

# Sun Enterprise Server Alternate Pathing 2.0.1 User's Guide

---



THE NETWORK IS THE COMPUTER™

**Sun Microsystems, Inc.**  
901 San Antonio Road  
Palo Alto, CA 94303-4900 USA  
650 960-1300  
Fax 650 969-9131

Part No.: 806-1164-10  
May 1999, Revision A

Send comments about this document to: [docfeedback@sun.com](mailto:docfeedback@sun.com)

Copyright 1999 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, California 94303-4900 U.S.A. All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third-party software, including font technology, is copyrighted and licensed from Sun suppliers .

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, SunStore, AnswerBook2, docs.sun.com, and Solaris are trademarks, registered trademarks, or service marks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-3(a).

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 1999 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, Californie 94303-4900 U.S.A. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, SunStore, AnswerBook2, docs.sun.com, et Solaris sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" SANS GARANTIE D'AUCUNE SORTE, NI EXPRESSE NI IMPLICITE, Y COMPRIS, ET SANS QUE CETTE LISTE NE SOIT LIMITATIVE, DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DES PRODUITS A REpondre A UNE UTILISATION PARTICULIERE OU LE FAIT QU'ILS NE SOIENT PAS CONTREFAISANTS DE PRODUITS DE TIERS.



# Contents

---

- 1. Introduction to Alternate Pathing 15**
  - The Purpose of Alternate Pathing 15
  - Basic Alternate Pathing Concepts 18
    - Physical Path 18
    - Metadisk 19
    - Metanetwork 20
    - Disk Pathgroup 21
    - Network Pathgroup 22
  - Supported Devices and Software Versions 23
  - Example AP Configurations 23
  - AP and Domains 25
  
- 2. Alternate Pathing Database 27**
  - Managing Copies of the Database 27
    - ▼ To Create a Copy of the AP Database 28
    - ▼ To Delete a Copy of the AP Database 29
  - Viewing Database Information 29
    - ▼ To View Information About Database Copies 29
  - Viewing Pathgroup Information 30
    - ▼ To View Uncommitted Disk Entries 30

- ▼ To View Committed Disk Entries 31
- ▼ To View Uncommitted Network Entries 31
- ▼ To View Committed Network Entries 32

### **3. Using Metadisks and Disk Pathgroups 33**

Device Nodes for Metadisks 33

Automatic Switching of Metadisks 34

Working With Disk Pathgroups and Metadisks 35

- ▼ To Create a Disk Pathgroup and Metadisk 35
- ▼ To Switch From the Primary Path to the Alternate Path 39
- ▼ To Switch Back to the Primary Path 41
- ▼ To Delete a Disk Pathgroup 42

### **4. Using AP Boot Devices 45**

Placing the Boot Disk Under AP Control 45

- ▼ To Place a Boot Disk under AP Control 45
- ▼ To Alternately Path a Mirrored Boot Disk 46
- ▼ To Remove a Mirrored Boot Disk From AP Control 47
- ▼ To Remove the Boot Disk From AP Control 47

AP Boot Sequence 48

Using Single-User Mode 48

### **5. Using Metanetworks and Network Pathgroups 51**

Metanetwork Interfaces 51

Working With Network Pathgroups 52

- ▼ To Create a Network Pathgroup 52
- ▼ To Switch an Ethernet or SunFDDI Pathgroup 55
- ▼ To Delete a Network Pathgroup 56

Alternately Pathing the Primary Network Interface 57

▼	To Alternately Path the Primary Network Interface	57
<b>6.</b>	<b>Interaction Between AP and DR</b>	<b>59</b>
	Using AP and DR Together	59
	Maintaining the Correct AP State	61



# Figures

---

- FIGURE 1-1 Alternately Pathed I/O Device 16
- FIGURE 1-2 Switching Paths After an I/O Controller Failure 17
- FIGURE 1-3 Switching Paths for a DR Detach Operation 18
- FIGURE 1-4 Physical Path 19
- FIGURE 1-5 Metadisk Example 20
- FIGURE 1-6 Metanetwork 21
- FIGURE 1-7 Disk Pathgroup 22
- FIGURE 1-8 Network Pathgroup 23
- FIGURE 1-9 Typical AP Configuration 24
- FIGURE 1-10 AP and Disk Mirroring 25





# Preface

---

This guide describes the Alternate Pathing (AP) software that runs on the Sun™ Enterprise™ server product line. Some AP features apply only to the Sun Enterprise 10000 server; these features are noted throughout this guide.

---

## Before You Read This Book

This manual is intended for the Sun Enterprise system administrator, who should have a working knowledge of UNIX® systems, particularly those based on the Solaris™ operating environment. If you do not have such knowledge, you should first read the Solaris User and System Administrator AnswerBook2™ documentation provided with this system, and consider UNIX system administration training.

---

## How This Book Is Organized

This document contains the following chapters:

**Chapter 1 “Introduction to Alternate Pathing”** introduces AP.

**Chapter 2 “Alternate Pathing Database”** covers the AP database operations.

**Chapter 3 “Using Metadisks and Disk Pathgroups”** describes metadisks and disk pathgroups, and explains how to use them.

**Chapter 4 “Using AP Boot Devices”** covers unattended system boot issues.

**Chapter 5 “Using Metanetworks and Network Pathgroups”** describes metanetworks and network pathgroups, and explains how to use them.

**Chapter 6 “Interaction Between AP and DR”** describes how Dynamic Reconfiguration (DR) and AP work together.

**Appendix A “AP Commands”** provides a list of all AP commands.

**Appendix B “AP Components”** provides an overview of the underlying AP architecture.

**Appendix C “Driver Layers”** provides an overview of the underlying AP drivers.

---

## Using UNIX Commands

This document does not contain information on basic UNIX commands and procedures such as shutting down the system, booting the system, and configuring devices.

See one or more of the following for this information:

- AnswerBook2 online documentation for the Solaris™ 2.x software environment, particularly those dealing with Solaris system administration.
- Other software documentation that you received with your system

---

# Typographic Conventions

TABLE P-1 Typographic Conventions

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

---

# Shell Prompts

Shell	Prompt
C shell	<i>machine_name%</i>
C shell superuser	<i>machine_name#</i>
Bourne shell and Korn shell	\$
Bourne shell and Korn shell superuser	#

---

## Related Documentation

Application	Title	Part Number
AP 2.0.1 Installation and Release Notes	<i>Sun Enterprise Server Alternate Pathing 2.0.1 Installation Guide and Release Notes</i>	805-7644-10
Reference (man pages)	<i>Sun Enterprise Server Alternate Pathing 2.0.1 Reference Manual</i>	806-1165-10
Other	<i>Sun Enterprise 6x00, 5x00, 4x00, 3x00 Dynamic Reconfiguration User's Guide</i>	806-0280-10
	<i>Sun Enterprise 10000 Dynamic Reconfiguration User's Guide</i>	805-7985-10
	<i>Sun Enterprise 10000 Dynamic Reconfiguration Reference Manual</i>	805-7986-10

---

## Ordering Sun Documentation

You can order Sun documentation from the SunStore<sup>SM</sup> web site at:

<http://sunstore.sun.com>

---

## Accessing Sun Documentation Online

The [docs.sun.com](http://docs.sun.com)<sup>SM</sup> web site enables you to access Sun technical documentation on the Web. You can browse the [docs.sun.com](http://docs.sun.com) archive or search for a specific book title or subject at:

<http://docs.sun.com>

---

# Sun Welcomes Your Comments

We are interested in improving our documentation and welcome your comments and suggestions. You can email your comments to us at:

`docfeedback@sun.com`

Please include the part number of your document in the subject line of your email.



# Introduction to Alternate Pathing

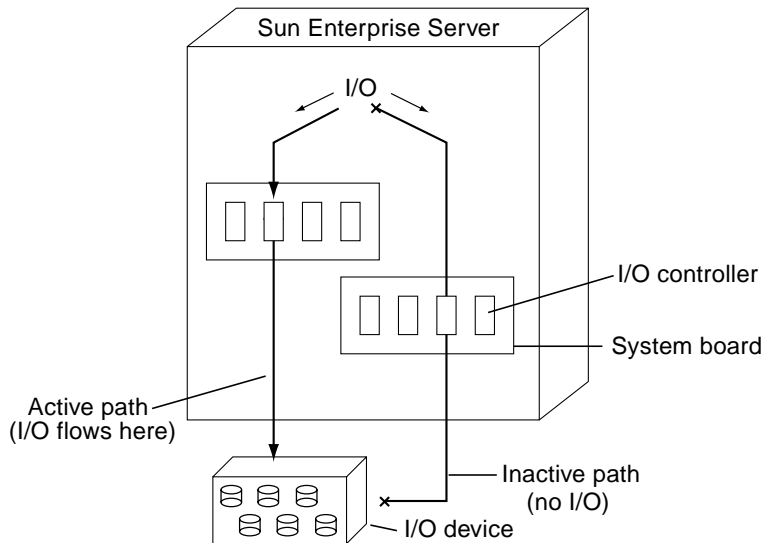
---

This chapter describes the basic purpose of Alternate Pathing and provides an overview of Alternate Pathing concepts and terminology.

---

## The Purpose of Alternate Pathing

Alternate Pathing (AP) supports high availability of I/O controllers—the hardware components that reside on system boards and enable the Sun Enterprise server to communicate with I/O devices such as disks and networks. With AP, each I/O device connects to *two* I/O controllers.

