

Alternate Pathing 2.0 User's Guide

Sun Microsystems Computer Company
A Sun Microsystems, Inc. Business
2550 Garcia Avenue
Mountain View, CA 94043
U.S.A.

Part No: 805-0082-10
Revision A, March 1997



THE NETWORK IS THE COMPUTER™

Copyright 1997 Sun Microsystems, Inc., 2550 Garcia Avenue, Mountain View, California 94043-1100 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.

Portions of this product may be derived from the UNIX[®] system, licensed from Novell, Inc., and from the Berkeley 4.3 BSD system, licensed from the University of California. UNIX is a registered trademark in the United States and other countries and is exclusively licensed by X/Open Company Ltd. Third-party software, including font technology in this product, is protected by copyright and licensed from Sun's suppliers.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 and FAR 52.227-19.

Sun, Sun Microsystems, the Sun logo, Solaris, SunSoft, SunPics, the SunSoft logo, SunOS, OpenWindows, DeskSet, ONC, ONC+, Power Management, TurboGX, TurboGX Plus, S24, SunFastEthernet, Online: DiskSuite, OpenBoot, IPX, Power Management, AnswerBook, Jump Start, XGL, SLC, ELC, IPC, Sun-3, SunCD, SunCD Plus, SunCD 2Plus, SunVideo, SunDials, SunButtons, NeWSprinter, NeWSprinter CL+, NeWSprint, SunDiag, Ultra, UltraServer, UltraSPARC, Ultra Enterprise, and NFS are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the United States and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK[®] and Sun[™] Graphical User Interfaces were 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.

X Window System is a trademark of X Consortium, Inc.

THIS PUBLICATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.



Copyright 1997 Sun Microsystems Inc., 2550 Garcia Avenue, Mountain View, Californie 94043-1100, 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 de sa documentation associée 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.

Des parties de ce produit pourront être dérivées du système UNIX® licencié par Novell, Inc. et du système Berkeley 4.3 BSD licencié par l'Université de Californie. UNIX est une marque enregistrée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company Ltd. 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.

Sun, Sun Microsystems, le logo Sun, Solaris, SunSoft, le logo SunSoft, SunOS, OpenWindows, DeskSet, ONC, ONC+, Power Management, TurboGX, TurboGX Plus, S24, SunFastEthernet, Online: DiskSuite, OpenBoot, IPX, Power Management, AnswerBook, Jump Start, XGL, SLC, ELC, IPC, Sun-3, SunCD, SunCD Plus, SunCD 2Plus, SunVideo, SunDials, SunButtons, NeWSprinter, NeWSprinter CL+, NeWSprint, SunDiag, Ultra, UltraServer, UltraSPARC, Ultra Enterprise, et NFS sont des marques déposées ou enregistrées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC, utilisées sous licence, sont des marques déposées ou enregistrées 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.

Les interfaces d'utilisation graphique OPEN LOOK® et Sun™ ont été développées 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 aussi 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.

Le système X Window est un produit du X Consortium, Inc.

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 RÉPONDRE A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ILS NE SOIENT PAS CONTREFAISANTS DE PRODUITS DE TIERS.

Table of Contents

Preface.....	xi
1. Introduction to Alternate Pathing	1-1
Installing AP.....	1-1
The Purpose of Alternate Pathing	1-1
Basic Alternate Pathing Concepts	1-3
Physical Path	1-3
Meta-Disk	1-4
Meta-Network	1-5
Disk Pathgroup	1-6
Network Pathgroup.....	1-7
Alternate Pathing Terminology	1-8
Supported Devices.....	1-10
Example AP Configurations.....	1-11
AP and RSAS	1-12
AP and Domains	1-13

2. Overview of AP Operations	2-1
Working with the AP Database	2-1
▼ To Create a Copy of the AP Database	2-1
▼ To View Database Properties	2-2
▼ To View Uncommitted Disk Entries	2-2
▼ To View Committed Disk Entries	2-3
▼ To View Uncommitted Network Entries	2-3
▼ To View Committed Network Entries	2-4
▼ To Delete a Copy of the AP Database	2-4
Working with Disk Pathgroups	2-4
▼ To Create a Disk Pathgroup	2-4
▼ To Place Boot Disk Under AP Control	2-6
▼ To Remove a Boot Disk From AP Control	2-6
▼ To Switch from One Disk Pathgroup To Another	2-7
▼ To Delete a Disk Pathgroup	2-7
Working with Network Pathgroups	2-8
▼ To Create Network Pathgroup	2-8
▼ To Switch a Network Pathgroup (Ethernet or FIDDI) ..	2-9
▼ To Delete a Network Pathgroup	2-9
3. Managing the AP Database	3-1
Creating a Database	3-2
Deleting a Database	3-3
4. Using Meta-Disks and Disk Pathgroups	4-1

