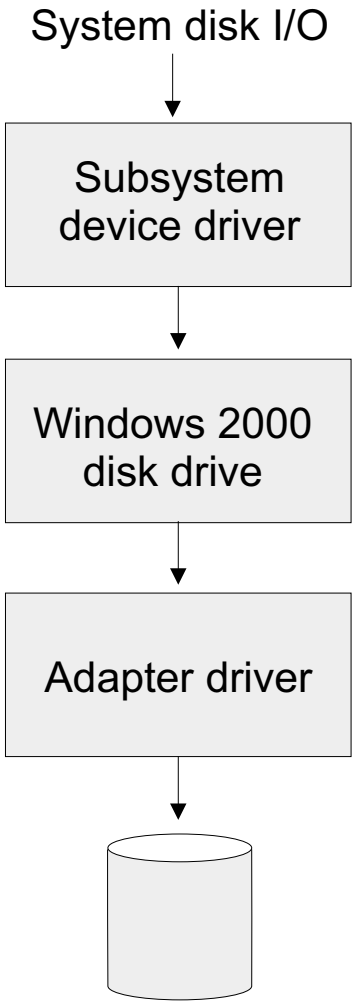
1. Install any additional hardware on the Windows NT server or ESS.
2. Restart the Windows NT server.
3. Verify the path is added correctly.

Verifying the Subsystem Device Driver configuration

Perform the following steps to verify the configuration of the Subsystem Device Driver on a Windows NT host system:

1. Log on as the administrator user.
2. Click **Start—> Program—> Administrative Tools—> Disk Administrator**. The Disk Administrator panel is displayed.
3. Verify that the number of devices displayed as online equals the actual number of devices.
4. Verify that the additional paths are offline.

Chapter 5. Installing and configuring the IBM Subsystem Device Driver on a Windows 2000 host



S009024

Figure 5. Where the Subsystem Device Driver fits in the protocol stack

Notes:

1. For updated and additional information not included in this manual, see the README file on the (Compact Disk Recordable) CDR or visit the Subsystem Device Driver Web site www.ibm.com/storage/support/techsup/swtechsup.nsf/support/sddupdates
2. You cannot run the Subsystem Device Driver in a non-concurrent environment in which more than one host is attached to the same logical unit number (LUN) on a Enterprise Storage Server; for example, in a multi-host environment. This restriction includes clustered hosts, such as servers running Windows NT clustering. However, concurrent multi-host environments are supported.
3. The Subsystem Device Driver supports 32-bit mode applications.

This chapter provides instructions to install and set up the Subsystem Device Driver on a Windows 2000 host system attached to an ESS.

Hardware and software requirements

To be able to install the Subsystem Device Driver on your Windows 2000 host system, you must meet the following minimum hardware and software requirements:

- Configure the host system to the ESS as an IBM or Compaq server with Windows NT 4.0 or higher.
- The Windows 2000 host system should be an Intel-based system. The host system can be a uni-processor or a multi-processor system. NT clustering is not supported.
- To install the Subsystem Device Driver and use the input-output load balancing and failover features, you need a minimum of two SCSI or fibre-channel adapters. The maximum number of SCSI adapters that are supported is 32; the maximum number of fibre-channel adapters that are supported is 256.
- The Subsystem Device Driver also supports one SCSI adapter on the host system. With single-path access, concurrent download of licensed internal code is supported. However, the load balancing and failover features are not available.

  Notes:

  1. A host server with a single-path fibre connection to an ESS is not supported.
  2. A host server with SCSI connections and a single-path fibre connection to an ESS is not supported.
- For information about the SCSI adapters or fibre-channel adapters that can attach to your Windows 2000 host system go to http://www.storage.ibm.com/hardsoft/products/ess/supserver.htm
- You need a SCSI cable to connect each SCSI host adapter to a storage system controller port.
- You need a fiber-optic cable to connect each fibre-channel adapter to an ESS port.

Configuring the ESS

Before you install the Subsystem Device Driver, configure your ESS for single-port or multiple-port access for each LUN. The Subsystem Device Driver requires a minimum of two independent paths that share the same logical unit to use the load balancing and failover features.

It is strongly recommended that the paths be on different clusters to achieve maximum availability.

For information about configuring your ESS, see *IBM Enterprise Storage Server Introduction and Planning Guide*, GC26–7294.

Note: During heavy usage, the Windows 2000 operating system might slow down while trying to recover from error conditions.

Configuring SCSI adapters for the Subsystem Device Driver on Windows 2000 host systems

Before you install and use the Subsystem Device Driver, you must configure your SCSI adapters. For SCSI adapters that attach boot devices, ensure that the BIOS for the adapter is *enabled*. For all other adapters that attach non-boot devices, ensure the BIOS for the adapter is *disabled*.

Note: When the adapter shares the SCSI bus with other adapters, the BIOS must be *disabled*.

Configuring fibre-channel adapters for the Subsystem Device Driver on Windows 2000 host systems

Unlike SCSI adapters, there are no special configuration requirements for fibre-channel adapters attached to Windows 2000 host systems.

Installing the Subsystem Device Driver on a Windows 2000 host system

Note: You must have root access and you must log on as an administrator user to install the Subsystem Device Driver.

To install all components, you must have 1 MB (MB equals approximately 1 000 000 bytes) of disk space available.

Perform the following steps to install the Subsystem Device Driver filter and application programs on your system:

1. Log on as the administrator user.
2. Insert the Subsystem Device Driver installation CD-ROM into the selected drive. The Subsystem Device Driver panel is displayed.
3. Start the Windows 2000 Explorer program.
4. Select the CD-ROM drive. A list of all the installed directories on the CDR is displayed.
5. Select the /win2k directory.
6. Run the **setup.exe** program. This starts the Installshield.
7. Click **Next**. The Software Licensing Agreement panel is displayed.
8. Click **Yes**. The User Information panel is displayed.
9. Type your name and your company name.
10. Click **Next**. The Choose Destination Location panel is displayed.
11. Click **Next**. The Setup panel is displayed.
12. Select the type of setup you prefer from the three setup choices described below. IBM recommends that you select **Typical**.
    Typical
    Selects all options.
    Compact
    Selects the minimum required options *only* (installation driver and readme file).
    Custom
    Select the options that you need.
13. Click **Next**. The Setup Complete panel is displayed.

14. Click **Finish**. The Subsystem Device Driver program prompts you to start your computer again.
15. Click **Yes** to start your computer again. When you log on again, you see a **Subsystem Device Driver** entry in your Program menu containing the following files:
    a. Subsystem Device Driver management
    b. Subsystem Device Driver manual
    c. README file

Note: You can verify that the Subsystem Device Driver has been successfully installed by issuing the **datapath query device** command. If the command executes, the Subsystem Device Driver is installed.

Configuring the Subsystem Device Driver for a Windows 2000 host system

Note: You must log on as an administrator user to have access to the Windows 2000 disk administrator.

To activate the Subsystem Device Driver you need to restart your Windows 2000 system after it is installed. In fact, a restart is required to activate multipath support whenever a new file system or partition is added.

Adding paths to Subsystem Device Driver devices

Attention: Ensure that the Subsystem Device Driver is installed *before* you add additional paths to a device. Otherwise, the Windows 2000 server's ability to access existing data on that device could be lost.

Before adding any additional hardware, you should review the configuration information for the adapters and devices currently on your Windows 2000 server. Perform the following steps to display information about the adapters and devices:

1. Click **Start—> Program—> Subsystem Device Driver—> Subsystem Device Driver Management**. An MS-DOS window is displayed.
2. Type **datapath query adapter** and press Enter. The output should include information about all the installed adapters. In this example, one SCSI adapter is installed on the Windows 2000 host server. The following output is displayed:

```
Active Adapters :1

Adpt#     Adapter Name   State     Mode     Select     Errors  Paths  Active
    0  Scsi Port1 Bus0  NORMAL   ACTIVE       4057          0      8       8
```

3. Next, type **datapath query device** and press Enter. In this example, 8 devices are attached to the SCSI path. The following output is displayed:

```
Total Devices : 8

DEV#:   0  DEVICE NAME: Disk7 Part7  TYPE: 2105E20   SERIAL: 01312028
===========================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk7 Part0     OPEN   NORMAL       1045          0

DEV#:   1  DEVICE NAME: Disk6 Part6  TYPE: 2105E20   SERIAL: 01212028
===========================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk6 Part0     OPEN   NORMAL        391          0

DEV#:   2  DEVICE NAME: Disk5 Part5  TYPE: 2105E20   SERIAL: 01112028
===========================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk5 Part0     OPEN   NORMAL       1121          0

DEV#:   3  DEVICE NAME: Disk4 Part4  TYPE: 2105E20   SERIAL: 01012028
===========================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk4 Part0     OPEN   NORMAL        332          0

DEV#:   4  DEVICE NAME: Disk3 Part3  TYPE: 2105E20   SERIAL: 00F12028
===========================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk3 Part0     OPEN   NORMAL        375          0

DEV#:   5  DEVICE NAME: Disk2 Part2  TYPE: 2105E20   SERIAL: 31412028
===========================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk2 Part0     OPEN   NORMAL        258          0

DEV#:   6  DEVICE NAME: Disk1 Part1  TYPE: 2105E20   SERIAL: 31312028
===========================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk1 Part0     OPEN   NORMAL        267          0

DEV#:   7  DEVICE NAME: Disk0 Part0  TYPE: 2105E20   SERIAL: 31212028
===========================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk0 Part0     OPEN   NORMAL        268          0
```

Perform the following steps to activate additional paths to a vpath device:

1. Install any additional hardware on the Windows 2000 server or the ESS.
2. Restart the Windows 2000 server.
3. Verify that the path is added correctly. See “Verifying additional paths to Subsystem Device Driver devices are correctly installed”.

Verifying additional paths to Subsystem Device Driver devices are correctly installed

After installing additional paths to Subsystem Device Driver devices, you should verify that the additional paths have been installed correctly.

Perform the following steps to verify that the additional paths have been installed correctly:

1. Click **Start—> Program—> Subsystem Device Driver—> Subsystem Device Driver Management**. An MS-DOS window appears.
2. Type **datapath query adapter** and press Enter. The output should include information about any additional adapters that were installed. In this example, an additional SCSI adapter has been installed. The following output is displayed:

```
Active Adapters :2

Adpt#     Adapter Name   State     Mode     Select     Errors  Paths  Active
    0  Scsi Port1 Bus0  NORMAL   ACTIVE       1325          0      8       8
    1  Scsi Port2 Bus0  NORMAL   ACTIVE       1312          0      8       8
```

3. Next, type **datapath query device** and press Enter. The output should include information about any additional devices that were installed. In this example, the output includes information about the new SCSI adapter and the new device numbers that were assigned. The following output is displayed:

```
Total Devices : 8

DEV#:   0  DEVICE NAME: Disk7 Part7  TYPE: 2105E20   SERIAL: 01312028
============================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk7 Part0     OPEN   NORMAL        190          0
    1   Scsi Port2 Bus0/Disk15 Part0     OPEN   NORMAL        179          0

DEV#:   1  DEVICE NAME: Disk6 Part6  TYPE: 2105E20   SERIAL: 01212028
============================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk6 Part0     OPEN   NORMAL        179          0
    1   Scsi Port2 Bus0/Disk14 Part0     OPEN   NORMAL        184          0

DEV#:   2  DEVICE NAME: Disk5 Part5  TYPE: 2105E20   SERIAL: 01112028
============================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk5 Part0     OPEN   NORMAL        194          0
    1   Scsi Port2 Bus0/Disk13 Part0     OPEN   NORMAL        179          0

DEV#:   3  DEVICE NAME: Disk4 Part4  TYPE: 2105E20   SERIAL: 01012028
============================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk4 Part0     OPEN   NORMAL        187          0
    1   Scsi Port2 Bus0/Disk12 Part0     OPEN   NORMAL        173          0

DEV#:   4  DEVICE NAME: Disk3 Part3  TYPE: 2105E20   SERIAL: 00F12028
============================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk3 Part0     OPEN   NORMAL        215          0
    1   Scsi Port2 Bus0/Disk11 Part0     OPEN   NORMAL        216          0

DEV#:   5  DEVICE NAME: Disk2 Part2  TYPE: 2105E20   SERIAL: 31412028
============================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk2 Part0     OPEN   NORMAL        115          0
    1   Scsi Port2 Bus0/Disk10 Part0     OPEN   NORMAL        130          0

DEV#:   6  DEVICE NAME: Disk1 Part1  TYPE: 2105E20   SERIAL: 31312028
============================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk1 Part0     OPEN   NORMAL        122          0
    1    Scsi Port2 Bus0/Disk9 Part0     OPEN   NORMAL        123          0

DEV#:   7  DEVICE NAME: Disk0 Part0  TYPE: 2105E20   SERIAL: 31212028
============================================================================
Path#              Adapter/Hard Disk    State     Mode     Select     Errors
    0    Scsi Port1 Bus0/Disk0 Part0     OPEN   NORMAL        123          0
    1    Scsi Port2 Bus0/Disk8 Part0     OPEN   NORMAL        128          0
```

Chapter 6. Installing and configuring the IBM Subsystem Device Driver on an HP host

Note: For updated and additional information not included in this manual, please see the README file on the CDR or go to the Subsystem Device Driver Web site
www.ibm.com/storage/support/techsup/swtechsup.nsf/support/sddupdates

This chapter provides instructions to install and set up the Subsystem Device Driver on an HP host system attached to an ESS.