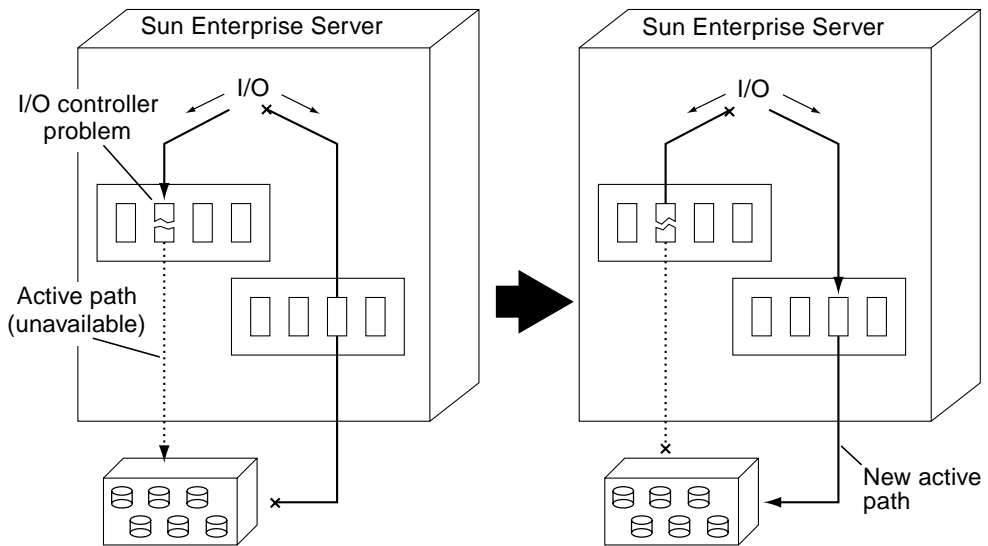


**FIGURE 1-1** Alternately Pathed I/O Device

The I/O controllers are part of two separate electrical pathways to the I/O device, known as *alternate paths*. AP enables you to set up and use alternate paths on the Sun Enterprise servers.

There are two purposes for AP. Perhaps the most obvious purpose is to help protect against I/O controller failures. With AP, if one I/O controller fails, you can switch to the alternate controller.





**FIGURE 1-2** Switching Paths After an I/O Controller Failure

For disk controllers, this switch occurs *automatically* whenever a path failure is detected during normal operation. For network controllers, you must manually switch paths (using a single AP command).

The second purpose of AP is to support Dynamic Reconfiguration (DR). DR is used to logically attach and detach system boards from the operating system without having to halt and reboot. For example, with DR you can detach a board from the operating system, physically remove and service the board, and then re-insert the board and attach it to the operating system again. You can do all of this without halting the operating system or terminating any user applications.

If you want to detach a board that is connected to an I/O device, and if that I/O device is alternately pathed, you can first use AP to redirect the I/O flow to a controller on a different board. You can then use DR to detach the system board without interrupting the I/O flow. On the Sun Enterprise 10000 server, the switch occurs automatically during the DR operation (for both disk and network devices), assuming a viable alternate controller exists on another board. FIGURE 1-3 shows the relationship between AP and DR.

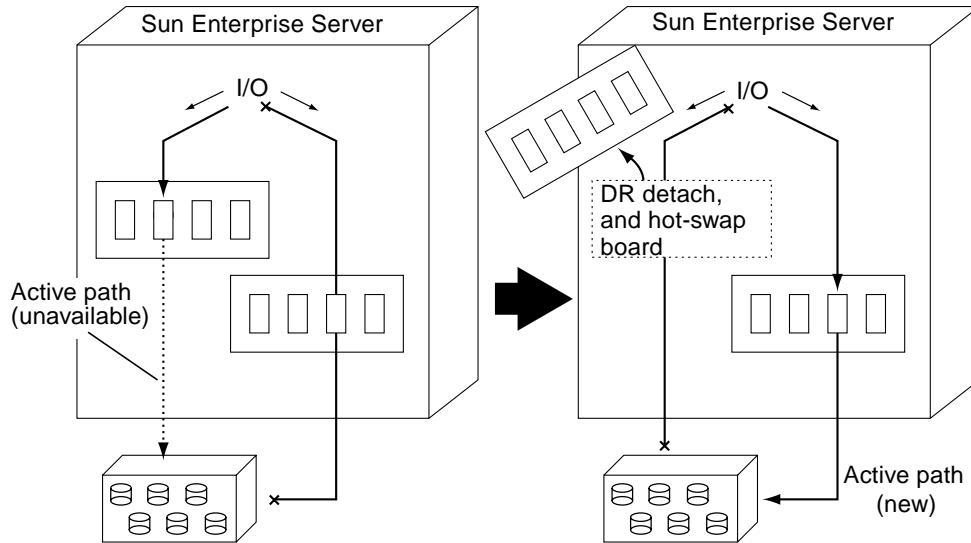


FIGURE 1-3 Switching Paths for a DR Detach Operation

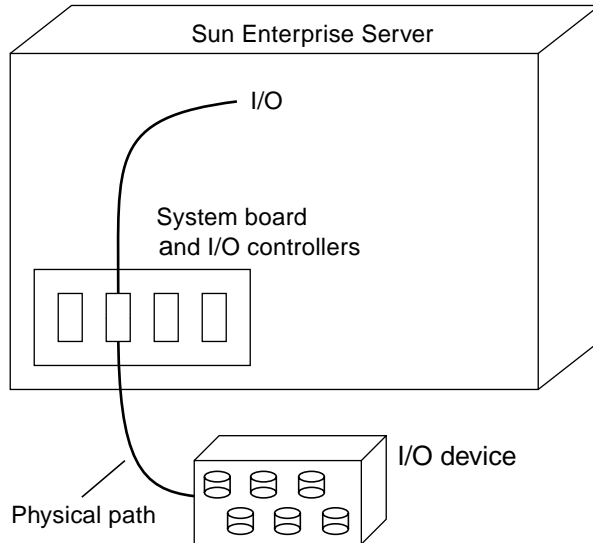
## Basic Alternate Pathing Concepts

This section discusses basic AP concepts and introduces the terminology that is used throughout this chapter.

### Physical Path

For the purposes of AP, an *I/O device* is either a disk or network. An *I/O controller* is the controller card for an I/O device. An *I/O port* is a connector on a controller card. (Usually there are two ports per controller card.) A *device node* is a path in the devices directory that is used to specify a physical device, for example, `/dev/dsk/c0t0d1s0`.

The term *physical path* refers to the electrical path from the host to a disk or network.



**FIGURE 1-4** Physical Path

You reference a physical device via a device node, for example, `/dev/dsk/c0t1d1s0`.

## Metadisk

A *metadisk*, as illustrated in FIGURE 1-5, is a construct that enables you to access a disk via either of two physical paths without having to reference either path explicitly within your scripts and programs. You reference a metadisk (in your scripts and programs) using an AP-specific device node such as `/dev/ap/dsk/mc0t1d1s0`. (See “Device Nodes for Metadisks” on page 33 for more information.)

In FIGURE 1-5, an AP-specific device node is used to perform disk I/O, regardless of which `p1n` port (`p1n2` or `p1n9`) is currently handling I/O.

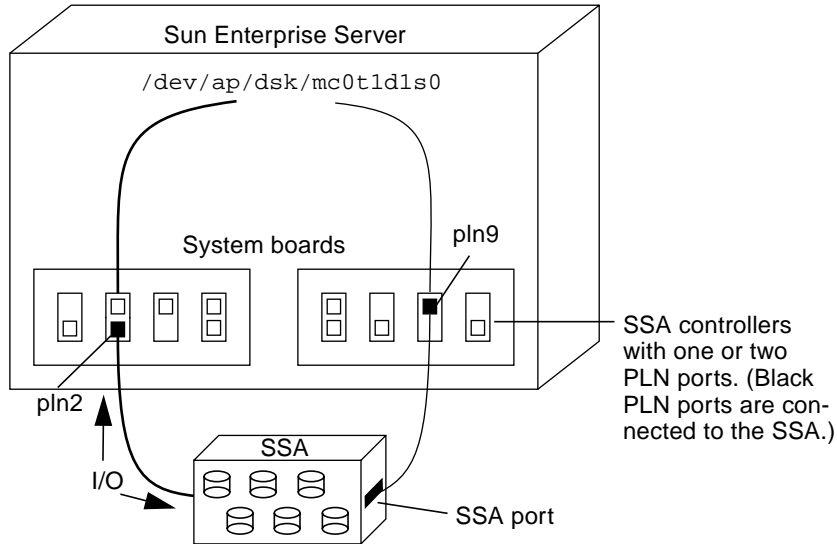


FIGURE 1-5 Metadisk Example

## Metanetwork

A *metanetwork*, as illustrated in FIGURE 1-6, is a construct that enables you to access a network via either of two physical paths without having to reference either path explicitly within your scripts and programs. You reference a metanetwork (in your scripts and programs) using a *metanetwork interface* name such as `m1e1`. (See “Metanetwork Interfaces” on page 51 for more information.)

In FIGURE 1-6, `m1e1` is used to access a metanetwork, regardless of which controller (1e1 or 1e6) is currently processing I/O for the metanetwork.