Device Nodes for Meta-Disks	4-1
Creating Disk Pathgroups and Meta-Disks	4-2
Switching Disk Pathgroups	4-6
Deleting Disk Pathgroups	4-8
5. Using AP Boot Devices	5-1
Placing the Boot Disk Under AP Control	5-1
▼ To Place a Boot Disk under AP Control	5-1
▼ To Remove the Boot Disk from AP Control	5-2
AP Boot Sequence	5-3
Using Single User Mode	5-4
6. Using Meta-Networks and Network Pathgroups	6-1
Meta-Network Interfaces	6-1
Creating Network Pathgroups and Meta-Networks	6-2
Switch Ethernet or FDDI Pathgroup	6-5
Deleting Network Pathgroups	6-6
Alternately Pathing the Primary Network Interface	6-7
7. Interaction Between AP and DR	7-1
Notes	7-2
8. VxVM / AP Interaction	8-1
Creating Disk Pathgroups with VxVM	8-1
Deleting Disk Pathgroups with VxVM	8-2
9. AP Commands	9-1
10. AP Components	10-1

11. Driver Layers	11-1
--------------------------------	-------------

Figures

Figure 1-1	AP and Fault Tolerance	1-2
Figure 1-2	AP and Dynamic Reconfiguration	1-3
Figure 1-3	Physical Path	1-4
Figure 1-4	Meta-Disk	1-5
Figure 1-5	Meta-Network	1-6
Figure 1-6	Disk Pathgroup	1-7
Figure 1-7	Network Pathgroup	1-8
Figure 1-8	Typical AP Configuration	1-11
Figure 1-9	AP and Disk Mirroring	1-12

Preface

This document describes the Alternate Pathing (AP) feature of the Ultra Enterprise 10000 system.

Who Should Use This Book

This manual is intended for the Ultra Enterprise 10000 system administrator. It assumes that you have a working knowledge of UNIX® systems, particularly those based on the Solaris operating environment. If you do not have such knowledge, refer to the Solaris™ User and System Administrator AnswerBooks provided with this system.

How This Book Is Organized

This document contains the following chapters:

- Chapter 1, “Introduction to Alternate Pathing,”** introduces AP.
- Chapter 2, “Overview of AP Operations,”** provides examples of typical AP command sequences.
- Chapter 3, “Managing the AP Database,”** covers the AP database operations.
- Chapter 4, “Using Meta-Disks and Disk Pathgroups,”** describes metadisks, disk pathgroups, and how to use them.
- Chapter 5, “Using AP Boot Devices,”** covers unattended system boot issues.
- Chapter 6, “Using Meta-Networks and Network Pathgroups,”** describes metanetworks, network pathgroups, and how to use them.
- Chapter 7, “Interaction Between AP and DR,”** describes how Dynamic Reconfiguration (DR) and Alternate Pathing (AP) work together.
- Chapter 8, “VxVM / AP Interaction,”** covers extra steps necessary for using VERITAS Volume Manager (VxVM) in conjunction with AP.
- Chapter 9, “AP Commands,”** introduces all AP commands.
- Chapter 10, “AP Components,”** describes the underlying AP architecture.
- Chapter 11, “Driver Layers,”** describes the underlying AP drivers.

Related Documentation

See also:

- *Alternate Pathing 2.0 Reference Manual*, part number 805-0083-10
- *Alternate Pathing 2.0 Installation and Product Notes*, part number 805-0522
- *Ultra Enterprise 10000 SSP 3.0 User’s Guide*, part number 805-0076-10
- *Ultra Enterprise 10000 SSP Reference*, part number 805-0077-10

Ordering Sun Documents

The SunDocsSM program provides more than 250 manuals from Sun Microsystems, Inc. If you live in the United States, Canada, Europe, or Japan, you can purchase documentation sets or individual manuals using this program.

For a list of documents and how to order them, see the catalog section of the SunExpressTM Internet site at <http://www.sun.com/sunexpress>.

What Typographic Changes Mean

The following table describes the typographic changes used in this book.

Table P-1 Typographic Conventions

Typeface or Symbol	Meaning	Example
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. <code>machine_name%</code> You have mail.
AaBbCc123	What you type, contrasted with on-screen computer output	<code>machine_name%</code> su Password:
<i>AaBbCc123</i>	Command-line placeholder: replace with a real name or value	To delete a file, type <code>rm filename</code> .
<i>AaBbCc123</i>	Book titles, new words or terms, or words to be emphasized	Read Chapter 6 in <i>User's Guide</i> . These are called <i>class</i> options. You <i>must</i> be root to do this.