Understanding how Subsystem Device Driver works on an HP host

Subsystem Device Driver failover system

The Subsystem Device Driver failover system is designed to provide recovery upon failure of a data path. If this situation occurs, the failover system selects an alternate data path, while minimizing any disruptions in operation. This failover process consists of:

- Detecting a failure
- Signaling to the HP host that a failure has occurred
- Compensating for the failure by selecting an alternate data path.

The Subsystem Device Driver dynamically selects an alternate I/O data path when a software or hardware problem is detected.

Subsystem Device Driver

Note: The Subsystem Device Driver supports 64-bit mode applications.
The Subsystem Device Driver resides above the HP SCSI disk driver (sdisk) in the protocol stack. There can be a maximum of eight sdisk devices underneath each Subsystem Device Driver device in the protocol stack. Each sdisk device represents a different path to the physical device. There can be up to eight sdisk devices that represent up to eight different paths to the physical device.

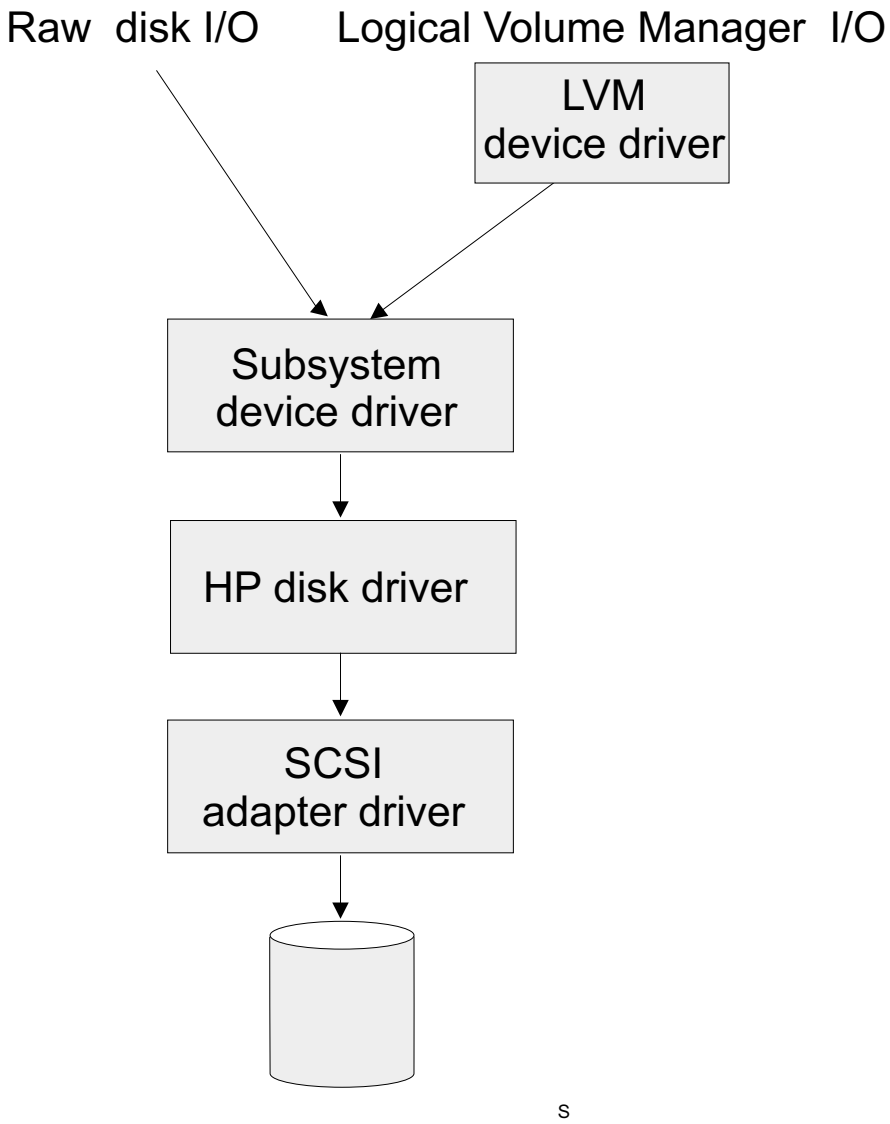


Figure 6. Where the Subsystem Device Driver fits in the protocol stack

Subsystem Device Driver devices behave exactly like sdisk devices. Any operation on an sdisk device can be performed on the Subsystem Device Driver device, including commands such as **mount**, **open**, **close**, **umount**, **dd**, **newfs**, or **fsck**. For example, with Subsystem Device Driver the user types `mount /dev/dsk/vpath0 /mnt1` instead of the HP-UX `mount /dev/dsk/clt2d0 /mnt1` command .

The Subsystem Device Driver acts as a pass-through agent. I/Os sent to the Subsystem Device Driver are passed to an sdisk driver after path selection. When an active path experiences a failure (such as a cable or controller failure), the Subsystem Device Driver dynamically switches to another path. The device driver dynamically balances the load, based on the workload of the adapter.

The Subsystem Device Driver also supports one SCSI adapter on the host system. With single-path access, concurrent download of licensed internal code is supported. However, the load balancing and failover features are not available.

Hardware and software requirements

To be able to install the Subsystem Device Driver on your HP host system, you must meet the following minimum hardware and software requirements:

- A PA-RISC system running HP-UX 11.00
- A multiport storage subsystem, such as ESS
- At least one SCSI host adapter (two are required for load balancing and failover)
- Subsystem LUNs which have been created and confirmed for multiport access
- You need a SCSI cable to connect each SCSI host adapter to a storage system controller port.

Configuring the ESS

Before you install the Subsystem Device Driver, configure your ESS for single-port or multiple-port access for each LUN. The Subsystem Device Driver requires that you provide a minimum of two independent paths that share the same logical unit to use the load balancing and failover features.

It is strongly recommended that the paths be on different ESS clusters to achieve maximum availability.

For information about configuring your ESS, see *IBM Enterprise Storage Server Introduction and Planning Guide*, GC26–7294.

Planning for installation

Before you install Subsystem Device Driver on your HP host, you need to understand what kind of software runs on your host. The way you install Subsystem Device Driver depends on the kind of software you have running. Basically there are two types of software that talk directly to raw or block disk device interfaces such as sdisk and the Subsystem Device Driver:

- UNIX file systems, where there is no logical volume manager present
- Major software applications, such as certain database managers (DBMSs).

There are three possible scenarios for installing Subsystem Device Driver. The scenario you choose depends on the kind of software you have installed:

Scenario 1
Your system has no software applications (other than UNIX) or DBMSs that talk directly to the HP-UX disk device layer

Scenario 2
Your system already has a software application or DBMS, such as Oracle, that talks directly with the HP-UX disk device layer

Scenario 3
Your system already has Subsystem Device Driver and you want to upgrade the software

The following table further describes the various installation scenarios and how you should proceed.

Table 4. Subsystem Device Driver installation scenarios

Installation Scenario	Description	How To Proceed
Scenario 1	• Subsystem Device Driver not installed • No software application or DBMS that talks directly to sdisk interface	Go to: 1. "Installing the Subsystem Device Driver" 2. "Standard UNIX applications" on page 53
Scenario 2	• Subsystem Device Driver not installed • Existing application package or DBMS that talks directly to sdisk interface	Go to: 1. "Installing the Subsystem Device Driver" 2. "Using applications with Subsystem Device Driver" on page 53
Scenario 3	Subsystem Device Driver installed	Go to "Upgrading the Subsystem Device Driver" on page 53

Installing the Subsystem Device Driver

You need to complete the following procedure if you are installing Subsystem Device Driver for the first time on your HP host:

1. Make sure the Subsystem Device Driver (CDR) is available.
2. Insert the CDR into your CD-ROM drive.
3. Mount the CD-ROM drive using the **mount** command. Here is an example of the **mount** command:

   ```
   mount dev/dsk/c0t2d0 /cdrom
   ```

4. Run **sam**

   ```
   > sam
   ```

5. Select **Software Management**.
6. Select **Install Software to Local Host**.
7. At this point, the **SD Install - Software Selection** panel is displayed. Almost immediately afterwards, a Specify Source menu is displayed:
   - For **Source Depot Type**, use the default.
   - For **Source Depot Path**, choose the directory and the IBMdpo.depot file (for example, /cdrom/hp-ux/IBMdpo.depot).
   - Click **OK**.
8. At this point, the following is displayed on the SD Install - Software Selection window:

   ```
   Name           Revision          Information                        Size(Kb)
   IBMdpo_tag ->  1.01              IBMdpo Software for HP-UX 11.xx/   64 ....
   ```

   a. Choose the **IBMdpo_tag** product.
   b. Select **Actions** from the Bar menu, then select **Mark for Install**.
   c. Select **Actions** from the Bar menu, then select **Install (analysis)**. You will see an Install Analysis window, and on it you will see the status of **Ready**.
   d. Select **OK** to proceed. A Confirmation window is displayed which states that the installation will begin.
   e. Type `Yes` and press Enter. The analysis phase starts.

f. After the analysis phase has finished, another Confirmation window is displayed informing you that the system will be restarted after installation is complete. Type `Yes` and press Enter. The installation of IBMdpo will now proceed.

g. Next, an Install window is displayed which informs you about the progress of the IBMdpo software installation. This is what the window looks like:

```
Press 'Product Summary' and/or 'Logfile' for more target information.
Target          : XXXXX
Status          : Building kernel
Percent Complete     : 17%
Kbytes Installed   :  276 of 1393
Time Left (minutes) : 1
Product Summary       Logfile
Done                              Help
```

While the installation process is going on, the **Done** option is not available for you to select. Once the installation of the software has completed, the option is available for you to select.

9. Click **Done**. A Note window is displayed informing you that the local system will restart with the newly installed software.
10. Select **OK** to proceed. The following message is displayed on the machine console before it restarts:

```
* A reboot of this system is being invoked. Please wait.

*** FINAL System shutdown message (XXXXX) ***
System going down IMMEDIATELY
```

Note: You can verify that the Subsystem Device Driver has been successfully installed by issuing the **datapath query device** command. If the command executes, the Subsystem Device Driver is installed.

Post-installation

After Subsystem Device Driver is installed, the device driver resides above the HP SCSI disk driver (sdisk) in the protocol stack. In other words, Subsystem Device Driver now talks to the HP-UX device layer. The Subsystem Device Driver software installation procedure installs a number of Subsystem Device Driver components and updates some system files. Those components and files are listed in the following tables:

Table 5. Subsystem Device Driver components installed

File	Location	Description
libvpath.a	/usr/conf/lib	Subsystem Device Driver device driver
vpath	/usr/conf/master.d	Subsystem Device Driver configuration file
Executables	/opt/IBMdpo/bin	Configuration and status tools
README.sd	/opt/IBMdpo	README file
defvpath	/sbin	Subsystem Device Driver configuration file used during startup

Table 6. System files updated

File	Location	Description
system	/stand/build	Forces the loading of the Subsystem Device Driver device driver
lvmrc	/etc	Causes **defvpath** to run at boot time

If you are not using a DBMS or an application package that talks directly to the sdisk interface, then the installation procedure is nearly complete. However, you still need to customize HP-UX so that standard UNIX applications can use Subsystem Device Driver. Go to section “Standard UNIX applications” on page 53. If you have a DBMS or an application package installed that talks directly to the sdisk interface, such as Oracle, go to “Using applications with Subsystem Device Driver” on page 53 and read the information specific to the application you will be using.

Note: During the installation process, the following files were copied from the IBMdpo_depot to the system:

 # Kernel-related files

- /usr/conf/lib/libvpath.a
- /usr/conf/master.d/vpath

 # Subsystem Device Driver driver related files

- /opt/IBMdpo
- /opt/IBMdpo/bin
- /opt/IBMdpo/README.sd
- /opt/IBMdpo/bin/cfgvpath
- /opt/IBMdpo/bin/datapath
- /opt/IBMdpo/bin/defvpath
- /opt/IBMdpo/bin/libvpath.a
- /opt/IBMdpo/bin/pathtest
- /opt/IBMdpo/bin/showvpath
- /opt/IBMdpo/bin/vpath
- /sbin/defvpath

In addition, the /stand/vmunix kernel was created with the device driver. The /stand/system directory was modified in order to add the device driver entry into the file. After these files were created, the /opt/IBMdpo/bin/cfgvpath program was initiated in order to create vpaths in the /dev/dsk and /dev/rdsk directories for all IBM disks which are available on the system. This information is stored in the /etc/ibmvpath file for use after rebooting the machine.

Note: Subsystem Device Driver devices are found in /dev/rdsk and /dev/dsk. The device is named according to the Subsystem Device Driver number. A device with a number of 0 would be /dev/rdsk/vpath0.