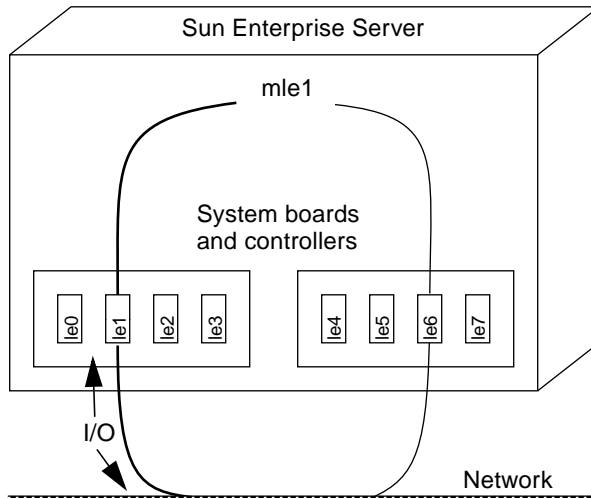


FIGURE 1-6 Metanetwork

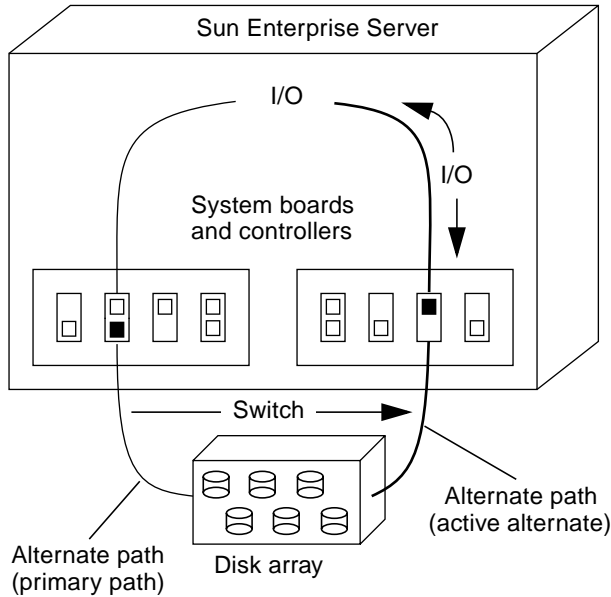
## Disk Pathgroup

A *disk pathgroup*, as illustrated in FIGURE 1-7, consists of two physical paths leading to the same disk array. When a physical path is part of a pathgroup, it is called an *alternate path*. An alternate path to a disk can be uniquely identified by the `pln` port or `sf` port that the alternate path uses. Only one alternate path at a time handles disk I/O. The alternate path that is currently handling I/O is called the *active alternate*.

Note that whereas a metadisk provides a means for you to access a *disk* (in your scripts and programs), a disk pathgroup provides a means for you to manipulate the *path* to that disk (when you run AP commands). For example, to perform a *switch* operation (that is, change the active alternate from one alternate path to another), you reference a disk pathgroup within an `apconfig(1M)` command.

One of the alternate paths is designated as the *primary path*. The primary path is initially the active alternate. Although the active alternate changes when you perform a switch operation, the primary path remains constant. You reference a disk pathgroup by specifying the `pln` port (for example, `pln1`) or `sf` port (for example, `sf1`) that corresponds to the primary path. (For information about determining the `pln` or `sf` port name, see “Device Nodes for Metadisks” on page 33.)

For example, FIGURE 1-7 shows the results of using the `apconfig` command to switch the active alternate of a disk pathgroup.



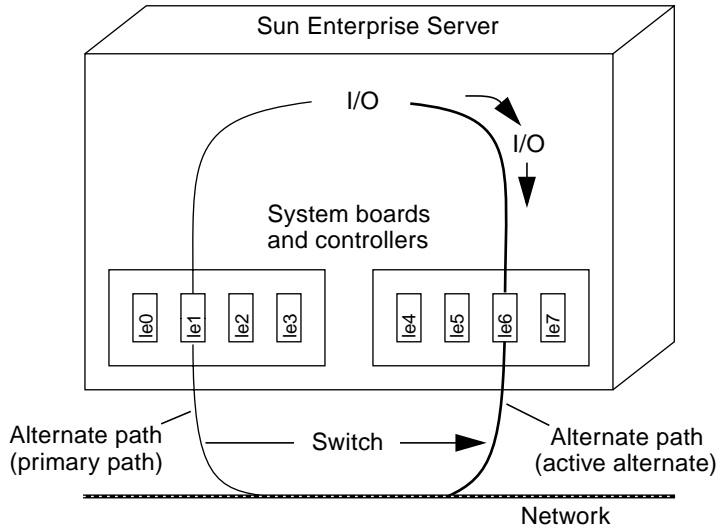
**FIGURE 1-7** Disk Pathgroup

You reference a disk pathgroup (for example, to switch from one path to another) by specifying the primary path, for example, `apconfig-P pln2-apln9`.

## Network Pathgroup

A *network pathgroup*, as illustrated in , consists of two network controllers connected to the same physical network. The terms *alternate path*, *active alternate*, *primary path*, and *switch* have basically the same meaning as they do for disk pathgroups.

To specify a network pathgroup, reference the corresponding metanetwork interface name, for example, `mle1`. (Metanetwork interface names are described in “Metanetwork Interfaces” on page 51.) For example, shows the results of using the `apconfig(1M)` command to switch the active alternate of a network pathgroup.



**FIGURE 1-8** Network Pathgroup

You reference a network pathgroup (for example, to switch from one path to another) by specifying the metanetwork interface, for example, `apconfig -P mle1 -a le6`.

---

## Supported Devices and Software Versions

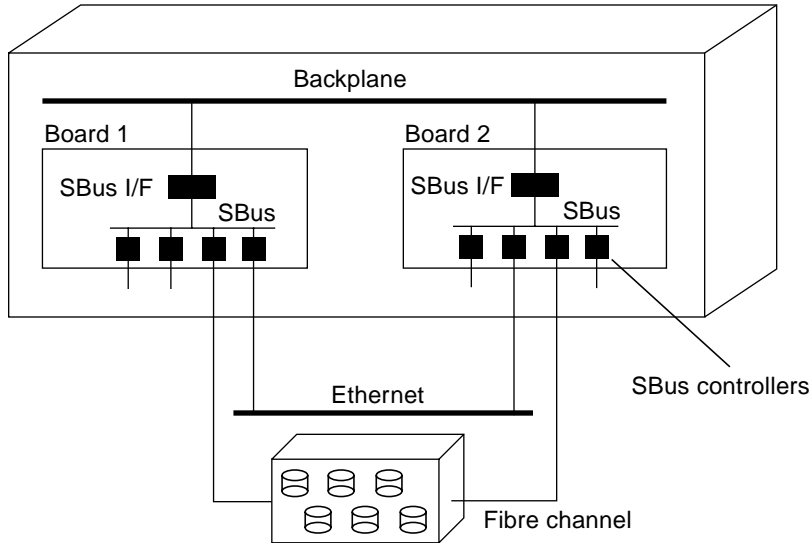
AP 2.0.1 supports Solaris 2.5.1 only.

AP supports SSAs attached to `pln` ports on SSA controllers, as well as Photons attached to `sf` ports. For a list of the network devices supported by AP, see the release notes for this product.

---

## Example AP Configurations

FIGURE 1-9 shows how you can use AP to support an Ethernet network and a disk array.

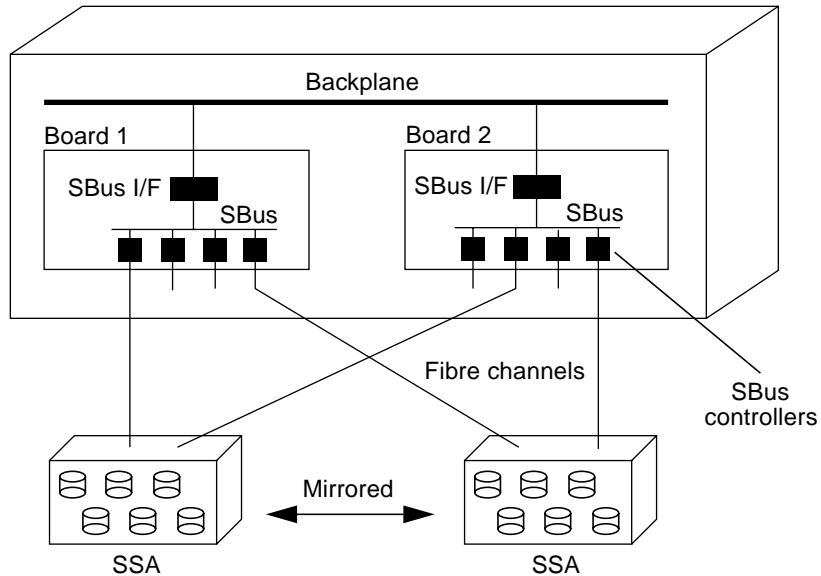


**FIGURE 1-9** Typical AP Configuration

In this example, two network controllers—one each on Board 1 and Board 2—are connected to the same network. Similarly, two SSA controllers on the two boards are connected to the same SSA. In this situation, if Board 1 is detached with a DR Detach operation, AP can switch usage to Board 2 without interfering with any I/O operations that may be in progress.

AP is not equivalent to disk mirroring. Disk mirroring achieves *data* redundancy. AP achieves *pathing* redundancy. Disk mirroring and AP are complementary; you can use them together to achieve both data redundancy and pathing redundancy. To do this, you must configure your volume manager software (Sun Enterprise Volume Manager or Veritas Volume Manager) so that it uses the AP metadisk paths.





**FIGURE 1-10** AP and Disk Mirroring

This type of configuration enables you to switch the paths used to implement the mirror from one board to another board, without disrupting the disk mirroring or any active I/O.

---

## AP and Domains

The Sun Enterprise 10000 server supports Dynamic System Domains, or simply “domains.”

AP cannot be used across two domains. For example, suppose a board contains a controller that is part of a pathgroup, and you move that board into a different domain using DR. (You can do this only if the alternate path on that board is not currently active.) In this case, you can no longer switch to the alternate path on that board.