Shell Prompts in Command Examples

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

Table P-2 Shell Prompts

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

Introduction to Alternate Pathing



Installing AP

Alternate Pathing (AP) is provided as a separate package for your Sun server system. To utilize its features you must install the following AP packages:

- SUNWapr – AP subsystem (root)
- SUNWapssp – AP subsystem (SSP)
- SUNWapu – AP subsystem (usr)
- SUNWapdoc – AP documentation

The Purpose of Alternate Pathing

AP supports Dynamic Reconfiguration (DR) and provides your Sun server system with a degree of fault recovery, defined here as the ability of a system to recover from a single-point failure in a manner that enables an uninterrupted execution of the workload. Specifically, AP provides fault recovery associated with disk and network adapter failure. If a failure occurs in

a disk or network adapter, and if an alternate adapter has been designated and is available, AP allows uninterrupted access to the disk or network. The following figure illustrates this feature:

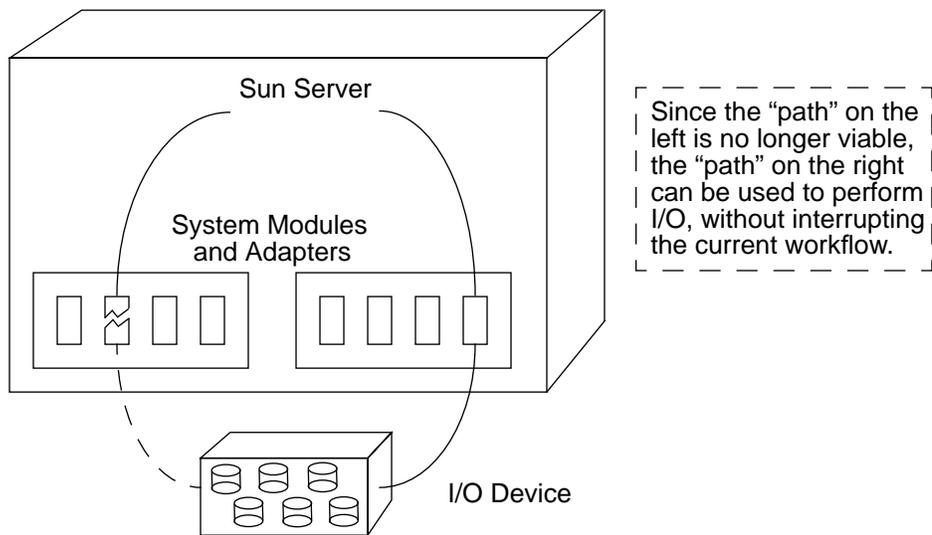


Figure 1-1 AP and Fault Tolerance

In addition to providing fault tolerance for I/O adapters, AP supports the Dynamic Reconfiguration (DR) Detach operation. If a system module is removed via DR Detach, AP allows continued access to the disks and networks

connected via that system module as long as alternate adapters have been designated and are available on other system module(s). The following figure illustrates this feature:

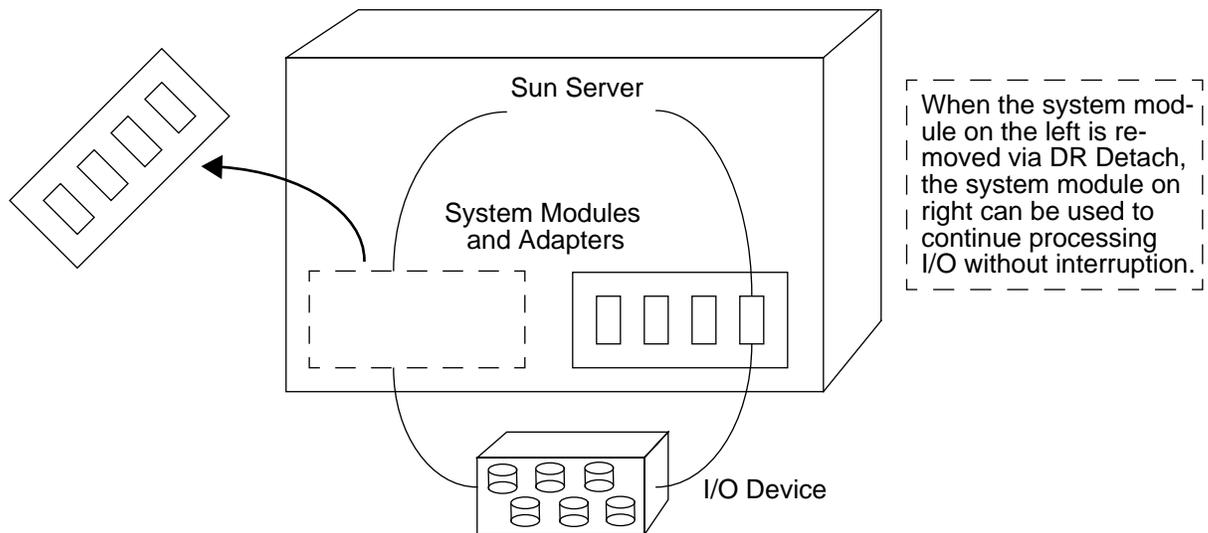


Figure 1-2 AP and Dynamic Reconfiguration

Basic Alternate Pathing Concepts

This section covers 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. The only type of disk device currently supported by AP is a SPARCStorage™ Array (SSA). In this chapter, the term *disk* always refers to an SSA. An *I/O adapter* is the controller for an I/O device, such as an SSA adapter. 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:

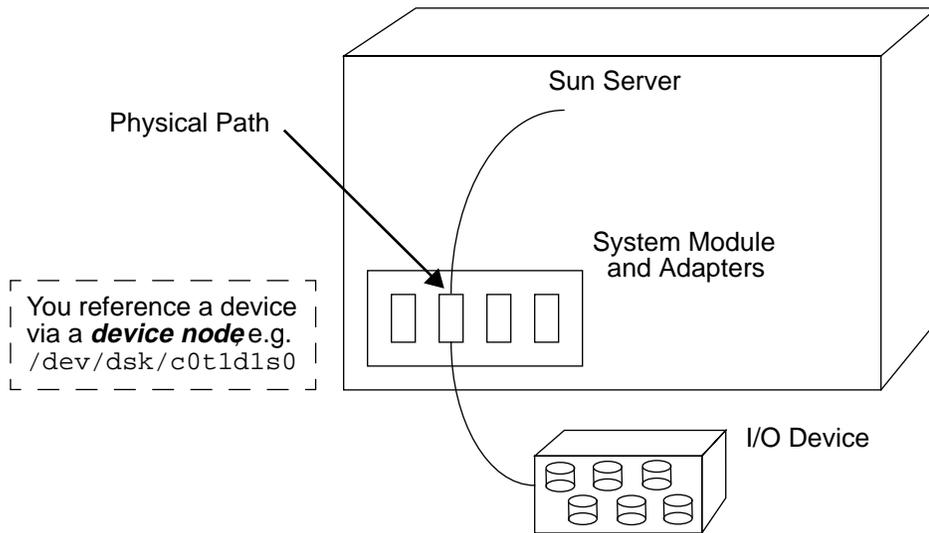


Figure 1-3 Physical Path

Meta-Disk

A *meta-disk*, as illustrated in Figure 1-4, is a construct that allows 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 meta-disk (in your scripts and programs) using an AP-specific device node such as /dev/ap/dsk/mc0t1d1s0. (See “Device Nodes for Meta-Disks” on page 4-1.)

In the following figure the `/dev/ap/dsk/mc0t1d1s0` device node is used to access a slice on a meta-disk, regardless of which `pln` port (`pln2` or `pln9`) is currently handling I/O for the meta-disk.