Upgrading the Subsystem Device Driver

Upgrading the Subsystem Device Driver consists of uninstalling and reinstalling the IBMdpo package. If you are upgrading Subsystem Device Driver, go to “Uninstalling Subsystem Device Driver” on page 62 and then go to “Installing the Subsystem Device Driver” on page 50.

Using applications with Subsystem Device Driver

If your system already has a software application or an DBMS installed that communicates directly with the HP-UX disk device drivers, you need to insert the new Subsystem Device Driver device layer between the software application and the HP-UX disk device layer. You also need to customize the software application in order to have it communicate with the Subsystem Device Driver devices instead of the HP-UX devices.

In addition, many software applications and DBMSs need to control certain device attributes such as ownership and permissions. Therefore, you must ensure that the new Subsystem Device Driver devices that these software applications or DBMSs access in the future have the same attributes as the HP-UX sdisk devices that they replace. You need to customize the application or DBMS to accomplish this.

This section contains the procedures for customizing the following software applications and DBMS for use with Subsystem Device Driver:

- Standard UNIX applications
- Network File System file systems
- Oracle.

Standard UNIX applications

If you have not already done so, install Subsystem Device Driver using the procedure in “Installing the Subsystem Device Driver” on page 50. When this is done, the Subsystem Device Driver resides above the HP SCSI disk driver (sdisk) in the protocol stack. In other words, Subsystem Device Driver now talks to the HP-UX device layer. To use standard UNIX applications with Subsystem Device Driver, you must make some changes to your logical volumes. You must either convert your existing logical volumes or create new ones.

Standard UNIX applications such as **newfs**, **fsck**, **mkfs**, and **mount**, that normally take a disk device or raw disk device as a parameter, also accept the Subsystem Device Driver device as a parameter. Similarly, entries in files such as vfstab and dfstab (in the format of cntndnsn) can be replaced by entries for the corresponding Subsystem Device Driver devices’ vpathNs. Make sure that the devices that are replaced are replaced with the corresponding Subsystem Device Driver device. Running the **showvpath** command lists all Subsystem Device Driver devices and their underlying disks.

In order to use the Subsystem Device Driver driver for an existing logical volume, it is necessary to remove the existing logical volume and volume group and recreate it using the Subsystem Device Driver device.

Attention: Do not use the Subsystem Device Driver for critical file systems needed at boot time, such as /(root), /stand, /usr, /tmp or /var. Doing so may render your system unusable if Subsystem Device Driver is ever uninstalled (for example, as part of an upgrade).

Converting existing logical volumes

The task of converting an existing logical volume to use Subsystem Device Driver can be broken down into the following subtasks:

1. Determining the size of the logical volume
2. Removing the existing logical volume
3. Removing the existing volume group
4. Recreating the logical volume.

Note: You must have super-user privileges to perform these subtasks.

As an example, suppose you have a logical volume called lvol1 under a volume group vgibm, which is currently using the disk directly, (for example, through path /dev path /dev/dsk/c3t4d0). You would like to convert logical volume lvol1 to use Subsystem Device Driver. In order to recreate the logical volume, you first need to determine the size of the logical volume.

Determining the size of the logical volume: Use the **lvdisplay** command to determine this:

```
# lvdisplay | grep LV Size
```

A message is displayed:

```
LV Size (Mbytes) 100
```

In this case, the logical volume size is 100 megabytes. Next, remove the logical volume from the system.

Removing the existing logical volume: Before the logical volume is removed, it must be unmounted. Here is an example of using the **umount** command to unmount logical volume lvol1:

```
# umount /dev/vgibm/lvol1
```

Next, remove the logical volume. You can use the following command to remove logical volume lvol1:

```
# lvremove /dev/vgibm/lvol1
```

A message is displayed:

```
The logical volume "/dev/vgibm/lvol1" is not empty;
do you really want to delete the logical volume (y/n)
```

Type `Y` and press Enter. A message is displayed that is similar to the following:

```
Logical volume "/dev/vgibm/lvol1" has been successfully removed.
Volume Group configuration for /dev/vgibm has been saved in
/etc/lvmconf/vgibm.conf
```

When prompted to delete the logical volume, type `y`.

Next, remove the volume group.

Removing the existing volume group: You can use the following command to remove the volume group vgibm:

```
# vgremove /dev/vgibm
```

You see a message similar to this:

```
Volume group "/dev/vgibm" has been successfully removed.
```

Now recreate the logical volume.

Recreating the logical volume: Recreating the logical volume consists of a number of smaller steps:

1. Recreating the physical volume
2. Recreating the volume group
3. Recreating the logical volume
4. Setting the proper timeout value for the logical volume manager.

Recreating the physical volume: Use the following command to recreate the physical volume:

```
# pvcreate /dev/rdsk/vpath0
```

You see a message similar to this:

```
Physical volume "/dev/rdsk/vpath0" has been successfully created.
```

This assumes that the Subsystem Device Driver device associated with the underlying disk is vpath0. Verify this with the **showvpath** command:

```
# /opt/IBMdpo/bin/showvpath
```

A message is displayed:

```
vpath0:
/dev/dsk/c3t4d0
```

Next, recreate the volume group.

Recreating the volume group: Use the following command to recreate the volume group:

```
# vgcreate /dev/vgibm /dev/dsk/vpath0
```

You see a message that says:

```
Increased the number of physical extents per physical volume to 2187.
Volume group "/dev/vgibm" has been successfully created.
Volume Group configuration for /dev/vgibm has been saved in
/etc/lvmconf/vgibm.conf
```

Now recreate the logical volume.

Recreating the logical volume: **Attention:** The recreated logical volume should be the same size as the original volume; otherwise, the recreated volume cannot store the data that was on the original.

Use the following command used to recreate the logical volume:

```
# lvcreate -L 100 -n lvol1 vgibm
```

You see a message that says:

```
Logical volume "/dev/vgibm/lvol1" has been successfully created with
character device "/dev/vgibm/rlvol1".
Logical volume "/dev/vgibm/lvol1" has been successfully extended.
Volume Group configuration for /dev/vgibm has been saved in
/etc/lvmconf/vgibm.conf
```

Note that the *-L 100* parameter comes from the size of the original logical volume, determined by using the **lvdisplay** command. In this example, the original logical volume was 100 MB in size.

Setting the correct timeout value for the logical volume manager: **Attention:** The timeout values for the logical volume manager must be set correctly for the Subsystem Device Driver to operate properly. This is particularly true if you are going to be using concurrent microcode download.

If you are going to be using concurrent microcode download with single-path SCSI, perform the following steps to set the correct timeout value for the logical volume manager:

1. Ensure the timeout value for a Subsystem Device Driver logical volume is set to `default`. Type **lvdisplay /dev/vgibm/lvol1** and press Enter. If the timeout value is not `default`, type **lvchange -t 0 /dev/vgibm/lvol1** and press Enter to change it. (vgibm is the name of the logical volume group previously configured to use the Subsystem Device Driver; in your environment the name may be different.)
2. Change the timeout value for a Subsystem Device Driver physical volume to `240`. Type **pvchange -t 240 /dev/dsk/vpathn** and press Enter. (*n* refers to the vpath number.) If you are not sure about the vpath number, type **/opt/IBMdpo/bin/showvpath** and press Enter to obtain this information.

If you are going to be using concurrent microcode download with multi-path SCSI, perform the following steps to set the proper timeout value for the logical volume manager:

1. Ensure the timeout value for a Subsystem Device Driver logical volume is set to `default`. Type **lvdisplay /dev/vgibm/lvoly** and press Enter. If the time-out value is not `default`, type **lvchange -t 0 /dev/vgibm/lvoly** and press Enter to change it. (vgibm is the name of logical volume group previously configured to use the Subsystem Device Driver; in your environment the name may be different, y=[0,1,2,...].)
2. Change the timeout value for a Subsystem Device Driver physical volume to `240`. Type **pvchange -t 240 /dev/dsk/vpathn** and press Enter. (*n* refers to the vpath number.) If you are not sure about the vpath number, type **/opt/IBMdpo/bin/showvpath** and press Enter to obtain this information.

Note: The recreated logical volume must be mounted before it can be accessed.

Creating new logical volumes

The task of creating a new logical volume to use Subsystem Device Driver can be broken down into the following subtasks:

Note: You must have super-user privileges to perform the following subtasks.

1. Determining the major number of the logical volume device
2. Creating a device node for the logical volume device
3. Creating a physical volume
4. Creating a volume group
5. Creating a logical volume

6. Creating a file system on the volume group
7. Mounting the logical volume.

In order to create a new logical volume that uses the Subsystem Device Driver, you first need to determine the major number of the logical volume device.

Determining the major number of the logical volume device: Use the **lsdev** command to determine this:

```
# lsdev | grep lv
```

A message is displayed:

```
64          64          lv              lvm
```

The first number is the major number of the character device, which is what you want to use. Next, create a device node for the logical volume device.

Creating a device node for the logical volume device: Creating a device node actually consists of:

1. Creating a directory in /dev for the volume group
2. Changing to the /dev directory
3. Creating a device node for the logical volume device.

Creating a directory in /dev for the volume group: Use the following command to create a directory in /dev for the volume group:

```
# mkdir /dev/vgibm
```

In this example, vgibm is the name of the directory.

Next, change to the directory that you just created

Changing to the /dev directory: Use the following command to change to the /dev directory:

```
# cd /dev/vgibm
```

Next, create a device node for the logical volume device.

Creating a device node for the logical volume device: If you do not have any other logical volume devices, you can use a minor number of 0x010000. In this example, assume that you have no other logical volume devices. Use the following command to create a device node:

```
# mknod group c 64 0x010000
```

Now create a physical volume.

Creating a physical volume: Use the following command to create a physical volume:

```
# pvcreate /dev/rdsk/vpath0
```

Now create the volume group

Creating a volume group: Use the following command to create a volume group:

```
# vgcreate /dev/vgibm /dev/dsk/vpath0
```

Now create the logical volume.

Creating a logical volume: Use the following command to create logical volume lvol1 :

```
# lvcreate -L 100 -n lvol1 vgibm
```

The -L 100 makes a 100 MB volume group; you can make it larger if you want to. Now you are ready to create a file system on the volume group.

Creating a file system on the volume group: Use the following command to create a file system on the volume group:

```
# newfs -F hfs /dev/vgibm/rlvol1
```

Finally, mount the logical volume (assuming that you have a mount point called /mnt).

Mounting the logical volume: Use the following command to mount the logical volume lvol1:

```
# mount /dev/vgibm/lvol1 /mnt
```

Network File System file server

The procedures in this section show how to install Subsystem Device Driver for use with an exported file system (Network File System file server).

Setting up Network File System for the first time

Follow the instructions in this section if you are installing exported file systems on Subsystem Device Driver devices for the first time:

1. If you have not already done so, install Subsystem Device Driver using the procedure in “Installing the Subsystem Device Driver” on page 50.
2. Determine which Subsystem Device Driver (**vpathN**) volumes you will use as file system devices.
3. Create file systems on the selected Subsystem Device Driver devices using the appropriate utilities for the type of file system you will use. If you are using the standard HP-UX LJFS file system, use the following command:

   ```
   # newfs /dev/rdsk/vpathN
   ```

   In this example, *N* is the Subsystem Device Driver device instance of the selected volume. Create mount points for the new file systems.
4. Install the file systems into the directory /etc/fstab. Be sure to set the **mount at boot** field to yes.
5. Install the file system mount points into /etc/exports for export.
6. Reboot.

Installing Subsystem Device Driver on a system that already has Network File System file server

Follow the instructions in this section if you have Network File System file server already configured for exported file systems that reside on a multiport subsystem, and if you want to use Subsystem Device Driver partitions instead of sdisk partitions to access them.

1. List the mount points for all currently exported file systems by looking in the /etc/exports directory.
2. Match the mount points found in step 1 with sdisk device link names (files named /dev/(r)dsk/cntndn) by looking in the /etc/fstab directory.

3. Match the sdisk device link names found in step 2 with Subsystem Device Driver device link names (files named /dev/(r)dsk/vpathN) by running the **showvpath** command.
4. Make a backup copy of the current /etc/fstab file.
5. Edit the /etc/fstab file, replacing each instance of an sdisk device link named /dev/(r)dsk/cntndn with the corresponding Subsystem Device Driver device link.
6. Reboot. Verify that each exported file system passes the boot time **fsck pass**, that each mounts properly, and that each is exported and available to NFS clients.

If there is a problem with any exported file system after completing step 6, restore the original /etc/fstab file and reboot to restore Network File System service. Then review your steps and try again.

Oracle

Notes:

1. Procedures listed below require you to have Oracle documentation on hand.
2. You must have super-user privileges to perform these procedures.
3. These procedures were tested with Oracle 8.0.5 Enterprise server, with the 8.0.5.1 patch set from Oracle.

Installing an Oracle database for the first time

You can set up your Oracle database in one of two ways. You can set it up to use a file system or raw partitions. The procedure for installing your database differs depending on the choice you make.