## Alternate Pathing Database

---

This chapter describes how to create and manage the AP database which maintains the state of the AP configuration.

---

### Managing Copies of the Database

AP maintains a database that contains information about all defined metadisks and metanetworks, and their corresponding alternate paths and properties. One set of data is maintained for each domain on the Sun Enterprise 10000 server. On other Sun Enterprise servers, one set of data is maintained for the entire machine. However, you should set up multiple copies of the database. In this way, if a given database copy is not accessible or becomes corrupted, AP can automatically begin to use a current, non-corrupted database copy.

You must dedicate an entire disk partition, one that has at least 300 Kbytes, to each database copy. You can use larger partitions, but doing so wastes disk space. Keep the following information in mind when selecting disk partitions for the AP database:

- You should set up three to five database copies.
- As configured at the factory, partition four of the root disk is appropriately sized for the AP database and is not allocated for any other purposes. This partition is a good choice for an AP database copy, assuming you are not using it for other purposes.
- The database copies should have no I/O controllers in common with each other. Following this rule allows maximum availability in case of controller failure.
- If you have configured your system to make use of Dynamic Reconfiguration (DR), the database copies should be hosted by I/O controllers on different system boards so that a database copy is accessible if one of the system boards is detached.

- If you want to place an AP database copy in a partition of an alternately pathed disk, create two copies of the database using each of the *physical paths* utilized by the AP metadisk to access the partition. AP behaves as if two copies of the database exist, when actually, there is only one, since the disk is accessible via two paths. The behavior does not result in database inconsistencies, however, since AP always updates and accesses database copies sequentially. This behavior does not result in performance problems since the AP database is not accessed frequently.

On the Sun Enterprise 10000 server, a subset of the information in the AP database is automatically maintained on the SSP for use at boot time. This database contains AP information for the boot disk.

---

**Note** – The following AP command examples assume that your command search path includes the directory where the commands are installed. See “Using Single-User Mode” on page 48.

---

## ▼ To Create a Copy of the AP Database

- Use `apdb(1M)` as follows:

```
# apdb -c /dev/rdisk/c0t1d0s4 -f
```

The `-c` option specifies the raw disk slice (under `/dev/rdisk`) where you want to create the database copy. You must dedicate an entire disk partition to each database copy. The disk partition must have at least 300 Kbytes.

The `-f` (force) option is only necessary to create the first AP database copy.



---

**Caution** – Do not create an AP database using a metadisk path (rather than a physical path to the disk slice). If you use a metadisk path, the `apdb(1M)` command will succeed, but you may experience unpredictable behavior at a later point in time. If you want to place an AP database in a partition of an alternately pathed disk, create two copies of the database using each of the physical paths utilized by the AP metadisk to access the partition.

---

## ▼ To Delete a Copy of the AP Database

- Use `apdb(1M)` as follows:

```
# apdb -d /dev/rdisk/c0t1d0s4 -f
# apconfig -D
#
```

The `-d` option specifies the raw disk slice (under `/dev/rdisk`) where the copy of the database that you want to delete is located.

The `-f` (force) option is only necessary to delete the second-to-last copy and the last copy of the database.

In this example, `apconfig -D` is used after the deletion operation to view information about the existing AP database copies. Since no information is returned, the `apdb(1M)` command must have deleted the last database copy.

---

## Viewing Database Information

You can easily view information in the database, including information about the database copies, the disk entries within the database, and the network entries within the database.

## ▼ To View Information About Database Copies

- Use `apconfig -D` as follows:

```
# apconfig -D

path: /dev/rdisk/c0t1d0s4
major: 32
minor: 12
timestamp: Thu Jul 27 16:24:27 1995
checksum: 687681819
corrupt: No
inaccessible: No
```

In this example, there is only one AP database. The command shows the path to this database, along with its major number, minor number, timestamp, and checksum. The `corrupt` field indicates whether the database is corrupt. (If `corrupt` is set to `Yes`, the data did not validate properly against the checksum.) The `inaccessible` field indicates whether the device that holds the database can be accessed.

---

## Viewing Pathgroup Information

The AP database contains information about disk and network pathgroups. When a pathgroup is initially defined (as described in Chapter 3 “Using Metadisks and Disk Pathgroups” and Chapter 5 “Using Metanetworks and Network Pathgroups”), its pathgroup definition is an *uncommitted* database entry. The metadisk or metanetwork associated with an uncommitted entry is not available until the pathgroup definition is *committed*. Conversely, when a pathgroup definition is deleted, the deletion must be committed before it takes effect. The two states (uncommitted and committed) enable you to review the effects of an operation before allowing the operation to proceed. To commit the uncommitted database entries, use `apdb -C`. Note that uncommitted entries remain in the database indefinitely, until you either commit them or remove them.

### ▼ To View Uncommitted Disk Entries

- Use `apconfig(1M)` with the `-S` and `-u` options as follows, where `S` stands for **SCSI** and `u` stands for **uncommitted**:

```
# apconfig -S -u

c1      sf0  P A
c2      sf1
        metadiskname(s):
                mc1t5d0  U
                mc1t4d0  U
                mc1t3d0  U
                mc1t2d0  U
                mc1t1d0  U
                mc1t0d0  U
```

For more information see Chapter 3 “Using Metadisks and Disk Pathgroups”.

## ▼ To View Committed Disk Entries

- Use `apconfig(1M)` with the `-S` option, as follows, where `S` stands for SCSI:

```
# apconfig -S

c1      pln0  P A
c2      pln1
        metadiskname(s):
                mc1t5d0  R
                mc1t4d0
                mc1t3d0
                mc1t2d0
                mc1t1d0
                mc1t0d0
```

For more information see Chapter 3 “Using Metadisks and Disk Pathgroups”.

## ▼ To View Uncommitted Network Entries

- Use `apconfig(1M)` with the `-N` and `-u` options, as follows, where `N` stands for network and `u` stands for uncommitted:

```
# apconfig -N -u

metanetwork:    mle0  U
physical devices:
                le2
                le0  P A
```

For more information see Chapter 5 “Using Metanetworks and Network Pathgroups”.

## ▼ To View Committed Network Entries

- Use `apconfig(1M)` with the `-N` option, as follows:

```
# apconfig -N  
  
metanetwork:    mle3  
physical devices:  
                 le4  
                 le3 P A
```

For more information see Chapter 5 “Using Metanetworks and Network Pathgroups”.



## Using Metadisks and Disk Pathgroups

---

You can create metadisks and disk pathgroups only for disks that are accessible via two paths. You should generally use two separate controllers on different system boards.

---

**Note** – AP does not modify a disk's data when that disk is placed under AP control or when a pathgroup is deleted (except for the data on the slices that contain AP database copies). AP does not repartition a disk. If a pathgroup is deleted, you can continue to access the data by using its physical device name.

---

---

## Device Nodes for Metadisks

Here are two examples of *physical* device nodes for disk devices under Solaris 2.x:

- `/dev/dsk/c0t0d0s0`
- `/dev/rdisk/c0t0d0s0`

where:

`c` references the I/O port on the host (not the disk array)

`t` is the bus within the disk array

`d` is the target ID of the disk on that bus

`s` is the slice number on the disk

These physical device nodes represent a particular physical path to a partition on a disk.

Each controller port has both a port number (such as `c0`) and a port name (such as `pln2` or `sf3`). The port name consists of the port type and *instance* number. See `/etc/path_to_inst` for more information.

When a disk array is connected to two ports, it can potentially be accessed from either path via the physical device node, for example, `/dev/dsk/c0t0d0s0` or `/dev/dsk/c1t0d0s0`.

The device node for a metadisk is derived from the physical device node of the primary path for a pathgroup. Here are two examples of metadisk device nodes:

- `/dev/ap/dsk/mc0t0d0s0`
- `/dev/ap/rdisk/mc0t0d0s0`

As you can see, an `ap` directory has been added, and an `m` (for “meta”) is prepended to the device specification. The device node for a metadisk has the ability to access the underlying physical disk drive from multiple paths.

---

## Automatic Switching of Metadisks

Metadisks can be automatically switched from the active path to the alternate path in two situations:

- The active path fails.
- The board containing the controller for the active path is detached by using a DR Detach operation. (Automatic switching during a DR detach is available only on the Sun Enterprise 10000 server.)

When the active path fails, an automatic switch is attempted only if an alternate path is available. The failed path is then marked unavailable, or *tried*. You can identify the tried paths with `apconfig -S`:

```
# apconfig -S

c1      pln0  P A
c2      pln1  T
        metadiskname(s):
            mc1t5d0
            mc1t4d0
            mc1t3d0
            mc1t2d0
            mc1t1d0
            mc1t0d0
```

In this example, the currently inactive path, `p1n1`, is marked with a “T”, which indicates that path was tried but failed.

The tried flag is only significant for automatic switch operations (not for manual switch operations). AP never attempts to automatically switch to a tried path. (This prevents thrashing in the case that both paths may have failed.)

You can reset the tried flag with any of the following actions:

- Reboot the corresponding domain.
- Perform a DR detach followed by a DR attach of a board that contains the controller marked as tried.
- Manual reset the tried flag for a particular controller.

You can manually reset the tried flag as shown in this example:

```
# apdisk -w p1n1
```

In this example, `p1n1` is a controller with the tried flag set to true. The `apdisk -w` feature should be used judiciously. This command merely clears the tried flag; it does not address any potential problems with the controller or device. This command should only be used in situations where the failed path has been restored without an intervening DR operation or reboot.

Note that you can attempt to perform a manual switch to a path that is marked as tried.