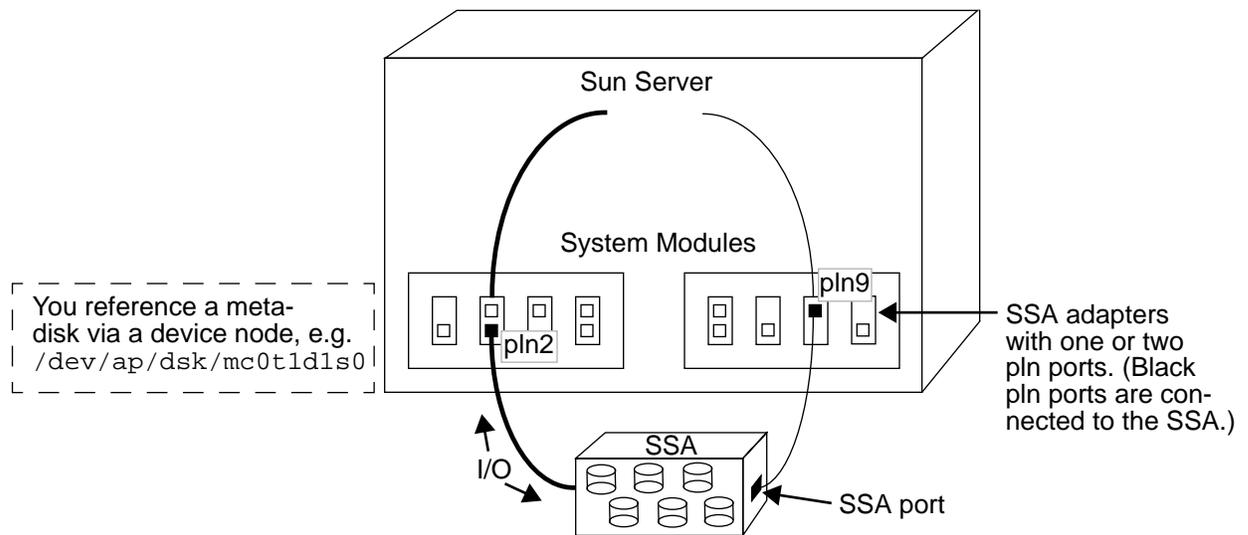


Figure 1-4 Meta-Disk

Meta-Network

A *meta-network*, as illustrated in Figure 1-5, is a construct that allows 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 meta-network (in your scripts and programs) using a *meta-network interface* name such as `mle1`. (See “Meta-Network Interfaces” on page 6-1.)

In the following figure, `mle1` is used to access a meta-network, regardless of which adapter (`le1` or `le6`) is currently processing I/O for the meta-network.

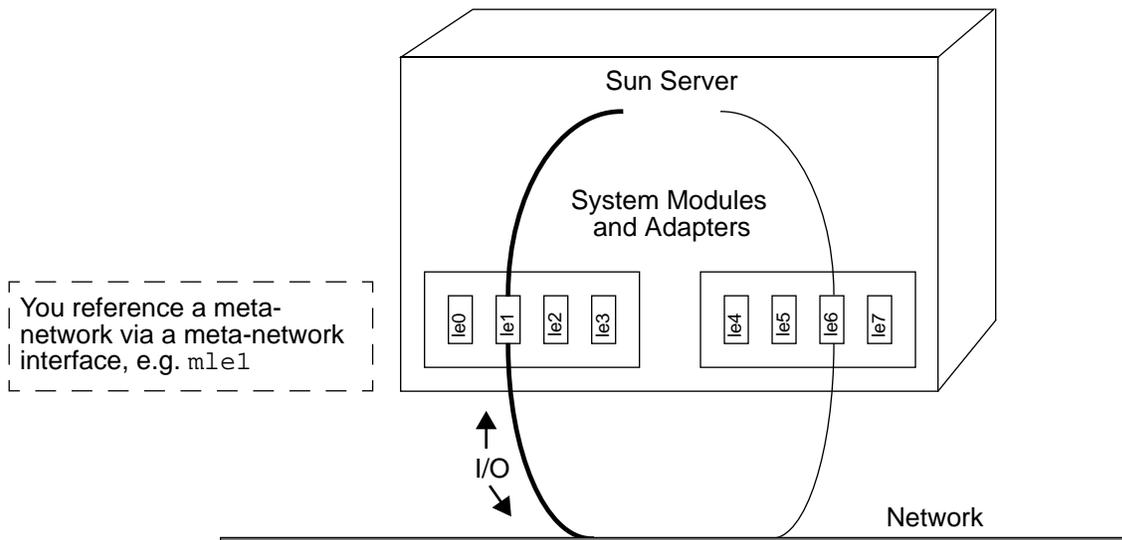


Figure 1-5 Meta-Network

Disk Pathgroup

A *disk pathgroup*, as illustrated in Figure 1-6, consists of two physical paths leading to the same SSA. 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 that the alternate path uses. Only one alternate path at a time is allowed to handle disk I/O. The alternate path that is currently handling I/O is called the *active alternate*.

Note that, whereas a meta-disk 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 (e.g., `pln1`) that corresponds to the primary path. (For information about determining the `pln` port name, see “Creating Disk Pathgroups and Meta-Disks” on page 4-2.)