If using a file system:

1. If you have not already done so, install Subsystem Device Driver using the procedure in “Installing the Subsystem Device Driver” on page 50.
2. Create and mount file systems on one or more Subsystem Device Driver partitions (Oracle recommends three mount points on different physical devices).
3. Follow the *Oracle Installation Guide* for instructions on installing to a file system. (During the Oracle installation, you will be asked to name three mount points. Supply the mount points for the file systems you created on the Subsystem Device Driver partitions).

If using raw partitions:

Notes:

1. Make sure that the ownership and permissions of the Subsystem Device Driver devices are the same as the ownership and permissions of the raw devices they are replacing.
2. Make sure that all the databases are closed before making changes.

In the following procedure you will be replacing the raw devices with the Subsystem Device Driver devices.

1. If you have not already done so, install Subsystem Device Driver using the procedure in “Installing the Subsystem Device Driver” on page 50.
2. Create the Oracle Software Owner user in the server’s local /etc/passwd file. You must also complete the following related activities:
   a. Complete the rest of the Oracle pre-installation tasks described in the *Oracle8 Installation Guide*.

b. Plan the installation of Oracle8 on a file system residing on a Subsystem Device Driver partition.

c. Set up the Oracle user's ORACLE_BASE and ORACLE_ HOME environment variables to be directories of this file system.

d. Create two more Subsystem Device Driver-resident file systems on two other Subsystem Device Driver volumes. Each of the resulting three mount points should have a subdirectory named oradata, to be used as a control file and redo log location for the Installer's Default Database (a sample database) as described in the *Oracle8 Installation Guide*. Oracle recommends using raw partitions for redo logs. To use Subsystem Device Driver raw partitions as redo logs, create symbolic links from the three redo log locations to Subsystem Device Driver raw device links (files named /dev/rdsk/vpathNs, where N is the Subsystem Device Driver instance number, and *s* is the partition ID) that point to the slice.

3. Determine which Subsystem Device Driver (vpathN) volumes you will use as Oracle8 database devices.
4. Partition the selected volumes using the HP-UX format utility. If Subsystem Device Driver raw partitions are to be used by Oracle8 as database devices, be sure to leave disk cylinder 0 of the associated volume unused. This protects UNIX disk labels from corruption by Oracle8, as described in the *Oracle8 Installation Guide* in the information on *raw devices*.
5. Ensure that the Oracle Software Owner has read and write privileges to the selected Subsystem Device Driver raw partition device files under the /devices directory.
6. Set up symbolic links from the oradata directory (under the first of the three mount points) that link the database files system<db>.dbf, tempdb.dbf, rbsdb.dbf, toolsd.bdbf, and usersdb.dbf to Subsystem Device Driver raw device links (files named /dev/rdsk/vpathNs) pointing to partitions of the appropriate size, where ″ db″ is the name of the database that you are creating. (The default is test.)
7. Install the Oracle8 Server following the instructions in the *Oracle8 Installation Guide*. Be sure to be logged in as the Oracle Software Owner when you run the **orainst /m** command. Select the **Install New Product - Create Database Objects** option. Select **Raw Devices** for storage type. Specify the raw device links set up in steps 2 on page 59 and 6 for the redo logs and database files of the default database.
8. To set up other Oracle8 databases you must set up control files, redo logs, and database files following the guidelines in the *Oracle8 Administrator's Reference*. Make sure any raw devices and file systems you set up reside on Subsystem Device Driver volumes.
9. Launch the sqlplus utility.
10. Use the **create database** SQL command, specifying the control, log, and system data files that you have set up.
11. Use the **create tablespace SQL** command to set up each of the temp, rbs, tools, and users database files that you created.
12. Use the **create rollback segment** SQL command to create the three redo log files that you set. For the syntax of these three **create** commands, see the *Oracle8 Server SQL Language Reference Manual*.

Installing Subsystem Device Driver on a system that already has Oracle in place

Your installation procedure for a new Subsystem Device Driver install will differ depending on whether you are using a file system or raw partitions for your Oracle database.

If using a file system: Follow this procedure if you are installing Subsystem Device Driver for the first time on a system with an Oracle database that uses a file system:

1. Record the raw disk partitions being used (they are in the cntndnsn format) or the partitions where the Oracle file systems reside. You can get this information from /etc/vfstab if you know where the Oracle files are. Your database administrator can tell you where the Oracle files are, or you can check for directories with the name oradata.
2. Complete the basic installation steps in "Installing the Subsystem Device Driver" on page 50.
3. Change to the directory where you installed the Subsystem Device Driver utilities. Run the **showvpath** command.
4. Check the display to see whether you find a cntndn directory that is the same as the one where the Oracle files are.
5. Use the Subsystem Device Driver partition identifiers instead of the original HP-UX identifiers when mounting the file systems.

   If you would originally have used:

   ```
   mount /dev/dsk/c1t3d2 /oracle/mp1
   ```

   You now use:

   ```
   mount /dev/dsk/vpath2 /oracle/mp1
   ```

   (assuming that you had found vpath2 to be the Subsystem Device Driver identifier)

Follow the instructions in the *Oracle Installation Guide* for setting ownership and permissions.

If using raw partitions: Follow this procedure if you have Oracle8 already installed and want to reconfigure it to use Subsystem Device Driver partitions instead of sdisk partitions (for example, partitions accessed through /dev/rdsk/cntndn files).

All Oracle8 control, log, and data files are accessed either directly from mounted file systems, or using links from the oradata subdirectory of each Oracle mount point set up on the server. Therefore, the process of converting an Oracle installation from sdisk to Subsystem Device Driver has two parts:

- Changing the Oracle mount points' physical devices in /etc/fstab from sdisk device partition links to the Subsystem Device Driver device partition links that access the same physical partitions.
- Recreating links to raw sdisk device links to point to raw Subsystem Device Driver device links that access the same physical partitions.

Converting an Oracle installation from sdisk to Subsystem Device Driver: Following are the conversion steps:

1. Back up your Oracle8 database files, control files, and redo logs.

2. Obtain the sdisk device names for the Oracle8 mounted file systems by looking up the Oracle8 mount points in /etc/fstab and extracting the corresponding sdisk device link name. (for example, /dev/rdsk/c1t4d0)
3. Launch the sqlplus utility.
4. Type the command:

   ```
   select * from sys.dba_data_files;
   ```

   Determine the underlying device that each data file resides on, either by looking up mounted file systems in /etc/fstab, or by extracting raw device link names directly from the **select** command output.
5. Fill in the following table, which is for planning purposes:

Oracle Device Link	File Attributes			Subsystem Device Driver Device Link
Owner		Group	Permissions	
/dev/rdsk/c1tld0	oracle	dba	644	/dev/rdsk/vpath4

6. Fill in column 2 by running **ls -l** on each device link listed in column 1 and extracting the link source device file name.
7. Fill in the **File Attributes** columns by running **ls -l** on each **Actual Device Node** from column 2.
8. Install Subsystem Device Driver following the instructions in the “Installing the Subsystem Device Driver” on page 50.
9. Fill in the **Subsystem Device Driver Device Links** column by matching each **cntndnsn** device link listed in the **Oracle Device Link** column with its associated **vpathN** device link name by running the command:

   ```
   /opt/IBMdpo/showvpath
   ```
10. Fill in the **Subsystem Device Driver Device Nodes** column by running **ls -l** on each Subsystem Device Driver Device Link and tracing back to the link source file.
11. Change the attributes of each node listed in the **Subsystem Device Driver Device Nodes** column to match the attributes listed to the left of it in the **File Attributes** column using the UNIX **chown, chgrp**, and **chmod** commands.
12. Make a copy of the existing /etc/fstab file. Edit the /etc/fstab file, changing each Oracle device link to its corresponding Subsystem Device Driver device link.
13. For each link found in an oradata directory, recreate the link using the appropriate Subsystem Device Driver device link as the source file instead of the associated sdisk device link listed in the Oracle Device Link column.
14. Reboot the server.
15. Verify that all file system and database consistency checks complete successfully.

Uninstalling Subsystem Device Driver

Note: You must uninstall the current level of Subsystem Device Driver must be uninstalled before upgrading to a newer level.

Complete the following procedure to uninstall the Subsystem Device Driver:

1. Reboot or umount all Subsystem Device Driver file systems.

2. If you are using the Subsystem Device Driver with a database, such as Oracle, edit the appropriate database configuration files (database partition) to remove all the Subsystem Device Driver devices.
3. Run **sam**

   ```
   > sam
   ```
4. Select **Software Management**.
5. Choose **Remove Software**.
6. Choose **Remove Local Host Software**.
7. Choose the **IBMdpo_tag** selection.
   a. Select **Actions** from the Bar menu, then select **Mark for Remove**.
   b. Select **Actions** from the Bar menu, then select **Remove (analysis)**. You will see a Remove Analysis window, and on it you will see the status of **Ready**.
   c. Select **OK** to proceed. A Confirmation window is displayed which states that the uninstall will begin.
   d. Type `Yes`. The analysis phase starts.
   e. After the analysis phase has finished, another Confirmation window is displayed informing you that the system will be rebooted after the uninstall is complete. Type `Yes` and press Enter. The uninstall of IBMdpo will now proceed.
   f. Next, an Uninstall window is displayed which informs you about the progress of the IBMdpo software uninstall. This is what the window looks like:

      ```
      Target        : XXXXX
      Status        : Executing unconfigure
      Percent Complete    : 17%
      Kbytes Removed      : 340 of 2000
      Time Left (minutes) : 5
      Removing Software   : IBMdpo_tag,..........
      ```

      While the uninstall process is going on, the **Done** option is not available for you to select. Once the uninstall is complete, the option is available for you to select.
8. Click **Done**. A Note window is displayed informing you that the local system will reboot with the newly installed software.
9. Select **OK** to proceed. The following message is displayed on the machine console before it reboots:

   ```
   * A reboot of this system is being invoked. Please wait.

   *** FINAL System shutdown message (XXXXX) ***
   System going down IMMEDIATELY
   ```

Note: When Subsystem Device Driver has been successfully uninstalled, the first part of the procedure for upgrading the Subsystem Device Driver is complete. To complete an upgrade, you need to reinstall Subsystem Device Driver. See the installation procedure in “Installing the Subsystem Device Driver” on page 50.

The uninstall of the Subsystem Device Driver involved the following actions:

- The /sbin/defvpath file was removed.
- The /usr/conf/master.d/vpath, /etc/ibmvpath, and /usr/conf/lib/libvpath.a files were removed.
- The files in the /opt/IBMdpo directory were removed.

- The /opt/IBMdpo directory was removed.
- The Subsystem Device Driver driver entry was removed from the /stand/system file.
- The Subsystem Device Driver driver was removed from the /stand/vmunix kernel.

Changing a Subsystem Device Driver hardware configuration

When adding or removing multiport SCSI devices from your system, you must reconfigure Subsystem Device Driver to recognize the new devices. Perform the following steps to reconfigure Subsystem Device Driver:

1. Run **cfgvpath** to remake /etc/ibmvpath:

   ```
   # cfgvpath -c
   ```

2. Reboot the system:

   ```
   # reboot
   ```

Chapter 7. Installing and configuring IBM Subsystem Device Driver on a Sun host

Note: For updated and additional information not included in this manual, see the README file on the CDR or visit the Subsystem Device Driver Web site www.ibm.com/storage/support/techsup/swtechsup.nsf/support/sddupdates

This chapter provides instructions to install and set up the Subsystem Device Driver on an host system attached to an ESS.

Understanding how Subsystem Device Driver works on a Sun host

Subsystem Device Driver failover system

The Subsystem Device Driver failover system is designed to provide recovery upon failure of a data path. If this situation occurs, then the failover system selects an alternate data path, while minimizing any disruptions in operation. This failover process consists of:

- Detecting a failure
- Signaling to the host that a failure has occurred
- Compensating for the failure by selecting an alternate data path.

The Subsystem Device Driver dynamically selects an alternate I/O data path when a software or hardware problem is detected. For the failover system to work, the device must be configured for multi-path access.

Subsystem Device Driver

Notes:

1. The Subsystem Device Driver supports 32-bit mode applications on Solaris 2.6.
2. The Subsystem Device Driver supports 64-bit mode applications on Solaris 7.

The Subsystem Device Driver resides above the Sun SCSI disk driver (sd) in the protocol stack. There can be a maximum of eight sd devices underneath each Subsystem Device Driver device in the protocol stack. Each sd device represents a different path to the physical device. There can be up to eight sd devices that represent up to eight different paths to the physical device.