---

## Working With Disk Pathgroups and Metadisks

---

**Note** – The example commands in this section use `p1n` ports (for SSA disk arrays). If you have Sun™ StorEdge™ A5000 disk arrays, you would simply specify `sF` ports wherever `p1n` ports are shown.

---

### ▼ To Create a Disk Pathgroup and Metadisk

1. Decide which two ports will make up the alternate paths for the pathgroup.

- a. You can use the `apinst(1M)` command to display all ports (for example, `pln0` and `pln1`) and their disk device nodes (for example, `/dev/dsk/c1t0d0`):

```
# apinst
pln0
    /dev/dsk/c1t0d0
    /dev/dsk/c1t1d0
    /dev/dsk/c1t2d0
    /dev/dsk/c1t3d0
    /dev/dsk/c1t4d0
    /dev/dsk/c1t5d0
pln1
    /dev/dsk/c2t0d0
    /dev/dsk/c2t1d0
    /dev/dsk/c2t2d0
    /dev/dsk/c2t3d0
    /dev/dsk/c2t4d0
    /dev/dsk/c2t5d0
```

- b. You must know your system hardware configuration to recognize when two ports are connected to the same disk array.

In this example, it is assumed that the SSA contains six disks and two SSA ports. One SSA port is connected to `pln` port `c1`, and the other SSA port is connected to `pln` port `c2`.

2. Use `apdisk(1M)` with the `-c`, `-p`, and `-a` options to create an uncommitted disk pathgroup:

```
# apdisk -c -p pln0 -a pln1
```

where:

- p specifies the primary path
- a specifies the alternate path
- c specifies that this information is to be created.

This `apdisk(1M)` command creates a metadisk name, as well as all of the necessary information in the AP database for maintaining the two alternate paths for all six disks.

### 3. Verify the results:

```
# apconfig -S -u

c1      pln0  P A
c2      pln1
        metadiskname(s):
            mc1t5d0  U
            mc1t4d0  U
            mc1t3d0  U
            mc1t2d0  U
            mc1t1d0  U
            mc1t0d0  U
```

The `apconfig -S -u` command lists the uncommitted metadisks. `-S` lists SCSI devices only (that is, disks rather than networks). `-u` lists uncommitted devices only. The `U` next to each metadisk name indicates that the metadisk entry is uncommitted.

The `P` next to `pln0` indicates that `pln0` is the primary path, and the `A` indicates that `pln0` is the active alternate. Although you can change the active alternate, the primary path always remains constant. The significance of the primary path is that it is initially the active alternate, it is used when the metadisk is named, and it is used to identify the metadisk. In this case, `c1t0d0` from the primary path name becomes part of `mc1t0d0` in the metadisk name.

### 4. If you are satisfied with the results shown in the previous step, use `apdb(1M)` with the `-C` option to commit the uncommitted database entries:

```
# apdb -C
```

`apconfig -S` lists the committed SCSI entries in the database. As shown in the example, this listing is exactly the same as the previous listing, except the `U` no longer appears after the metadisk names, indicating that the metadisks are no longer uncommitted.

5. Before continuing, you may want to verify the results by using `apconfig -S` to view the committed disk entries in the database:

```
# apconfig -S

c1      pln0  P A
c2      pln1
        metadiskname(s):
            mc1t5d0
            mc1t4d0
            mc1t3d0
            mc1t2d0
            mc1t1d0
            mc1t0d0
```

6. Use `drvconfig(1M)`:

```
# drvconfig -i ap_dmd
```

`drvconfig` rebuilds the devices directory, which represents the device tree in the kernel. The AP disk metadrivers are pseudo-devices.

7. Use the following command to verify the results:

```
# ls /devices/pseudo/ap_dmd*
/devices/pseudo/ap_dmd@0:128,blk
/devices/pseudo/ap_dmd@0:128,raw
/devices/pseudo/ap_dmd@0:129,blk
/devices/pseudo/ap_dmd@0:129,raw
/devices/pseudo/ap_dmd@0:130,blk
/devices/pseudo/ap_dmd@0:130,raw
...
```

As you can see from the listing, `drvconfig` created minor nodes for the alternately pathed device.

8. Use `apconfig(1M)` with the `-R` option to create symbolic links from the devices directories, `/dev/ap/dsk` and `/dev/ap/rdsk`, to the metadisk special files under `/devices/pseudo`:

```
# apconfig -R
```

9. Use the following command to view the symbolic links and verify the results:

```
# ls -l /dev/ap/dsk
total 8
lrwxrwxrwx 1 root          40 Jul 27 16:47 mclt0d0s0 ->
    ../../../../devices/pseudo/ap_dmd@0:128,blk
lrwxrwxrwx 1 root          40 Jul 27 16:47 mclt0d0s1 ->
    ../../../../devices/pseudo/ap_dmd@0:129,blk
lrwxrwxrwx 1 root          40 Jul 27 16:47 mclt0d0s2 ->
    ../../../../devices/pseudo/ap_dmd@0:130,blk
```

The device nodes that you need—under `/dev/ap/dsk` as well as `/dev/ap/rdisk`—are now ready to be used.

10. **Modify every reference that uses a physical device node (that is, a path that begins with `/dev/dsk` or `/dev/rdisk`) to use the corresponding metadisk device node (that is, a path that begins with `/dev/ap/dsk` or `/dev/ap/rdisk`).**

If a partition is currently mounted under a physical path name, it should be unmounted and remounted under the metadisk path name.

If you do not want to unmount a partition, perhaps because it is heavily used, you could delay placing the partition under AP control until you are ready to bring the system down for maintenance and then reboot. In this scenario, you would modify the `/etc/vfstab` file so that when the system is rebooted, the partition comes up under an AP device. (If you are placing the boot disk under AP control, you would also need to modify `/etc/vfstab` using `apboot(1M)` as described in Chapter 4 “Using AP Boot Devices”.)

## ▼ To Switch From the Primary Path to the Alternate Path

---

**Note** – You can perform a switch at any time, even while I/O is occurring on the device. You might want to experiment with the switching process to verify that you understand it and that your system is setup properly, rather than wait until a critical situation occurs.

---



---

**Caution** – When you switch paths, AP does not check to verify that data can be transferred over the path to which you are switching (although it does determine whether or not that path is detached or offline). You may want to verify the status of the path before switching to it by performing an I/O operation such as

prtvto(1M). AP does not produce any error or warning messages if you switch to a path that is not functioning properly. If you switch to a non-functioning path for your boot disk, your system may crash if the path is not switched back immediately.

---

**1. Use the `apconfig -S` command to view the current configuration:**

```
# apconfig -S

c1          pln0  P A
c2          pln1
           metadiskname(s):
                mc1t5d0
                mc1t4d0
                mc1t3d0
                mc1t2d0
                mc1t1d0
```

In this example, `pln0` is the active alternate since it is followed by an `A`. (It is also the primary path, since it is followed by a `P`.)

**2. To perform the switch, use `apconfig(1M)` with the `-P` and `-a` options:**

```
# apconfig -P pln0 -a pln1
```

`-P` specifies the primary path and thereby *identifies the pathgroup* for which you want to change the active alternate. Thus, `-P pln0` in the example above identifies the pathgroup for which `pln0` is the primary path. `-a` specifies the alternate that you want to make active.



3. You can verify the results using `apconfig(1M)` with the `-S` option to view the committed metadisks in the database:

```
# apconfig -S

c1          pln0  P
c2          pln1  A
            metadiskname(s):
                mc1t5d0
                mc1t4d0
                mc1t3d0
                mc1t2d0
                mc1t1d0
```

The active alternate has been switched to `pln1`.

Note that you do not have to commit a switch operation.

## ▼ To Switch Back to the Primary Path

- You can switch back to the primary path using the following commands:

```
# apconfig -P pln0 -a pln0
# apconfig -S

c1          pln0  P A
c2          pln1
            metadiskname(s):
                mc1t5d0
                mc1t4d0
                mc1t3d0
                mc1t2d0
                mc1t1d0
```

The first `apconfig` command, above, switches the active alternate for the pathgroup that has the primary controller `pln0`. The active alternate becomes `pln0`.

## ▼ To Delete a Disk Pathgroup

---

**Note** – All usage of the metadisks in a pathgroup must be discontinued (for example, the file systems must be unmounted) before you delete the pathgroup.

---

1. Use `apdisk(1M)` with the `-d` option to specify the primary path:

```
# apdisk -d pln0
```

2. To verify the results, use `apconfig(1M)` with the `-S` option to view the committed SCSI disk entries in the database:

```
# apconfig -S

c1                pln0  P A
c2                pln1
                  metadiskname(s):
                    mc1t5d0  D
                    mc1t4d0  D
                    mc1t3d0  D
                    mc1t2d0  D
                    mc1t1d0  D
                    mc1t0d0  D
```

If the pathgroup was not previously committed, the `apdisk -d` command deletes it from the database. However, if the pathgroup was previously committed, the `apdisk -d` command simply marks it as deleted, but the deletion is not completed until the next time you commit the entries in the database. In the example above, the `pln0` pathgroup was previously committed, so the letter `D` indicates that it is marked for deletion.

3. Use `apdb(1M)` as follows to commit the database entries, thereby completing the deletion:

```
# apdb -C
```

4. You can verify the deletion with `apconfig -S`:

```
# apconfig -S
```

---

**Note** – You can undo a deletion if the deletion is uncommitted. To undo a deletion, use `apdisk -z`, specifying the same port that you previously specified.

---



## Using AP Boot Devices

---

This chapter describes how you can alternately path the boot disk.

---

### Placing the Boot Disk Under AP Control

To allow for unattended system boot, even if the controller for the boot disk fails, you can place your the boot disk under AP control.