For example, the following figure shows the results of using the `apconfig(1M)` command to switch the active alternate of a disk pathgroup:

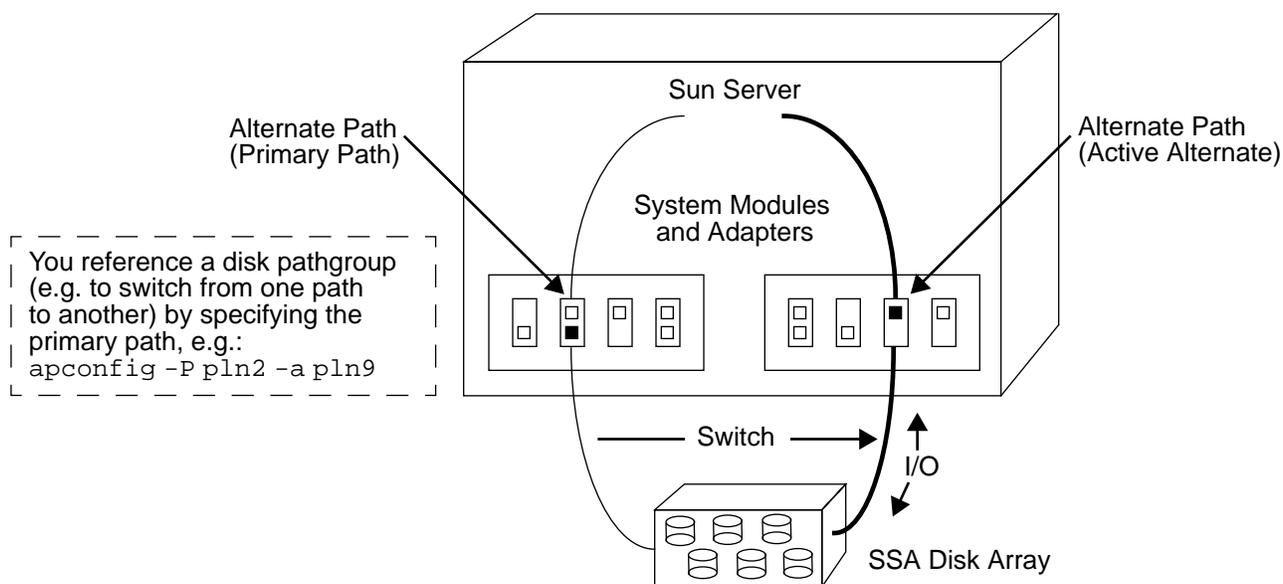


Figure 1-6 Disk Pathgroup

Network Pathgroup

A *network pathgroup*, as illustrated in Figure 1-7, consists of two network adapters 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 meta-network interface name, for example `mle1`. (Meta-network interface names are described in “Meta-Network Interfaces” on page 6-1.) For example, the following figure shows the results of using the `apconfig(1M)` command to switch the active alternate of a network pathgroup:

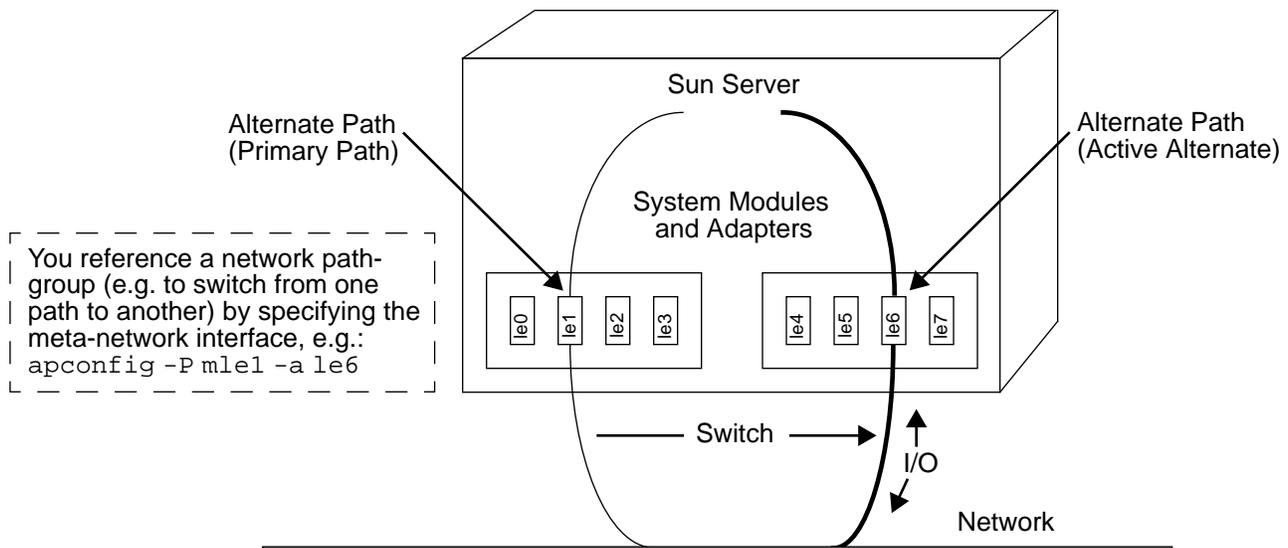


Figure 1-7 Network Pathgroup

Alternate Pathing Terminology

The following terms are used in this chapter:

active alternate

The alternate path that is currently handling I/O for a pathgroup.