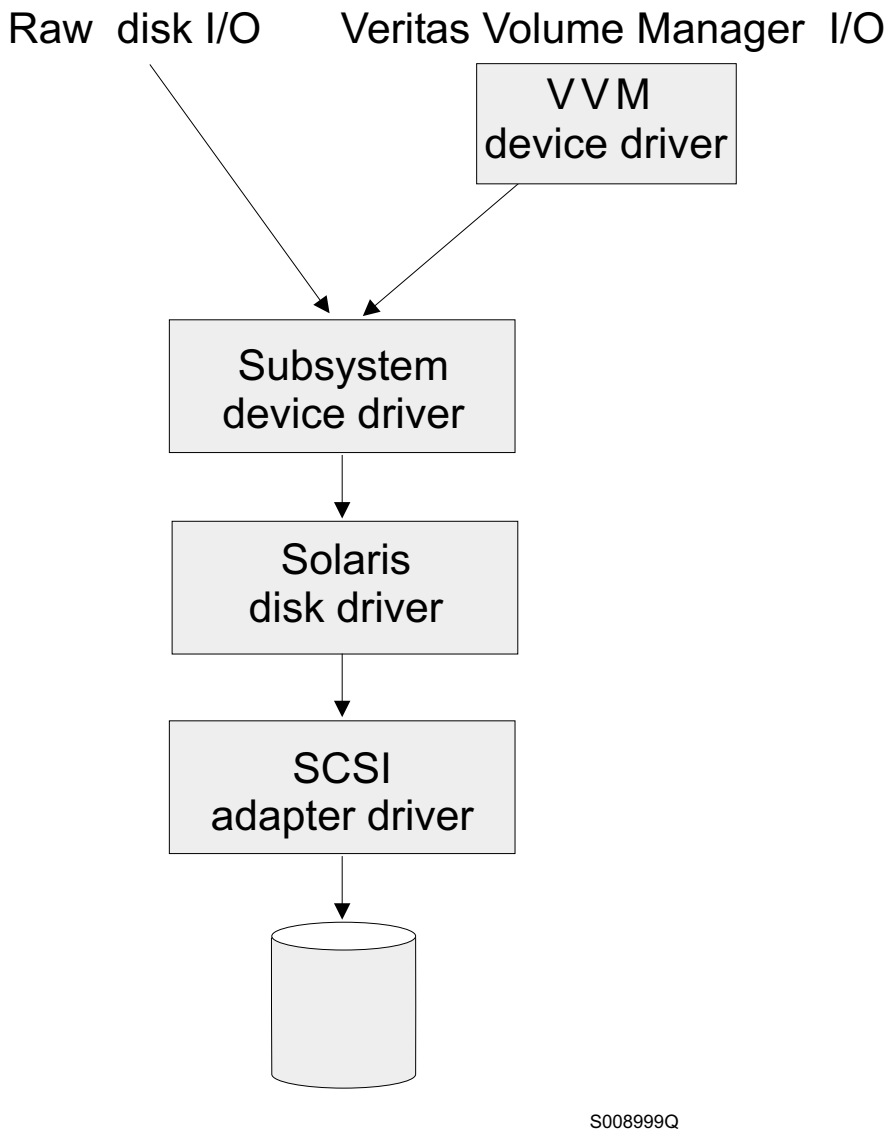


Figure 7. Where the Subsystem Device Driver fits in the protocol stack

Subsystem Device Driver devices behave exactly like sd devices. Any operation on an sd device can be performed on the Subsystem Device Driver device, including commands such as **mount**, **open**, **close**, **umount**, **dd**, **newfs**, or **fsck**. For example, with Subsystem Device Driver you enter **mount /dev/dsk/vpath0c /mnt1** instead of the Solaris **mount /dev/dsk/c1t2d0s2 /mnt1** command.

The Subsystem Device Driver acts as a pass-through agent. I/Os sent to the device driver are passed to an sdisk driver after path selection. When an active path experiences a failure (such as a cable or controller failure), the device driver dynamically switches to another path. The device driver dynamically balances the load, based on the workload of the adapter.

The Subsystem Device Driver also supports one SCSI adapter on the host system. With single-path access, concurrent download of licensed internal code is supported. However, the load balancing and failover features are not available.

Hardware and software requirements

You must meet the following minimum hardware and software requirements to install the Subsystem Device Driver on your host system:

- A Sparc system running Solaris 2.6 or Solaris 7
- A multiport storage subsystem; for example, multi-active redundant RAID controllers (ESS)
- One or more pairs of SCSI host adapters, Sun Fast/Wide/Differential (FWD) X6541A PCI bus cards
- Subsystem LUNs which have been created and confirmed for multiport access. There should be up to eight sdisk instances for each LUN. (One for each path on the server)
- You need a SCSI cable to connect each SCSI host adapter to a storage system controller port per cable.

Configuring the ESS

Before you install the Subsystem Device Driver, configure your ESS for single-port or multiple-port access for each LUN. The Subsystem Device Driver requires a minimum of two independent paths that share the same logical unit to use the load balancing and failover features.

It is strongly recommended that the paths be on different ESS clusters to achieve maximum availability.

For information about configuring your ESS, see *IBM Enterprise Storage Server Introduction and Planning Guide*, GC26–7294.

Planning for installation

Before you install Subsystem Device Driver on your Sun host, you need to understand what kind of software is running on it. The way you install Subsystem Device Driver depends on the kind of software you are running. Basically, there are three types of software that talk directly to raw or block disk device interfaces such as sd and Subsystem Device Driver:

- UNIX file systems, where there is no logical volume manager present
- Logical volume managers (LVMs), such as Sun's Solstice Disk Suite. LVMs allow the system manager to logically integrate, for example, several different physical volumes to create the image of a single large volume.
- Major application packages, such as certain database managers (DBMSs).

There are three possible scenarios for installing Subsystem Device Driver. The scenario you choose depends on the kind of software you have installed:

Scenario 1

Your system has no volume manager, DBMS, or software applications (other than UNIX) that talk directly to the Solaris disk device layer.

Scenario 2

Your system already has a volume manager, software application, or DBMS, such as Oracle, that talks directly with the Solaris disk device drivers.

Scenario 3

Your system already has Subsystem Device Driver and you want to upgrade the software.

The following table further describes the various installation scenarios and how you should proceed.

Table 7. Subsystem Device Driver installation scenarios

Installation Scenario	Description	How To Proceed
Scenario 1	• Subsystem Device Driver not installed • No volume managers • No software application or DBMS installed that talks directly to sd interface	Go to: 1. "Installing the Subsystem Device Driver" 2. "Standard UNIX applications" on page 71
Scenario 2	• Subsystem Device Driver not installed • Existing volume manager, software application, or DBMS installed that talks directly to sd interface	Go to: 1. "Installing the Subsystem Device Driver" 2. "Using applications with Subsystem Device Driver" on page 71
Scenario 3	Subsystem Device Driver installed	Go to: "Upgrading the Subsystem Device Driver" on page 71

Installing the Subsystem Device Driver

You need to complete the following procedure if you are installing Subsystem Device Driver for the first time on your Sun host.

1. Make sure the Subsystem Device Driver (CDR) is available.
2. Insert the CDR into your CD-ROM drive.
3. Change to the install directory:

   ```
   # cd /cdrom/cdrom0/solaris2.6 or
   # cd /cdrom/cdrom0/solaris7
   ```

4. Run **pkgadd**, and point the **–d** option of pkgadd to the directory containing IBMdpo. For example,:

   ```
   pkgadd -d /cdrom/cdrom0/solaris2.6 IBMdpo or
   pkgadd -d /cdrom/cdrom0/solaris7 IBMdpo
   ```

5. You should see messages similar to this:

   ```
   The following packages are available:
   1 IBMcli ibm2105cli
            (sparc) 1.1.0.0
   2 IBMdpo IBM DPO driver Version: May-10-2000 16:51
            (sparc) 1
   Select package(s) you wish to process (or 'all' to process
   all packages). (default: all) [?,??,q]:
   ```

   Type 2 and press Enter to proceed.

6. You should messages similar to this:

```
Processing package instance <IBMdpo> from <var/spool/pkg>


IBM DPO driver
(sparc) 1
## Processing package information.
## Processing system information.
## Verifying disk space requirements.
## Checking for conflicts with packages already installed.
## Checking for setuid/setgid programs.

This package contains scripts which will be executed with super-user
permission during the process of installing this package.

Do you want to continue with the installation of <IBMdpo> [y,n,?]
```

7. Type `Y` and press Enter to proceed.
8. You should see messages similar to this:

```
Installing IBM DPO driver as <IBMdpo>

## Installing part 1 of 1.
/etc/defvpath
/etc/rc2.d/S00vpath-config
/etc/rcS.d/S20vpath-config
/kernel/drv/vpathdd
/kernel/drv/vpathdd.conf
/opt/IBMdpo/cfgvpath
/opt/IBMdpo/datapath
/opt/IBMdpo/devlink.vpath.tab
/opt/IBMdpo/etc.system
/opt/IBMdpo/pathtest
/opt/IBMdpo/showvpath
/usr/sbin/vpathmkdev
[ verifying class <none>  ]
## Executing postinstall script.

DPO: Configuring 24 devices (3 disks * 8 slices)

Installation of <IBMdpo> was successful.

The following packages are available:
1 IBMcli ibm2105cli
         (sparc) 1.1.0.0
2 IBMdpo IBM DPO driver Version: May-10-2000 16:51
         (sparc) 1
Select package(s) you wish to process (or 'all' to process
all packages). (default: all) [?,??,q]:
```

   Type `q` and press Enter to proceed.
9. You should see messages similar to this:

```
*** IMPORTANT NOTICE ***
This machine must now be rebooted in order to ensure
sane operation. Execute
       shutdown -y -i6 -g0
and wait for the "Console Login:" prompt.

DPO is now installed. Proceed to Post-Installation.
```

Note: You can verify that the Subsystem Device Driver has been successfully installed by issuing the **datapath query device** command. If the command executes, the Subsystem Device Driver is installed.

Post-installation

After the installation is complete, manually unmount the CDR. Run the **umount /cdrom** command from the root directory. Go to the CD-ROM drive and press the Eject button.

After Subsystem Device Driver is installed, your system must be rebooted to ensure proper operation. Type the command:

```
# shutdown -i6 -g0 -y
```

Note: Subsystem Device Driver devices are found in the /dev/rdsk and /dev/dsk directories. The device is named according to the Subsystem Device Driver *instance* number. A device with an *instance* number of 0 would be: /dev/rdsk/vpath0a where *a* denotes the slice. Therefore, /dev/rdsk/vpath0c would be instance zero (0) and slice 2.

After Subsystem Device Driver is installed, the device driver resides above the Sun SCSI disk driver (sd) in the protocol stack. In other words, Subsystem Device Driver now talks to the Solaris device layer. The Subsystem Device Driver software installation procedure installs a number of Subsystem Device Driver components and updates some system files. Those components and files are listed in the following tables:

Table 8. System files updated

File	Location	Description
/etc/system	/etc	Forces the loading of the Subsystem Device Driver
/etc/devlink.tab	/etc	Tells the system how to name Subsystem Device Driver devices in /dev

Table 9. Subsystem Device Driver components installed

File	Location	Description
vpathdd	/kernel/drv	Device driver
vpathdd.conf	/kernel/drv	Subsystem Device Driver config file
Executables	/opt/IBMdpo/bin	Configuration and status tools
S20vpath-config	/etc/rcS.d	Boot initialization script*

Note: * This script must come before other LVM initialization scripts, such as Veritas initialization scripts.

If you are not using a volume manager, software application, or DBMS that talks directly to the sd interface, then the installation procedure is nearly complete. If you have a volume manager, software application, or DBMS installed that talks directly to the sd interface, such as Oracle, go to “Using applications with Subsystem Device Driver” on page 71 and read the information specific to the application you will be using.

Upgrading the Subsystem Device Driver

Upgrading the Subsystem Device Driver consists of uninstalling and reinstalling the IBMdpo package. If you are upgrading Subsystem Device Driver, go to “Uninstalling Subsystem Device Driver” on page 83 and then go to “Installing the Subsystem Device Driver” on page 68.

Using applications with Subsystem Device Driver

If your system already has a volume manager, software application, or DBMS installed that communicates directly with the Solaris disk device drivers, you need to insert the new Subsystem Device Driver device layer between the program and the Solaris disk device layer. You also need to customize the volume manager, software applicaton, or DBMS in order to have it communicate with the Subsystem Device Driver devices instead of the Solaris devices.

In addition, many software applications and DBMSs need to control certain device attributes such as ownership and permissions. Therefore, you must ensure that the new Subsystem Device Driver devices that these software applications or DBMSs access in the future have the same attributes as the Solaris sd devices that they replace. You need to customize the software application or DBMS to accomplish this.

This section describes how to use the following applications with Subsystem Device Driver:

- Standard UNIX applications
- Network File System file systems
- Oracle
- Veritas Volume Manager.

Standard UNIX applications

If you have not already done so, install Subsystem Device Driver using the procedure in section “Installing the Subsystem Device Driver” on page 68. When this is done, the device driver resides above the Solaris SCSI disk driver (sd) in the protocol stack. In other words, the Subsystem Device Driver now talks to the Solaris device layer.

Standard UNIX applications such as **newfs, fsck, mkfs, and mount**, that normally take a disk device or raw disk device as a parameter, also accept the Subsystem Device Driver device as a parameter. Similarly entries in files such as vfstab and dfstab (in the format of cntndnsn) can be replaced by entries for the corresponding Subsystem Device Driver devices’ vpathNs. Make sure that the devices that are replaced are replaced with the corresponding Subsystem Device Driver device. Running the **showvpath** command lists all Subsystem Device Driver devices and their underlying disks.