#### ▼ To Place a Boot Disk under AP Control

**1. Create an AP pathgroup for the boot disk.**

This process is described in **Chapter 3 “Using Metadisks and Disk Pathgroups”**.

**2. Use `apboot(1M)` to define the new AP boot device.**

`apboot` modifies `/etc/vfstab` and `/etc/system`. For example:

```
# apboot mc2t0d0
```

where `mc2t0d0` is the metadisk name of the boot disk. `apboot` examines `/etc/vfstab` and replaces the physical device name of the disk (for example, `/dev/dsk/c2t0d0*` or `/dev/dsk/c1t0d0*`) with the metadisk name (for example, `/dev/dsk/mc2t0d0*`). It also edits `/etc/system` so that the kernel drivers required for AP boot disk usage are loaded at the proper time.

Do not manually replace the physical devices in `/etc/vfstab` with metadisks for the boot disk. Instead, use `apboot` to ensure that all required changes are made.

3. Set the OpenBoot™ PROM (OBP) devalias variable `boot-device` to the physical path most likely to be used for booting.

For example:

```
# setenv boot-device \ /sbus@68,0/SUNW,soc@0,0/  
SUNW,pln@a0000000,78cab4/ssd@0,2
```

4. Define a devalias for the alternate boot device path as a convenience in case you need to perform a manual boot.

At this point, reboot the system to begin using the AP boot device.

Normally, the file systems that are mounted as part of the boot process are split across two separate disks (because of disk space requirements). If you place the boot disk under AP control (using `apboot(1M)`), you must manually edit the `/etc/vfstab` file to also place other file systems that are mounted during the boot process under AP control. In the `/etc/vfstab` file, you must change the `device` to mount and `device` to `fsck` paths for all other mount points that you want to place under AP control.

If you want to create a new AP database copy *after* you have placed the boot disk under AP control, and that database copy is to be located on a partition controlled by a controller port that does not control any of the current AP database partitions, you must first remove the boot disk from AP control. Make sure that the new AP database has been created. Then, place the boot disk under AP control again. Failure to follow this procedure may cause the database to become inaccessible during boot.

## ▼ To Alternately Path a Mirrored Boot Disk

Mirroring the boot disk is primarily a function of your disk management software. The purpose of this procedure is to notify AP about a mirrored boot disk. When you use mirrored boot disks that are also alternately pathed, you have four potential physical paths to the boot disk, two for each side of the mirror. (At least, this is the recommended configuration if you want to maximize protection against controller failure.) Once you perform the following procedure, AP will assure that the appropriate physical path is active, even if you boot using a different boot device path than the one that was active previously. Furthermore, on the Sun Enterprise 10000 server (which supports automatic switching of the boot device path at boot time), all four paths are available as alternates should an automatic switch be required at boot time.

1. Place the boot disk under AP control, as described in “To Place a Boot Disk under AP Control”, above.

2. Create an AP pathgroup for the mirror of the boot disk.

This process is described in Chapter 3 “Using Metadisks and Disk Pathgroups”.

3. Notify AP about the boot disk mirror. For example:

```
# apboot -m mc3t0d0
```

In this example, `mc3t0d0` is the metadisk for the mirror of the boot disk.

4. Create a mirror of your boot disk (using the two metadisks) with your disk management software.

## ▼ To Remove a Mirrored Boot Disk From AP Control

- Use `apboot(1M)` to undefine the AP mirrored boot device, for example:

```
# apboot -u mc3t0d0
```

## ▼ To Remove the Boot Disk From AP Control

- Use `apboot(1M)` to specify an appropriate physical device node, for example:

```
# apboot c2t0d0
```

In the above command, `c2t0d0` is the physical device node of an alternate path for the boot disk (as currently specified in `/etc/vfstab`). `apboot` also edits the `/etc/system` file to remove force loading of AP kernel driver modules, since they are no longer needed when the boot disk is not an AP device.



---

**Caution** – If you place the boot disk under AP control, and later decide to remove the AP package (using `pkgrm(1M)`), you must first use `apboot(1M)` to remove the boot disk from AP control. If you do not first remove the boot disk from AP control, the system on that disk becomes unbootable.

---

---

## AP Boot Sequence

This subsection briefly describes the flow of events that occur when the Sun Enterprise 10000 server is booted on an alternately pathed boot disk. This sequence of events illustrates how auto switching of the boot disk controller is achieved during the boot process, if such a switch is necessary. The boot sequence proceeds as follows:

1. By default, the system is booted from the device specified by the OBP devalias `boot-device`. Note that this device may be different from the last active alternate for the boot disk.
2. OBP stores the path to the boot disk on the SSP.
3. If a failure occurs, it is detected after about one minute. Then, the AP Reboot Host program is executed.

---

**Note** – One or two minutes may pass before action is taken, so do not immediately enter new commands if you notice that the boot process has failed. If you attempt a manual recovery from a boot failure, be aware that the automatic reboot recovery process will still be executing and may override your manual recovery command.

---

4. AP Reboot Host retrieves the path stored earlier by OBP, and communicates the path to the AP SSP daemon.
5. The AP SSP daemon looks up the alternate path for the boot disk in the AP SSP database, and retries the boot process with the other alternate path.
6. After the reboot succeeds, AP determines the alternate from which the system was booted, and makes it the active alternate.

---

## Using Single-User Mode

Normally, when the Sun Enterprise server is fully booted, you use the AP command versions located in `/usr/sbin`. However, if your server comes up in single-user mode because the boot process did not fully complete, you can use the commands in `/sbin`. The command versions under `/sbin` do not rely on the AP daemon services (which are not available in single-user mode). If the system enters single-user mode because of a problem related to AP, you may be able to resolve the problem by using the `/sbin` commands to perform needed AP operations.



Two AP-related problems may cause the system to come up in single-user mode:

- If two paths are supposed to lead to the same disk (according to the AP SSP database) but those paths actually lead to different disks, and if that disk needs to be mounted during the boot process. (This can only happen if you change the physical configuration of the pathgroup without running AP commands to update the database.)
- If an active alternate for a disk turns out to be inaccessible and that disk is required during the boot process. A disk is required during the boot process if it has file systems that are mounted during the boot process (that is, it has entries in the `/etc/vsftab` file).

These situations arise only with respect to disks, not networks. In either case, you may be able to use the AP commands under `/sbin` to resolve the problem.



## Using Metanetworks and Network Pathgroups

---

To use AP, both physical networks within a network pathgroup must be of the same type. For example, a network pathgroup could consist of two `le` networks or two `qe` networks, but not one of each.

Both alternates in a network pathgroup should be physically connected to the same network. For example, Ethernet controllers should be connected to the same subnet.

While multiple physical network connections exist, only one controller at a time is active. The controllers should be on different system boards so that DR operations (such as DR Detach) can be performed without affecting all potential active alternates.

The AP switch procedures in this section show how to switch the active alternate.

---

## Metanetwork Interfaces

A metanetwork interface name is derived from the name of the primary alternate for that metanetwork. A metanetwork interface name has the form `mxxx`, where `xxx` is the primary interface name such as `le0`. Some examples follow:

- SCSI-2/Buffered Ethernet FSBE/S and DSBE/S (LE Ethernet) metanetwork names have the form `mle#`, where `#` is the instance number. For example, assume the network controllers `le0` and `le1` connect to the same Ethernet network. A metanetwork `mle0` could encompass these two controllers (if the primary controller is `le0`). Similarly, Sun GigabitEthernet (QE Ethernet) metanetwork names have the form `mqe#`. Note that you cannot mix `le` and `qe` networks within the same pathgroup.
- SunFDDI™ metanetwork names have the form: `mnf#`. `nf` networks can be either SAS or DAS.

---

# Working With Network Pathgroups

## ▼ To Create a Network Pathgroup

---

**Note** – This procedure should *not* be used for the primary network. To alternately path the primary network, see “Alternately Pathing the Primary Network Interface” on page 57.

---

### 1. Use `apnet(1M)` with the `-c` option:

```
# apnet -c -p le0 -a le2
# apconfig -N -u

metanetwork:    mle0  U
physical devices:
                 le2
                 le0  P A
```

This `apnet(1M)` command creates the network pathgroup as well as the metanetwork interface name `mle0` for the two physical devices `le0` and `le2`. The metanetwork interface name is derived from the primary controller name (specified by `-p`).

The `apconfig(1M)` command lists the uncommitted network entries in the database. `-N` specifies that network database entries should be listed. `-u` specifies that uncommitted entries should be listed.

**2. If you are satisfied with the network pathgroup, commit the entry:**

```
# apdb -C
# apconfig -N

metanetwork:    mle0
physical devices:
                 le2
                 le0 P A
```

`apdb -C` commits the database entries. `apconfig -N` lists the committed network entries in the database. The listing appears exactly as it did before, except that the `U` no longer appears after `mle0`.

**3. Remove all direct usage of both members of the pathgroup (see `ifconfig(1M)`).**

**a. You may have to unplumb the physical interface, for example:**

```
# ifconfig le0 down; ifconfig le0 unplumb
```

If the interface you will be configuring down is the main network interface, or if it is the interface that you will be using as you use commands to configure the metanetwork, follow one of the procedures in “Alternately Pathing the Primary Network Interface” on page 57.

Usually network interfaces are configured during system boot via the file `/etc/hostname.xxx`, where `xxx` is the interface name (such as `le0`). This file contains the IP address or the hostname associated with the interface. You should remove or rename the `/etc/hostname.xxx` for all interfaces that have been made AP alternates, since direct usage of the alternate must not occur.