Note: The Subsystem Device Driver does not support being used for the root (/), /var, /usr, /opt, /tmp and swap partitions.

Network File System file server

The procedures in this section show how to install Subsystem Device Driver for use with an Exported File System (Network File System file server).

Setting up Network File System for the first time

Follow the instructions in this section if you are installing exported file systems on Subsystem Device Driver devices for the first time:

1. If you have not already done so, install Subsystem Device Driver using the procedure in “Installing the Subsystem Device Driver” on page 68.
2. Determine which Subsystem Device Driver (**vpathN**) volumes you will use as file system devices.
3. Partition the selected volumes using the Solaris format utility.
4. Create file systems on the selected Subsystem Device Driver devices using the appropriate utilities for the type of file system you will use. If you are using the standard Solaris LJFS file system, use the following command:

   ```
   # newfs /dev/rdsk/vpathNs
   ```

   In this example, ***N*** is the Subsystem Device Driver device instance of the selected volume. Create mount points for the new file systems.
5. Install the file systems into the **/etc/fstab** directory. Be sure to set the **mount at boot** field to yes.
6. Install the file system mount points into the directory **/etc/exports** for export.
7. Reboot.

Installing Subsystem Device Driver on a system that already has Network File System file server

Follow the instructions in this section if you have Network File System file server already configured for exported file systems that reside on a multiport subsystem, and if you want to use Subsystem Device Driver partitions instead of sd partitions to access them.

1. List the mount points for all currently exported file systems by looking in the /etc/exports directory.
2. Match the mount points found in step 1 with sdisk device link names (files named /dev/(r)dsk/cntndn) by looking in the /etc/fstab directory.
3. Match the sd device link names found in step 2 with Subsystem Device Driver device link names (files named /dev/(r)dsk/vpathN) by running the **showvpath** command.
4. Make a backup copy of the current /etc/fstab file.
5. Edit the /etc/fstab file, replacing each instance of an sd device link named /dev/(r)dsk/cntndn with the corresponding Subsystem Device Driver device link.
6. Reboot. Verify that each exported file system passes the boot time **fsck pass**, that each mounts properly, and that each is exported and available to NFS clients.

If there is a problem with any exported file system after completing step 6, restore the original /etc/fstab file and reboot to restore Network File System service. Then review your steps and try again.

Oracle

Notes:

1. Procedures listed below require you to have Oracle documentation on hand.
2. You must have super-user privileges to perform these procedures.
3. These procedures were tested with Oracle 8.0.5 Enterprise server, with the 8.0.5.1 patch set from Oracle.

Installing a Oracle database for the first time

You can set up your Oracle database in one of two ways. You can set it up to use a file system or raw partitions. The procedure for installing your database differs depending on the choice you make.

If using a file system:

1. If you have not already done so, install Subsystem Device Driver using the procedure in “Installing the Subsystem Device Driver” on page 68.
2. Create and mount file systems on one or more Subsystem Device Driver partitions. (Oracle recommends three mount points on different physical devices.)
3. Follow the *Oracle Installation Guide* for instructions on installing to a file system. (During the Oracle installation, you will be asked to name three mount points. Supply the mount points for the file systems you created on the Subsystem Device Driver partitions.)

If using raw partitions:

Notes:

1. Make sure all the databases are closed before going further.
2. Make sure that the ownership and permissions of the Subsystem Device Driver devices are the same as the ownership and permissions of the raw devices they are replacing.
3. Do not use disk cylinder 0 (sector 0), which is the disk label. Using it corrupts the disk. For example, slice 2 on Sun is the whole disk. If you use this device without repartitioning it to start at sector 1, the disk label is corrupted.

1. If you have not already done so, install Subsystem Device Driver using the procedure outlined in “Installing the Subsystem Device Driver” on page 68.
2. Create the Oracle Software Owner user in the server’s local /etc/passwd file. You must also complete the following related activities:
   a. Complete the rest of the Oracle pre-installation tasks described in the *Oracle8 Installation Guide*. Plan to install Oracle8 on a file system residing on a Subsystem Device Driver partition.
   b. Set up the Oracle user’s ORACLE_BASE and ORACLE_ HOME environment variables to be directories of this file system.
   c. Create two more Subsystem Device Driver-resident file systems on two other Subsystem Device Driver volumes. Each of the resulting three mount points should have a subdirectory named oradata, to be used as a control file and redo log location for the Installer’s Default Database (a sample database) as described in the *Installation Guide*. Oracle recommends using raw partitions for redo logs. To use Subsystem Device Driver raw partitions as redo logs, create symbolic links from the three redo log locations to Subsystem Device Driver raw device links (files named /dev/rdsk/vpathNs, where N is the Subsystem Device Driver instance number, and *s* is the partition ID) that point to the slice.
3. Determine which Subsystem Device Driver (vpathN’) volumes you will use as Oracle8 database devices
4. Partition the selected volumes using the Solaris format utility. If Subsystem Device Driver raw partitions are to be used by Oracle8 as database devices, be sure to leave sector 0/disk cylinder 0 of the associated volume unused. This protects UNIX disk labels from corruption by Oracle8.

5. Ensure the Oracle Software Owner has read and write privileges to the selected Subsystem Device Driver raw partition device files under the /devices/pseudo directory.
6. Set up symbolic links in the oradata directory under the first of the three mount points created in step 2 on page 73 to link the database files to Subsystem Device Driver raw device links (files named /dev/rdsk/vpathNs) pointing to partitions of the appropriate size.
7. Install the Oracle8 Server following the instructions in the *Oracle Installation Guide*. Be sure to be logged in as the Oracle Software Owner when you run the **orainst /m** command. Select the **Install New Product - Create Database Objects** option. Select **Raw Devices** for storage type. Specify the raw device links set up in step 2 on page 73 for the redo logs. Specify the raw device links set up in step 3 on page 73 for the database files of the default database.
8. To set up other Oracle8 databases you must set up control files, redo logs, and database files following the guidelines in the *Oracle8 Administrator's Reference*. Make sure any raw devices and file systems you set up reside on Subsystem Device Driver volumes.
9. Launch the sqlplus utility.
10. Use the **create database** SQL command, specifying the control, log, and system data files that you have set up.
11. Use the **create tablespace** SQL command to set up each of the temp, rbs, tools, and users database files that you created.
12. Use the **create rollback segment** SQL command to create the three redo log files that you set. For the syntax of these three **create** commands, see the *Oracle8 Server SQL Language Reference Manual*.

Installing Subsystem Device Driver on a system that already has Oracle in place

Your installation procedure for a new Subsystem Device Driver install will differ depending on whether you are using a file system or raw partitions for your Oracle database.

If using a file system: Follow this procedure if you are installing Subsystem Device Driver for the first time on a system with a Oracle database that uses a file system:

1. Record the raw disk partitions being used (they are in the cntndnsn format) or the partitions where the Oracle file systems reside.You can get this information from /etc/vfstab if you know where the Oracle files are. Your database administrator can tell you where the Oracle files are, or you can check for directories with the name oradata.
2. Complete the basic installation steps in "Installing the Subsystem Device Driver" on page 68.
3. Change to the directory where you installed the Subsystem Device Driver utilities. Enter the **showvpath** command.
4. Check the display to see whether you find a cntndn directory that is the same as the one where the Oracle files are. For example, if the Oracle files are on c1t8d0s4, look for c1t8d0s2. If you find it, you will know that /dev/dsk/vpath0c is the same as /dev/dsk/clt8d2s2. (Subsystem Device Driver partition identifiers end in abcdefg rather than s0, s1, s2, etc.) Write this down. The output from the **showvpath** command looks similar to this:

```
vpath0c
    c1t8d0s2   /devices/pci@1f,0/pci@1/scsi@2/sd@1,0:c,raw
    c2t8d0s2   /devices/pci@1f,0/pci@1/scsi@2,1/sd@1,0:c,raw
```

5. Use the Subsystem Device Driver partition identifiers instead of the original Solaris identifiers when mounting the file systems.

   If you would originally have used:

   mount /dev/dsk/c1t3d2s4 /oracle/mp1

   You now use:

   mount /dev/dsk/vpath2e /oracle/mp1

   (assuming you had found vpath2c to be the Subsystem Device Driver identifier)

Follow the instructions in *Oracle Installation Guide* for setting ownership and permissions.

If using raw partitions: Follow this procedure if you have Oracle8 already installed and want to reconfigure it to use Subsystem Device Driver partitions instead of sd partitions (for example, partitions accessed through /dev/rdsk/cntndn files).

If the Oracle8 installation is accessing Veritas logical volumes, go to "Veritas Volume Manager" on page 76 for information about installing Subsystem Device Driver with that application.

All Oracle8 control, log, and data files are accessed either directly from mounted file systems, or through links from the oradata subdirectory of each Oracle mount point set up on the server. Therefore, the process of converting an Oracle installation from sd to Subsystem Device Driver has two parts:

1. Changing the Oracle mount points' physical devices in /etc/fstab from sd device partition links to the Subsystem Device Driver device partition links that access the same physical partitions.
2. Recreating any links to raw sd device links to point to raw Subsystem Device Driver device links that access the same physical partitions.

Converting an Oracle installation from sd to Subsystem Device Driver partitions: Following are the conversion steps:

1. Back up your Oracle8 database files, control files, and redo logs.
2. Obtain the sd device names for the Oracle8 mounted file systems by looking up the Oracle8 mount points in /etc/vfstab and extracting the corresponding sd device link name (for example, /dev/rdsk/c1t4d0s4).
3. Launch the **sqlplus** utility.
4. Type the command:

   select * from sys.dba_data_files;

   The output lists the locations of all data files in use by Oracle. Determine the underlying device that each data file resides on, either by looking up mounted file systems in /etc/vfstab or by extracting raw device link names directly from the select command output.
5. Run the **ls -l** command on each device link found in step 4 on page 75 and extract the link source device file name. For example, if you type command:

   # ls -l /dev/rdsk/c1t1d0s4

You might see output that is similar to this:

```
/dev/rdsk/c1t1d0s4 /devices/pci@1f,0/pci@1/scsi@2/sd@1,0:e
```

6. Write down the file ownership and permissions by running the **ls -lL** command on either the files in /dev/ or /devices (it yields the same result). For example, if you type the command:

   ```
   # ls -lL /dev/rdsk/c1t1d0s4
   ```

   You might see output that is similar to this:

   ```
   crw-r--r-- oracle dba 32,252 Nov 16 11:49 /dev/rdsk/c1t1d0s4
   ```

7. Complete the basic installation steps in “Installing the Subsystem Device Driver” on page 68.
8. Match each cntndns device with its associated vpathNs device link name by running the **showvpath** command. Remember that vpathNs partition names use the letters [a-h] in the *s* position to indicate slices [0-7] in the corresponding cntndnsn slice names.
9. Run the **ls -l** command on each Subsystem Device Driver device link.
10. Write down the Subsystem Device Driver device nodes for each Subsystem Device Driver device link by tracing back to the link source file.
11. Change the attributes of each Subsystem Device Driver device to match the attributes of the corresponding disk device using the **chgrp** and **chmod** commands.
12. Make a copy of the existing /etc/vfstab file for recovery purposes. Edit the /etc/vfstab file, changing each Oracle device link to its corresponding Subsystem Device Driver device link.
13. For each link found in an oradata directory, recreate the link using the appropriate Subsystem Device Driver device link as the source file instead of the associated sd device link. As you perform this step, generate a reversing shell script that can restore all the original links in case of error.
14. Reboot the server.
15. Verify that all file system and database consistency checks complete successfully.

Veritas Volume Manager

For these procedures, you should have a copy of the *Veritas Volume Manager System Administrator's Guide*, and *Veritas Volume Manager Command Line Interface for Solaris*. These procedures were tested using Veritas 3.0.1. The Sun patches 105223 and 105357 must be installed with Veritas (this is a Veritas requirement).

Notes:

1. You must have super-user privileges to perform these procedures.
2. Subsystem Device Driver does not support being used for the root (/), /var, /usr, /opt, /tmp and swap partitions.

Installing Veritas Volume Manager for the first time

Follow the instructions in this section if you are installing Veritas on the multiport subsystem's server for the first time. Installing Veritas for the first time on a Subsystem Device Driver system consists of:

1. Installing Subsystem Device Driver using the procedure in “Installing the Subsystem Device Driver” on page 68, if you have not already done so.

2. Adding a Solaris hard disk device to the Veritas root disk group (rootdg).
3. Adding a Subsystem Device Driver device to Veritas.
4. Creating a new disk group from a Subsystem Device Driver device.
5. Creating a new volume from a Subsystem Device Driver device.

Adding a Solaris hard disk device to the Veritas root disk group (rootdg): During installation, Veritas requires that at least one disk device be added to the Veritas root disk group (rootdg). This device must be a standard Solaris hard disk device, and not a Subsystem Device Driver device. It is important that the last disk in the rootdg be a regular disk and not a Subsystem Device Driver device. Therefore, it is recommended that you use a different disk group for your Subsystem Device Driver disks.