**b. Create an `/etc/hostname.mxxx` file (such as `/etc/hostname.mle0`) for any metanetworks that you want to configure at system reboot.**

This file should contain the metanetwork’s IP address or the hostname for the interface. You can simply rename `/etc/hostname.le0` to `/etc/hostname.mle0`.

The normal operating state of a network interface is plumbed up when in use, and unplumbed when not in use. When you automatically configure network interfaces by using `/etc/hostname.*`, the interfaces are left in one of these states. It is possible to leave a network interface in a transitory state of plumbed when you manually configure your network interface. As this is not a normal operational mode it is unlikely that network interfaces will be left in this state. Generally, you should not leave metanetworks in this state during AP network configuration. A network metadvice may be deleted only if it and all other

network metadevices of that device type are either in the unplumbed state or the plumbed up state. Otherwise, AP ignores the delete request, and depending on your configuration, may display warning messages of the following form:

```
WARNING:mnf_setphyspath: APUNSET busy
WARNING:ap_db_commit: mnf3 not deleted, metadevice returned error
16
```

---

**Note** – If you are using SunFDDI, you must specify a unique Media Access Control IDentifiers (MACIDs) for the metanetwork. The MACID is set by using the `ether` parameter to the `ifconfig(1M)` command. You might first examine the MACID for each of the alternates. You can do this by bringing up each alternate and examining the `ether` field. Then, fabricate a MACID that does not match any of the alternates.

The allocation of MACIDs is described in RFC1340, “Assigned Numbers”, July 1992. When generating a MACID for your AP network interface, the new 48 bit hardware address should be acquired from the IEEE Standards Office. However, it is possible to “create” a number by transposing digits on an existing MACID of one of the metainterface alternate elements. After creating a number, it is important to verify that there is no other hardware on the same subnet which is a legitimate user of the created address.

---

This metanetwork MACID will be used to configure the active physical interface of the metanetwork. The use of this MACID is necessary to prevent duplication of MACIDs on the network when combining AP switching of interfaces and DR board insertion activities. The metanetwork defaults to the MACID of the active alternate on boot. To ensure that the MACID is set properly at boot time, place `ifconfig(1M)` commands in the `/etc/rcs.d/S30rootusr.sh` start-up script.

4. Bring up the metanetwork in the usual manner, but use the metanetwork name instead of the physical network name.

You can do this by either rebooting the machine or manually configuring the network as in the following example:

```
# ifconfig mle0 plumb
# ifconfig mle0 inet 136.162.65.30 up netmask + broadcast +
Setting netmask of mle0 to 255.255.255.0
# ifconfig -a
lo0: flags=849<UP,LOOPBACK,RUNNING,MULTICAST> mtu 8232
    inet 127.0.0.1 netmask ff000000
mle0: flags=843<UP,BROADCAST,RUNNING,MULTICAST> mtu 4352
    inet 136.162.65.30 netmask ffffffff broadcast 136.162.65.255
    ether 0:0:be:0:8:c5
```

At this point, the device node, such as `/dev/mle0`, can be used to access the network from Solaris commands such as `snoop(1M)`.

## ▼ To Switch an Ethernet or SunFDDI Pathgroup

---

**Note** – You can switch a network pathgroup even while the network sustains traffic.

---

● Use the `apconfig(1M)` command:

```
# apconfig -P mle0 -a le2
# apconfig -N

metanetwork:    mle0
physical devices:
                le2  A
                le0  P
```

The `-P` option specifies the pathgroup and `-a` specifies the alternate that you want to become active. The listing above shows that the active alternate has been switched to `le2`, as indicated by the `A` following `le2`.

You do not have to commit a switch operation.



---

**Caution** – When you switch paths, AP does not check to verify that data can be transmitted over the path to which you are switching (although it does determine whether or not that path is detached or offline). You may want to verify the status of

the path before switching to it by performing an I/O operation such as `snoop(1M)`. AP does not produce any error or warning messages if you switch to a path that is not functioning properly.

---

## ▼ To Delete a Network Pathgroup

1. **Remove all usage of the corresponding metanetwork, and use `apnet -d`:**

```
# ifconfig mle0 down unplumb
# apnet -d mle0
# apconfig -N

metanetwork:    mle0  D
physical devices:
                 le2  A
                 le0  P
```

In the listing produced by `apconfig -N`, a `D` follows `mle0`, indicating that the pathgroup is marked as deleted.

2. **Commit the entries in the database using `apdb -C`:**

```
# apdb -C
# apconfig -N
#
```

The `apconfig -N` command produces no listing, indicating that the network pathgroup (the only one that existed previously in this example) has been deleted.

You can undo a deletion if the deletion is uncommitted. To undo a deletion, run `apnet -z`, specifying the same metanetwork interface that you previously deleted.

When an `apnet -m -r` or `apnet -m -a` command is used, AP marks the current pathgroup configuration as deleted and creates a new uncommitted pathgroup definition. After the database change is committed by using `apdb -C`, the new definition replaces the old.



---

## Alternately Pathing the Primary Network Interface

The primary network interface between your Sun Enterprise server and the other machines on the network is the Ethernet interface that is on the same subnet as the SSP. One way to identify the primary network is to look in the `/etc/hostname.xxx` files until you find the one that contains the IP name that matches the IP name found in the file `/etc/nodename`. The corresponding `xxx` network (for example, `qe0`) is the primary network.

You can alternately path the primary network, if you want. The primary network is the only network interface that can be auto-switched at boot-time. During the boot process, if the active alternate for the primary network fails, the system attempts to find a working alternate. Note that the AP database on your Sun Enterprise server is used for this purpose.

A subset of the host's AP database that resides on the SSP is used at boot time to automatically switch to a functional path to the boot disk. By the time the system is ready to start using the network, the file systems on the host are already up and running, so the main AP database can be used.

When you configure an alternately pathed network, you must not configure the metanetwork while the underlying driver is still active. When you configure AP for a network that you are currently using, the transition period between configuring the physical interface down and the AP interface up generates a loss of network service for your Sun Enterprise server.

The following example shows how to reconfigure the primary network. This example assumes you have an Sun Enterprise server named `hmb` with a primary network interface on `qe0` and that you want to have a metanetwork interface composed of `qe0` and `qe4`.

### ▼ To Alternately Path the Primary Network Interface

---

**Caution** – This procedure requires you to reboot the machine. If you are not ready to reboot the machine, do not perform this procedure.

---

**1. Verify that the primary network interface is `qe0`:**

```
# cat /etc/nodename
hmb
# cat /etc/hostname.qe0
hmb
#
```

**2. Create the new network pathgroup and commit the changes:**

```
# apnet -c -p qe0 -a qe4
# apdb -C
```

**3. Verify the new pathgroup by looking at committed network entries in the AP database:**

```
# apconfig -N
metanetwork:    mqe0
physical devices:
qe4
qe0  P A
```

**4. Create the new `hostname.mxxx` file so the network will be automatically configured at boot time:**

```
# cat > /etc/hostname.mqe0
hmb
^D
# cat /etc/hostname.mqe0
hmb
```

**5. Remove the configuration files for the physical network interfaces:**

```
# rm -f /etc/hostname.qe0 /etc/hostname.qe4
```

**6. Bring down the physical network interfaces and bring up the metanetwork interface by rebooting the machine.**

## Interaction Between AP and DR

---

This chapter describes the relationship between AP and DR.

---

### Using AP and DR Together

Dynamic Reconfiguration (DR) and Alternate Pathing (AP) are designed to work closely together. DR enables you to attach and detach system boards without halting the operating system, as described in the *Sun Enterprise 10000 Dynamic Reconfiguration User's Guide*. AP enables you to switch usage *away from* a controller on a board that is being detached or possibly *to* a controller on a board that has been attached.

On the Sun Enterprise 10000 server, AP *automatically* switches each disk and network metadvice that has an active controller on a board being detached (assuming a viable alternate path exists on another board). Also, on the Sun Enterprise 10000 server, AP prevents you from manually switching to controllers on a board that is in the drain state of a DR detach operation.

The following AP command shows that the `pln1` controller is on a board that is detached (as indicated by the `DE` flag) and, thus, you cannot switch to that controller:

```
# apconfig -S

c1      pln0  P A
c2      pln1  DE
        metadiskname(s):
            mc1t5d0
            mc1t4d0
            mc1t3d0
            mc1t2d0
            mc1t1d0
```

Similarly, the following AP command shows that the `pln1` controller is on a board that is in the drain state (as indicated by the `DR` flag) and, thus, you cannot switch to that controller:

```
# apconfig -S

c1      pln0  P A
c2      pln1  DR
        metadiskname(s):
            mc1t5d0
            mc1t4d0
            mc1t3d0
            mc1t2d0
            mc1t1d0
```

AP is notified that a board is in the DR drain state only on the Sun Enterprise 10000 server.

When you plan to detach a board that hosts the active controller of a pathgroup, you can manually switch to a controller on another board before the DR detach operation, or even during the DR detach operation. On machines other than the Sun Enterprise 10000 server, however, you *must* perform any such switch before you enter the Complete Detach state, or the detach operation will fail. In this case, you can perform the switch and then retry the detach operation.

For more information about DR, see the *Sun Enterprise 10000 Dynamic Reconfiguration User's Guide*.

---

## Maintaining the Correct AP State

If you attach a board that hosts a controller that is part of a disk pathgroup, you must use `apconfig -F` to clear the detach flag (DE) for that board.

If you detach a board that hosts a network controller, and that network device has not been used since the previous boot, you must run `apconfig -F` to notify the system that the network device is no longer available.