Subsystem Device Driver disks may only be added to a Veritas disk group as a whole, for example, any previous partitioning is ignored. The c partition (the whole disk) is used, so the Subsystem Device Driver device name for the disk in the /dev/dsk and /dev/rdsk directories would be vpath0c, for example. Veritas always looks in these directories by default, so only the device name is needed, for example, vpath0c, when issuing Veritas commands.

Partitioning of the given disk once it has been added to a Veritas disk group is achieved by dividing the Veritas disk into Veritas subdisks.

Adding a Subsystem Device Driver device to Veritas: The following is an example of a command that adds a Subsystem Device Driver device to Veritas:

```
vxdisk -f init vpath0c
```

After running this command, the Veritas graphical user interface tool (VMSA) can be used to create a new disk group and, a new volume from a Subsystem Device Driver device.

Attention: VMSA and the command-line interface are the only supported methods of creating new disks or volumes with Veritas.

Creating a new disk group from a Subsystem Device Driver device: The following command creates a new disk group from the Subsystem Device Driver physical device. In this example, the new disk group is called ibmdg and the disk is vpath0c.

```
vxdg init ibmdg vpath0c
```

You can add a Subsystem Device Driver device to an existing disk group using the **vxdgadd** command.

Creating a new volume from a Subsystem Device Driver device: This command gets the maximum size of the disk vpath0c in blocks:

```
/usr/sbin/vxassist -g ibmdg -p maxsize [vpath0c]
```

Write down the output of the last command and use it in the next command, which creates a volume called ibmv within the disk group called ibmdg.

The command to create a volume is:

```
/usr/sbin/vxassist -g ibmdg make ibmv 17846272 layout=nostripe
```

You can change the size of the volume and use less that the maximum number of blocks.

Installing Subsystem Device Driver on a system that already has Veritas Volume Manager in place

This section describes the Veritas command-line instructions needed to reconfigure a Veritas volume for use as a Subsystem Device Driver disk device. This reconfiguration consists of:

- Adding Subsystem Device Driver devices to the disk group that corresponds to the existing sd disks.
- Setting the size of a Subsystem Device Driver device to that of the original disk.
- Setting the size of the original device to zero.

At the conclusion, you will have a disk group that contains twice the number of devices as the original disk group. The new Subsystem Device Driver devices in the disk group will be the same size as the original sd disks. The Solaris operating system will use the Subsystem Device Driver devices, and not the original sd disk.

Note: Versions of Veritas that support multi-pathing (dpm) must be disabled. See *Veritas Volume Manager Release Notes* for instructions on doing this. Some versions of Veritas do not support the disabling of multi-pathing (dpm). In that case, you must first upgrade to a version of Veritas that supports this before proceeding. See the Veritas Volume Manager documentation for further details.

The following procedure assumes that you have:

1. Configured Veritas volumes to use Solaris disk device drivers for accessing the multiport subsystem drives.
2. Created Subsystem Device Driver devices that refer to the same multiport subsystem drive.

These instructions allow you to replace all sd references to the original hard disks that occur in the Veritas volume's configuration with references to the Subsystem Device Driver devices. The example provided shows the general method for replacing the sd device with the corresponding Subsystem Device Driver device in an existing Veritas volume.

Note: At least one device in the rootdg disk group must be a non-Subsystem Device Driver disk; do not attempt to change all the disks in rootdg to Subsystem Device Driver devices.

The example uses the following identifiers:

ibmv the Veritas volume

ibmv-01
the plex associated with the ibmv volume

disk01-01
Veritas VM disk containing the original Sun hard disk device

vpath0c
the Subsystem Device Driver device that refers to the same hard disk that disk01-01 does

c1t1d0s2
the sd disk associated with vpath0c, and disk01-01

disk02
Veritas VM disk containing the vpath0c device

rootdg
the name of the Veritas disk group to which ibmv belongs

A simplifying assumption is that the original volume, ibmv, contains exactly one subdisk. However, the method outlined here should be easy to adapt to other cases.

Before proceeding, record the multiport subsystem device links (/dev/(r)dsk/cntndnsn) being used as Veritas volume device files. Next, determine the corresponding Subsystem Device Driver device link (/dev/(r)dsk/vpathNs) using the **showvpath** command. Record this information.

Reconfiguring a Veritas Volume to use a Subsystem Device Driver disk device:

1. If you have not already done so, install Subsystem Device Driver using the procedure in "Installing the Subsystem Device Driver" on page 68.
2. Display information about the disk used in the volume **ibmv**.

   ```
   vxdisk list c1t1d0
   ```

   The resulting display includes information about the disk, including its public and private offset and length:

   ```
   public:  slice=4 offset=0 len=17846310
   private:  slice=3 offset=1 len=2189
   ```

   From this information, calculate the parameters *privlen* (length of the private region) and *puboffset* (offset of the public region). In this case, *privlen=2189*, and *puboffset=2190* because *puboffset* is one block more than the length of *privlen*.
3. Initialize the Subsystem Device Driver device for use by Veritas as a simple disk, using the *privlen* and *puboffset* values from step 2.

   ```
   vxdisk -f init vpath0c puboffset=2190 privlen=2189
   ```
4. Add the Subsystem Device Driver device to the disk group:

   ```
   vxdg -g rootdg adddisk disk02=vpath0c
   ```
5. Make sure that the file systems that are part of this volume are not mounted and then stop the volume

   ```
   umount /ibmvfs
   vxvol -g rootdg stop ibmv
   ```
6. Get the volume length (in sectors). This information is used in later steps. For this example, a volume length of 17846310 is assumed.

   ```
   vxprint ibmv
   ```
7. Disassociate the plex but do not delete it.

   ```
   vxplex -g rootdg dis ibmv-01
   vxvol -g rootdg set len=0 vol01
   ```

   Attention: The plex should remain to serve as backup should backing out of the Subsystem Device Driver installation be necessary.
8. Create a subdisk from the Subsystem Device Driver VM disk:

   ```
   vxmake -g rootdg sd disk02-01 disk02,0,17846310
   ```

   (Use *len* from step 6)
9. Create a new plex called ibmv-02 containing the disk02-01 subdisk

   ```
   vxmake -g rootdg plex ibmv-02 sd=disk02-01
   ```

10. Attach the plex to the volume

```
vxplex -g rootdg att ibmv ibmv-02
vxvol set len=17846310 ibmv
```

(Use *length* from step 6 on page 79)

11. Make the volume active:

```
vxvol -g rootdg init active ibmv
```

Notes:

a. When a disk is initialized for use by Veritas, it is repartitioned as a sliced disk containing a private region at slice 3 and a public region at slice 4. The length and offsets of these regions can be displayed using the **vxdisk list cntndn** command.

b. When using an sd device as a Subsystem Device Driver device, you must initialize the Subsystem Device Driver disk as a simple disk. This simple disk uses only a single slice (slice 2). The private region starts at block 1, after the disk's VTOC region, which is situated at block 0. Note that the length of the private region varies with the type of disk used, with the public region following the private region.

At this stage you can delete the original disk, after verifying that everything is working correctly.

Solstice DiskSuite

For these procedures, you need access to the Solaris answerbook facility. These procedures were tested using Solstice DiskSuite 4.2, with the patch, 106627-04 (DiskSuite patch), installed. You should have a copy of the *DiskSuite Administration Guide* available to complete these procedures.

Notes:

1. You must have super-user privileges to perform these procedures.
2. Subsystem Device Driver vpath does not support being used for the root (/), /var, /usr, /opt, /tmp and swap partitions.

Installing Solstice DiskSuite for the first time

Perform the following steps if you are installing Solstice DiskSuite on the multiport subsystem's server for the first time. The installation of Solstice DiskSuite for the first time on a Subsystem Device Driver system consists of:

1. Installing Subsystem Device Driver using the procedure in "Installing the Subsystem Device Driver" on page 68, if you have not already done so.
2. Configuring the Sparc server to recognize all devices over all paths using the **boot -r** command.
3. Installing the Solstice DiskSuite packages and the answerbook. Do not reboot yet.

   Note: Do not install the DiskSuite Tool (metatool)
4. Determine which vpath devices you will use to create Disk Suite metadevices. Partition these devices by selecting them in the Solaris format utility. The devices appear as vpathNs, where *N* is the vpath driver instance number). Use the partition submenu, just as you would for an sd device link of the form, cntndn. If you want to know which cntndn links correspond to a particular vpath device, type the **showvpath** command and press Enter. Reserve at least three partitions of three cylinders each for use as DiskSuite Replica database locations.

Note: You do not need to partition any sd (cntndn) devices.

5. Set up the replica databases on a partitions of its own. This partition needs to be at least three partitions of three cylinders, and do not use a partition that includes Sector 0 for this database replica partition. Follow the instructions for setting up replica databases on vpathN's partitions, where *N* is the vpath device instance number and *s* is the letter denoting the three cylinder partition, or slice, of the device that you wish to use as a replica. Remember that partitions [a-h] of a vpath device correspond to slices [0-7] of the underlying multiport subsystem device.
6. Follow the instructions in the *DiskSuite Administration Guide* to build the types of metadevices you need, using the **metainit** command and the /dev/(r)dsk/vpathNs device link names, wherever the instructions specify /dev/(r)dsk/cntndnsn device link names.
7. Insert the setup of all vpathNs devices used by DiskSuite into the /etc/opt/SUNWmd/md.tab file

Installing Subsystem Device Driver on a system that already has Solaris DiskSuite in place

Perform the following steps if Solaris DiskSuite is already installed:

1. Back up all data.
2. Back up the current Solstice configuration by making a copy of the /etc/opt/SUNWmd/md.tab file, and recording the output of the **metastat** and **metadb -i** commands. Make sure all sd device links in use by DiskSuite are entered in md.tab, and that they all come up properly after a reboot.
3. Installing Subsystem Device Driver using the procedure in "Installing the Subsystem Device Driver" on page 68, if you have not already done so. After the installation completes, type the **shutdown -i6 -y -g0** command and press Enter. This verifies the vpath installation.

   Note: Do not do a reconfiguration reboot
4. Using a plain sheet of paper, make a two-column list matching up the /dev/(r)dsk/cntndnsn device links found in step 2 with the corresponding /dev/(r)dsk/vpathNs device links using the **showvpath** command.
5. Delete each replica database currently configured with an /dev/(r)dsk/cntndnsn device, by using the **metadb -d -f <device>** command, and replace it with the corresponding /dev/(r)dsk/vpathNs device found in step 2, by using the **metadb -a <device>** command.
6. Create a new md.tab file, inserting the corresponding vpathNs device link name in place of each cntndnsn device link name. Do not do this for boot device partitions (vpath does not currently support these). When you are confident that the new file is correct, install it in the /etc/opt/SUNWmd directory.
7. Reboot the server or proceed to the next step, if you wish to avoid rebooting your system.
8. Stop all applications using DiskSuite, including file systems.
9. Enter the following commands for each existing metadevice:

   ```
   metaclear <device>
   metainit -a
   ```
10. Restart your applications.

Note: To back out vpath in case of any problems following step 7, reverse the procedures in step 6, reinstall the original md.tab into /etc/opt/SUNWmd, and run the command **pkgrm IBMdpo**, and reboot.

Setting up UFS logging on a new system (no existing UFS logging that need to be switched over to vpath)

For these procedures, you need access to the Solaris answerbook facility.

Notes:

1. You must have super-user privileges to perform these procedures.

Perform the following steps if you are installing a new UFS logging file system on vpath devices:

1. Installing Subsystem Device Driver using the procedure in “Installing the Subsystem Device Driver” on page 68, if you have not already done so.
2. Determine which vpath (vpathNs) volumes you will use as file system devices. Partition the selected vpath volumes using the Solaris format utility. Be sure to create partitions for UFS logging devices as well as for UFS master devices.
3. Create file systems on the selected vpath UFS master device partitions using the **newfs** command.
4. Install Solstice DiskSuite if you have not already done so.
5. Create the metatrans device using metainit. For example, assume /dev/dsk/vpath0d is your UFS master device used in step 3, /dev/dsk/vpath0e is its corresponding log device, and d0 is the trans device you want to create for UFS logging. Type **metainit d0 -t vpath0d vpath0e** and press Enter.
6. Create mount points for each UFS logging file system you have created using steps 3 and 5.
7. Install the file systems into the /etc/vfstab directory, specifying /dev/md/(r)dsk/d <metadevice number> for the raw and block devices. Be sure to set the **mount at boot** field to `yes`.
8. Reboot.

Installing vpath on a System that already has UFS Logging in Place

Perform the following steps if you have UFS logging file systems already residing on a multiport subsystem and you wish to use vpath partitions instead of sd partitions to access them.