The `apconfig -N` command may incorrectly indicate that a network controller resides on a board that is detached (or incorrectly indicate that it resides on a board that is present) if the corresponding AP metadriver has not been loaded. Use `apconfig -F` to ensure the correct information is shown by `apconfig -N`.



## AP Commands

---

The AP man pages are in the *Sun Enterprise Server Alternate Pathing 2.0.1 Reference Manual* portion of your Sun Enterprise server's documentation set, and online as well (once you've installed the AP packages). This chapter introduces the AP commands and files. Unless noted otherwise, AP commands are executable on the Sun Enterprise server, not in the SSP environment. Following is a list of the AP man pages:

- `ap(1M)` – alternate pathing overview
- `ap_reboot_host(1M)` – fast alternate path boot (This command is executed on the SSP by other commands. Do not execute it manually.)
- `ap_daemon(1M)` – alternate pathing daemon
- `ap_ssp_daemon(1M)` – AP SSP daemon (This daemon runs on the SSP.)
- `apboot(1M)` – define an AP boot device
- `apcheck(1M)` – determine accessibility of AP SCSI meta-device
- `apconfig(1M)` – display and manage AP configuration
- `apdb(1M)` – manage replicas of AP database
- `apdisk(1M)` – manage AP for SCSI disks
- `apinst(1M)` – identify SCSI bus controller
- `apnet(1M)` – manage AP for networks
- `apssp(1M)` – client of AP SSP daemon (This command is executed on the SSP by other commands. Do not execute it manually.)
- `ap(7)` – AP driver
- `ap_dmd(7)` – AP disk metadriver
- `ap_nmd(7)` – AP network metadriver





## AP Components

---

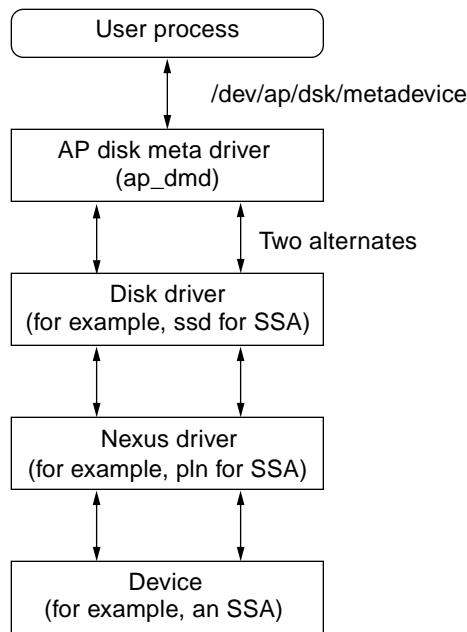
AP consists of the following components:

- AP daemon – `ap_daemon(1M)` runs on the Sun Enterprise server and receives user requests via the AP commands executed on the server. The daemon acts as an intermediary between the commands and the AP Librarian, `ap(7)`. It passes requests received via RPCs to the librarian by invoking `ioctl`s. The actual work of maintaining the database is performed by the librarian.
- AP librarian – `ap(7)` manages the AP database and interacts with the metadrivers as necessary. It receives requests via `ioctl`s and acts on them by updating the database or calling entry points in the metadrivers.
- AP metadrivers – The low-level capability for re-routing I/O accesses to alternate paths is implemented in the metadrivers. The metadrivers are `ap_dmd(7)`, `ap_nmd(7)`, `mge`, `mhme`, `mqfe`, `mle`, `mqe`, and `mnf`. All application I/O requests that use the appropriate metadisk go through a metadriver that passes them to the physical device drivers. As a result, the metadrivers can decide which physical path to use, whether a given path is no longer functioning, and so forth. The information on which the metadrivers base their decisions comes from the AP librarian and AP database.
- AP SSP daemon – `ap_ssp_daemon(1M)` executes on the SSP and receives RPC requests from the AP daemon whenever the AP database changes. (Note that the SSP is part of the Sun Enterprise 10000 server; the SSP is not available with any other Sun Enterprise servers.) This daemon is responsible for maintaining a file on the SSP containing alternate pathing information for booting.



# Driver Layers

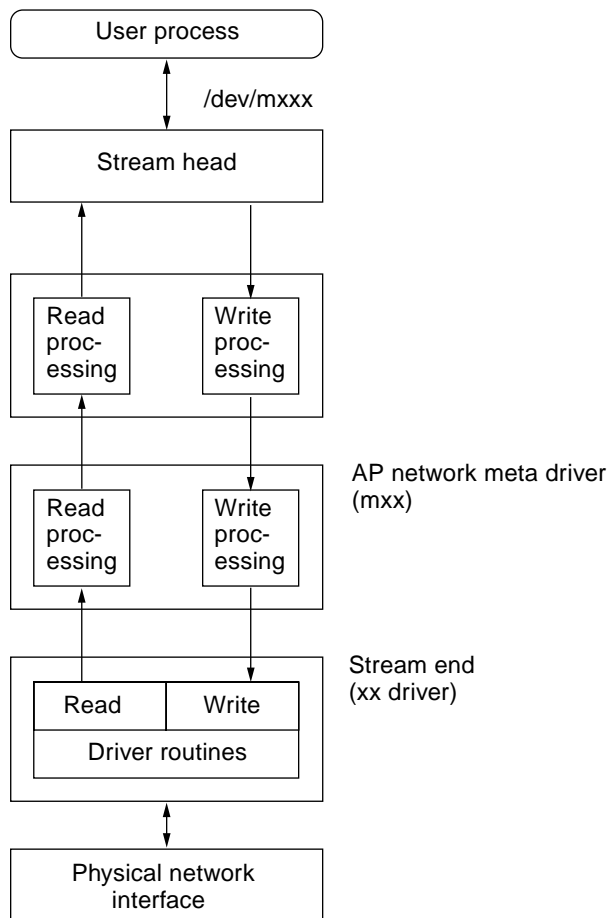
FIGURE C-1 illustrates the driver layers that are used when AP controls disk devices.



**FIGURE C-1** AP Disk Driver Layers

A user process references a metadisk, which provides access to the AP disk meta driver. The AP disk meta driver controls two instances of the physical disk driver which, in turn, controls two instances of the nexus driver (or controller driver). The nexus driver controls the physical device.

FIGURE C-2 illustrates the driver layers that are used when AP controls networks.



**FIGURE C-2** AP Network Driver Layers

A user process references a metanetwork, which provides access to the stream head. The AP network meta driver is inserted into the stream between the high-level read/write processing components and the physical driver routines.

# Glossary

---

<b>active alternate</b>	The alternate path that is currently handling I/O for a pathgroup.
<b>AP database (or simply “database”)</b>	A database that is maintained by the AP subsystem. The AP database contains all of the information needed to maintain the configuration’s alternate paths.
<b>alternate path</b>	One of the physical paths within a pathgroup.
<b>committed database entry</b>	An entry in the AP database that is currently being used by AP to manage access to a disk or network. (Compare with <i>uncommitted database entry</i> .)
<b>pathgroup</b>	A set of two alternate paths that provide access to the same device or set of devices.
<b>metadisk</b>	A disk abstraction that provides access to an underlying group of two physical paths to a disk.
<b>metanetwork</b>	A network abstraction that provides access to an underlying group of two physical paths to a network.
<b>physical path</b>	The electrical path from the host to a disk or network.
<b>pln port</b>	An optical link connection (OLC) module on an SSA controller that can be connected to an SSA port.
<b>primary path</b>	The alternate path in a pathgroup that is initially the active alternate. The name of the primary path is used when constructing the name of the metadisk or metanetwork. The primary path does not change when a switch occurs.
<b>SSA</b>	A SPARCStorage Array, a collection of disks within a hardware peripheral. The SSA provides access to each of its housed disks via two ports.

**SSA port** An optical link connection (OLC) module on an SSA that can be connected to a `pln` port.

**SSA controller** A controller that resides on the host system and has one or two `pln` ports (also called a SOC controller).

**switch** The act of establishing a new alternate path as the path to be used for a given pathgroup. Note that switching does not change the primary path.

**uncommitted database**

**entry** An entry in the AP database that has not been committed and is therefore not currently in effect. If a pathgroup has been created but the database entry has not been committed, that pathgroup is not currently used by AP to manage access to a disk or network. If a previously committed pathgroup has been deleted, but that database entry has not been committed, that pathgroup is still being used by AP to manage access to the disk or network.

# Index

---

## A

- active alternate, 69
- alternate path, 69
- alternate pathing
  - active alternate, 69
  - alternate path, 69
  - AP Daemon, 65
  - AP database, 69
  - AP Librarian, 65
  - AP Meta-Drivers, 65
  - AP SSP Daemon, 65
  - apconfig, 29
  - boot disk, 45
  - boot-device devalias, 46
  - creating meta-devices, 35
  - database partition, 27
  - DR / AP interaction, 59
  - meta-device, 33, 69
  - meta-network, 69
  - meta-network names, 51
  - path group, 69
  - physical path, 69
  - primary path, 69
  - single user mode, 48
  - switch, 70
- AP Daemon, 65
- AP database, 69
- AP Librarian, 65
- AP Meta-Driver, 65
- AP SSP Daemon, 65
- apconfig, 29

## B

- boot disk and AP, 45
- boot-device and AP, 46

## C

- creating meta-devices, 35

## D

- devalias boot-device and AP, 46
- disk, boot under AP control, 45
- DR / AP interaction, 59

## E

- Ethernet meta-network names, 51

## F

- FDDI meta-network names, 51

## L

- LE meta-network names, 51

## **M**

meta-device, 33, 69

meta-network, 69

## **P**

path group, 69

physical path, 69

primary path, 69

## **S**

single user mode and AP, 48

switch, 70