1. Make a list of the DiskSuite metatrans devices for all existing UFS logging file systems by looking in the /etc/vfstab directory. Make sure that all configured metatrans devices are correctly set up in the /etc/opt/SUNWmd/md.tab file. If the devices are not set up now, set them up before continuing. Save a copy of md tab.
2. Match the device names found in step 1 with sd device link names (files named /dev/(r)dsk/cntndnsn) through the **metastat** command.
3. Install Subsystem Device Driver using the procedure in “Installing the Subsystem Device Driver” on page 68, if you have not already done so.
4. Match the sd device link names found in step 2 with vpath device link names (files named /dev/(r)dsk/vpathNs) by executing the **/opt/IBMdpo/showvpath** command.
5. Unmount all current UFS logging file systems known to reside on the multiport subsystem through the **umount** command.
6. Type **metaclear -a** and press Enter.
7. Create new metatrans devices from the vpathNs partitions found in step 4 corresponding to the sd device links found in step 2. Remember that vpath partitions [a-h] correspond to sd slices [0-7]. Use the **metainit d <metadevice number> -t <″vpathNs″ - master device> <″vpathNs″ - logging device>** command . Be sure to use the same metadevice numbering as was originally

used with the sd partitions. Edit the /etc/opt/SUNWmd/md.tab file to change each metatrans device entry to use vpathNs devices.

8. Reboot.

Note: If there is a problem with a metatrans device after steps 7 and 8, restore the original /etc/opt/SUNWmd/md.tab file and reboot. Review your steps and try again.

```
Create:
metadb -a -c 3 -f vpath0f # add database replicas
metainit d0 1 1 vpath0e # add metadevice

Info
metastat
metadb -i

Delete:
metaclear d0 # delete metadevice
metadb -d -f vpat
```

Uninstalling Subsystem Device Driver

Note: You must uninstall the current level of Subsystem Device Driver before upgrading to a newer level.

Attention: Do not reboot between the uninstall and the reinstall of the Subsystem Device Driver.

Upgrading the Subsystem Device Driver consists of uninstalling and reinstalling the IBMdpo package. Perform the following steps to uninstall the Subsystem Device Driver:

1. Reboot or umount all Subsystem Device Driver file systems.
2. If you are using the Subsystem Device Driver with a database, such as Oracle, edit the appropriate database configuration files (database partition) to remove all the Subsystem Device Driver devices.
3. If you are using a database, restart the database.
4. Type `# pkgrm IBMdpo` and press Enter.

   Attention: A number of different installed packages is displayed. Make sure you specify the correct package to uninstall.

   A message similar to the following is displayed:

   ```
   The following packages are available:
   1 IBMcli ibm2105cli
            (sparc) 1.1.0.0
   2 IBMdpo IBM DPO driver Version: May-10-2000 16:51
            (sparc) 1
   ```

5. Type `Y` and press Enter. A message similar to the following is displayed:

   ```
   ## Removing installed package instance <IBMdpo>

   This package contains scripts that will be executed with super-user
   permission during the process of removing this package.

   Do you want to continue with the removal of this package [y,n,?,q] y
   ```

6. Type `Y` and press Enter. A message similar to the following is displayed:

```
## Verifying package dependencies.
## Processing package information.
## Executing preremove script.
Device busy
Cannot unload module: vpathdd
Will be unloaded upon reboot.
## Removing pathnames in class <none>
/usr/sbin/vpathmkdev
/opt/IBMdpo
/kernel/drv/vpathdd.conf
/kernel/drv/vpathdd
/etc/rcS.d/S20vpath-config
/etc/rc2.d/S00vpath-config
/etc/defvpath
## Updating system information.

Removal of <IBMdpo> was successful.
```

Attention: Do not reboot at this time.

Note: When the Subsystem Device Driver has been successfully uninstalled, the first part of the procedure for upgrading the Subsystem Device Driver is complete. To complete the upgrade, you now need to reinstall Subsystem Device Driver. See “Installing the Subsystem Device Driver” on page 68 for detailed procedures.

Changing a Subsystem Device Driver hardware configuration

When adding or removing multiport SCSI devices from your system, you must reconfigure Subsystem Device Driver to recognize the new devices. Perform the following steps to reconfigure the Subsystem Device Driver:

1. Shut down the system. Type **# shutdown -i0 -g0 -y** and press Enter.
2. Do a configuration reboot. From the **OK** prompt, type **boot -r** and press Enter. This uses the current Subsystem Device Driver entries during reboot, not the new entries. The reboot forces the new disks to be recognized.
3. After the reboot, run the Subsystem Device Driver configuration utility to make the changes to /etc/ibmvpath. Type **# cfgvpath -c** and press Enter.
4. Shut down the system. Type **shutdown -i6 -g0 -y** and press Enter.

Chapter 8. Using commands

The Subsystem Device Driver provides commands that you can use to display the status of adapters that are used to access managed devices, or to display the status of devices that the device driver manages. You can also set individual path conditions either to online or offline, or you can set all paths that are connected to an adapter or bus either to online or offline. This chapter includes descriptions of these commands. Table 10 provides an alphabetical list of these commands, a brief description, and where to go in this chapter for more information.

Table 10. Commands

Command	Description	Page
datapath query adapter	Displays information about adapters	86
datapath query adaptstats	Displays performance information for all SCSI adapters that are attached to Subsystem Device Driver devices	88
datapath query device	Displays information about devices	89
datapath query devstats	Displays performance information for a single Subsystem Device Driver device or all Subsystem Device Driver devices	91
datapath set adapter	Sets all device paths that are attached to an adapter	93
datapath set device	Sets the path of a device	94

datapath query adapter command

The **datapath query adapter** command displays information about a single adapter or all adapters.

Syntax

```
►►—datapath query adapter— adapter number———————————————————————►◄
```

Parameters

adapter number
The adapter number for which you want information displayed. If you do not enter an adapter number, information about all adapters is displayed.

Examples

If you enter the following command, **datapath query adapter**, the following output is displayed:

```
Active Adapters : 4
```

Adpt #	Adapter Name	State	Mode	Select	Errors	Paths	Active
0	Scsi Port2 Bus0	Normal	Active	1298	0	3	3
1	vscsi1	Normal	Active	404452	0	4	3
2	Scsi Port3 Bus0	Degrade	Active	32124	6	3	2
3	vscsi2	Failed	Offline	534	14	2	0

The terms used in the output are defined as follows:

Adpt #
The number of the adapter.

Adapter Name
The name of the adapter.

State The condition of the named adapter. It can be either:
Normal
Adapter is in use.
Degraded
One or more paths are not functioning.
Failed The adapter is no longer being used by Subsystem Device Driver.

Mode The mode of the named adapter, which is either `Active` or `Offline`.

Select The number of times this adapter was selected for input or output.

Errors The number of errors on a path that is attached to this adapter.

Paths The number of paths that are attached to this adapter.

Note: In the Windows NT host system, this is the number of physical and logical devices that are attached to this adapter.

Active The number of functional paths that are attached to this adapter. The number of functional paths is equal to the number of paths minus any that are identified as `failed` or `offline`.

datapath query adaptstats command

The **datapath query adaptstats** command displays performance information for all SCSI adapters that are attached to Subsystem Device Driver devices. If you do not enter a device number, information about all devices is displayed.

Syntax

```
►►—datapath query adaptstats— device number————————————————————————►◄
```

Parameters

device number
The device number refers to the device *index* number, rather than the Subsystem Device Driver device number.

Examples

If you enter the following command, **datapath query adaptstats 0**, the following output is displayed:

```
Adapter #:  0
=============
                Total Read  Total Write  Active Read  Active Write   Maximum
I/O:                  1442     41295166            0             2        75
SECTOR:             156209    750217654            0            32      2098

/*------------------------------------------------------------------------*/
```

The terms used in the output are defined as follows:

Total Read

- I/O: total number of completed Read requests
- SECTOR: total number of sectors that have been read

Total Write

- I/O: total number of completed Write requests
- SECTOR: total number of sectors that have been written

Active Read

- I/O: total number of Read requests in process
- SECTOR: total number of sectors to read in process

Active Write

- I/O: total number of Write requests in process
- SECTOR: total number of sectors to write in process

Maximum

- I/O: the maximum number of queued I/O requests
- SECTOR: the maximum number of queued sectors to Read/Write

datapath query device command

The **datapath query device** command displays information about a single device or all devices. If you do not enter a device number, information about all devices is displayed.

Syntax

Parameters

device number
The device number refers to the device *index* number, rather than the Subsystem Device Driver device number.

Examples

If you enter the following command, **datapath query device**, the output is displayed as follows:

```
DEV#:  35  DEVICE NAME: vpath0  TYPE: 2105E20   SERIAL: 60012028
===========================================================
Path#      Adapter/Hard Disk   State      Mode    Select        Errors
 0             scsi6/hdisk58    OPEN    NORMAL   7861147            0
 1             scsi5/hdisk36    OPEN    NORMAL   7762671            0
```

Note: Usually, the *device number* and the device *index* number are the same. However, if the devices are configured out of order, the two numbers are not always consistent. To find the corresponding index number for a specific device, you should always run the **datapath query device** command.

The terms used in the output are defined as follows:

Dev# The number of this device.

Name The name of this device.

Type The device product ID from inquiry data.

Serial The logical unit number (LUN) for this device.

Path The path number.

Adapter
The name of the adapter attached to the path.

Hard Disk
The name of the hard disk attached to the path.

State The condition of the named device:
- **Open** Path is in use.
- **Close** Path is not being used.
- **Dead** Path is no longer being used. It was either removed by the Subsystem Device Driver due to errors or manually removed using the **datapath set device M path N offline** or **datapath set adapter N offline** command.
- **Invalid** Path verification failed. The path was not opened.

Mode The mode of the named device. It is either `Normal` or `Offline`.

Select The number of times this path was selected for input or output.

Errors The number of errors on a path that is attached to this device.

datapath query devstats command

The **datapath query devstats** command displays performance information for a single Subsystem Device Driver device or all Subsystem Device Driver devices. If you do not enter a device number, information about all devices is displayed.

Syntax

```
►►—datapath query devstats— device number————————————————————————►◄
```

Parameters

device number
The device number refers to the device *index* number, rather than the Subsystem Device Driver device number.

Examples

If you enter the following command, **datapath query devstats 0**, the following output is displayed:

```
Device #:   0
=============
                Total Read  Total Write  Active Read  Active Write   Maximum
I/O:                   387     24502563            0             0        62
SECTOR:               9738    448308668            0             0      2098

Transfer Size:      <= 512        <= 4k       <= 16K        <= 64K     > 64K
                   4355850      1024164     19121140          1665       130

/*------------------------------------------------------------------------*/
```

The terms used in the output are defined as follows:

Total Read

- I/O: total number of completed Read requests
- SECTOR: total number of sectors that have been read

Total Write

- I/O: total number of completed Write requests
- SECTOR: total number of sectors that have been written

Active Read

- I/O: total number of Read requests in process
- SECTOR: total number of sectors to read in process

Active Write

- I/O: total number of Write requests in process
- SECTOR: total number of sectors to write in process

Maximum

- I/O: the maximum number of queued I/O requests
- SECTOR: the maximum number of queued sectors to Read/Write

Transfer size

- <= 512: the number of I/O requests received, whose transfer size is 512 bytes or less

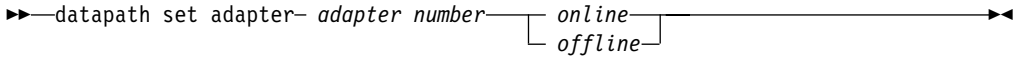
- <= 4k: the number of I/O requests received, whose transfer size is 4 KB or less (where KB equals 1024 bytes)
- <= 16K: the number of I/O requests received, whose transfer size is 16 KB or less (where KB equals 1024 bytes)
- <= 64K: the number of I/O requests received, whose transfer size is 64 KB or less (where KB equals 1024 bytes)
- > 64K: the number of I/O requests received, whose transfer size is greater than 64 KB (where KB equals 1024 bytes)

datapath set adapter command

The **datapath set adapter** command sets all device paths attached to an adapter either to online or offline.

Note: This command will not remove the last path to a device.

Syntax



Parameters

adapter number
The adapter number that you want to change.

online
Sets the adapter online.

offline
Sets the adapter offline.

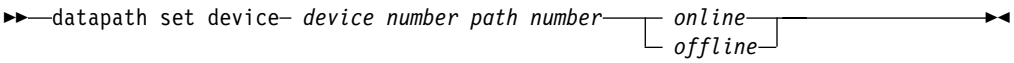
Examples

If you enter the following command, **datapath set adapter 0 offline**, adapter 0 changes to `Offline` mode and its state changes to `failed`; while all paths attached to adapter 0 change to `Offline` mode and their states change to `Dead`.

datapath set device command

The **datapath set device** command sets the path of a device either to online or offline.

Syntax



Parameters

device number
The device index number that you want to change.

path number
The path number that you want to change.

online
Sets the path online.

offline
Removes the path from service.

Note: You cannot remove the last path to a device from service. This prevents a data access failure from occurring.

Examples

If you enter the following command, **datapath set device 0 path 0 offline**, path 0 for dev 0 changes to `Offline` mode.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION ″AS IS″ WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Information Enabling Requests

Dept. M13
5600 Cottle Road
San Jose, CA.
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

AIX
Enterprise Storage Server
IBM
IBMLink
OS/400
RS/6000
Subsystem Device Driver

UNIX is a registered trademark of The Open Group in the United States and other countries

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

Other company, product, and service names may be trademarks or service marks of others.

INDEX

A

B

C

D

E

F

H

I

K

L

N

O

P

R

S

U

V

W

Program Number:

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.