AP database (or simply “database”)

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.

alternate path

One of the physical paths within a pathgroup.

committed database entry	An entry in the AP database that is currently being used by AP to manage access to a disk or network. (Compare with uncommitted database entry.)
pathgroup	A set of two alternate paths that provide access to the same device or set of devices.
meta-disk	A disk abstraction that provides access to an underlying group of two physical paths to a disk.
meta-network	A network abstraction that provides access to an underlying group of two physical paths to a network.
physical path	The electrical path from the host to a disk or network.
p1n port	An optical link connection (OLC) module on an SSA adapter that can be connected to an SSA port.
primary path	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 meta-disk or meta-network. The primary path does not change when a switch occurs.
SSA	A SPARCStorage Array. An SSA is 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 p1n port.
SSA adapter	An adapter that resides on the host system and has one or two p1n ports (a.k.a., SOC adapter).
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.

Supported Devices

AP supports SPARCStorage Arrays attached to `pln` ports on SSA adapters. For a list of the network devices supported by AP, see the release notes for this product.

After you set up Alternate Pathing for disks, you can use Online:DiskSuite version 3 (but not Solstice DiskSuite version 4) and Sun Volume Manager normally. However, if any of the physical disks have alternate adapters, those disks must be known to Online:DiskSuite™ or Volume Manager by their AP meta-disk names. This requirement enables the AP system to switch to alternate paths for these disks without interfering with Online:DiskSuite or Volume Manager.

You can place your Sun server's host boot disk and the primary network interface under AP control. AP makes it possible for the system to boot unattended even if the primary network or boot disk controller is not accessible, as long as a usable alternate path for these devices is defined.

Example AP Configurations

The following diagram shows how you can use AP to provide fault tolerance for an Ethernet network and a SPARCStorage Array (SSA):

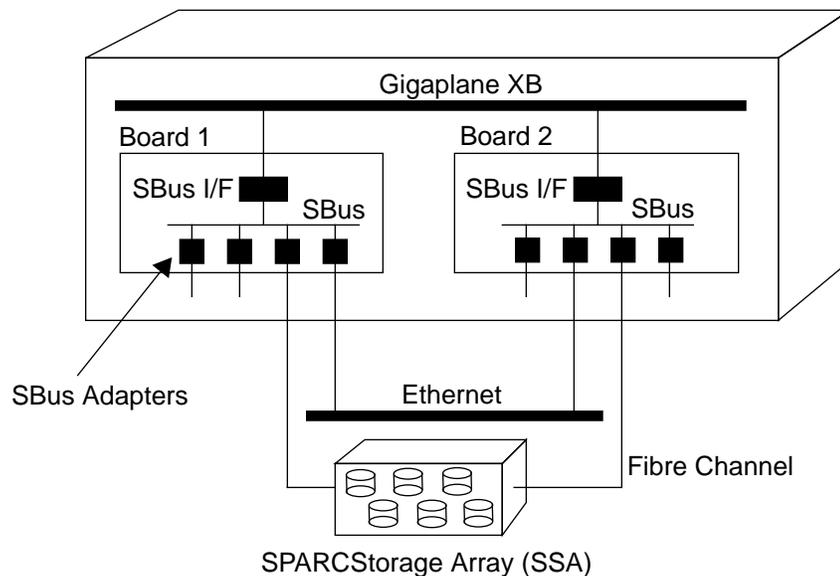


Figure 1-8 Typical AP Configuration

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

AP is similar to, but not the same as, disk mirroring. Disk mirroring replicates data to separate devices and thereby achieves *data* redundancy. AP, on the other hand, achieves *pathing* redundancy. Disk mirroring and AP are complementary; you can use them together to achieve both data redundancy *and* pathing redundancy.

The following figure shows how AP might be used in conjunction with disk mirroring:

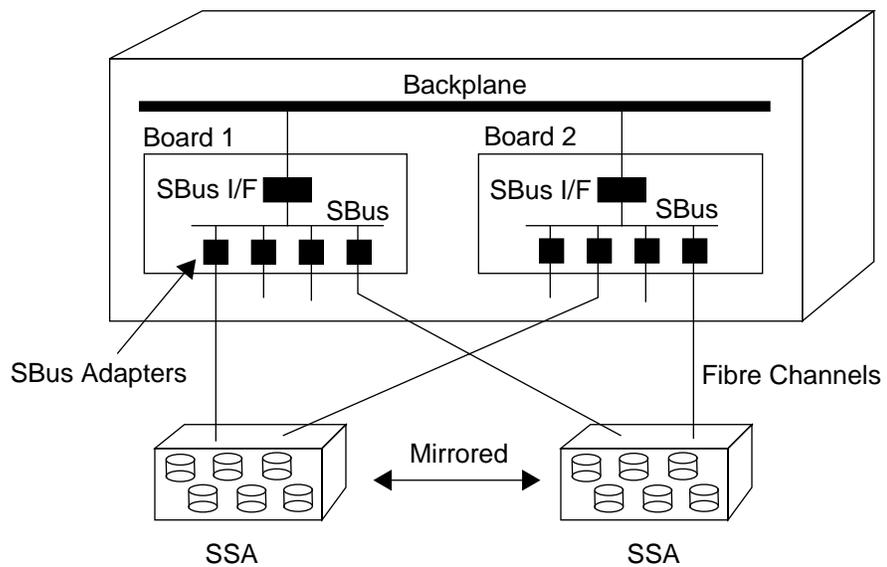


Figure 1-9 AP and Disk Mirroring

In this example, the mirroring occurs *on top of AP*, which enables switching of the underlying adapters used to implement the SSA mirror from one board to another without disruption of the disk mirroring or any active I/O.

AP and RSAS

Redundant Single Attach Station (RSAS) is a method of configuring a FDDI logical network interface with two independent pairs of Physical Layer Protocol (PHY) and Medial Access Control (MAC) entities. RSAS is a feature of FDDI 2.0. FDDI 3.0 does not support RSAS, nor does AP.

Despite the presence of multiple PHY/MAC pairs, an RSAS cannot connect directly to the counter-rotating ring, but instead must be connected via a concentrator. The advantage of RSAS over SAS is that RSAS provides

automatic *failover* (switching on failure). In the event of a detected error, the active interface is automatically switched to the alternate interface, ensuring continuous service.

AP meta-networks operate in a similar fashion. AP does not currently support automatic hardware failover; all switching is done manually. You can execute `apconfig(1M)` with its `-P` option at any time to execute a switch. The advantage of using an AP meta-network is that you can explicitly choose the hardware that is used to support a logical interface, and thereby prepare the system for Dynamic Reconfiguration (DR).

AP and Domains

AP cannot be used across two Dynamic System Domains. For example, suppose a board contains an adapter that is part of a pathgroup, and you move that board into a different domain. (You can only do this 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.

Overview of AP Operations



This section provides a summary overview of the major AP operations. You must log onto your Sun server as superuser to execute these commands. The rest of this chapter further describes the operations shown here. After you read through the chapter, you may wish to use this section as a quick reference guide.

Working with the AP Database

▼ To Create a Copy of the AP Database

- ♦ **Execute `apdb(1M)` with its `-c` and `-f` options to specify the raw disk slice where you wish to create the database copy:**

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

The `-f` (force) option is necessary to create the first AP database copy. You must dedicate an entire disk partition, one that has at least 300KB, to each database copy.

For more information see Chapter 3, “Managing the AP Database”.

▼ To View Database Properties

- ◆ **Execute** `apconfig(1M)` **with its** `-D` **option as follows, where** `D` **stands for database:**

```
# apconfig -D

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

For more information see Chapter 3, “Managing the AP Database”.

▼ To View Uncommitted Disk Entries

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

```
# apconfig -S -u

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

For more information see Chapter 4, “Using Meta-Disks and Disk Pathgroups”.

▼ To View Committed Disk Entries

- ◆ **Execute** `apconfig(1M)` **with its** `-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 4, “Using Meta-Disks and Disk Pathgroups”.

▼ To View Uncommitted Network Entries

- ◆ **Execute** `apconfig(1M)` **with its** `-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
```

▼ To View Committed Network Entries

◆ Execute `apconfig(1M)` with its `-N` option, as follows:

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

▼ To Delete a Copy of the AP Database

◆ Execute `apdb(1M)` with its `-d` and `-f` options, as follows, to specify the raw disk slice of the copy that you wish to delete:

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

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

For more information, see Chapter 3, “Managing the AP Database”.

Working with Disk Pathgroups

▼ To Create a Disk Pathgroup

1. Execute `apdisk(1M)` with its `-c` and `-a` options to create an uncommitted disk pathgroup.

`-p` specifies the primary path, `-a` specifies the alternate path:

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

2. Execute `apconfig(1M)` with its `-S` and `-u` options to view the uncommitted disk entries in the database.

3. Execute `apdb(1M)` with its `-C` option to commit the uncommitted database entries:

```
# apdb -C
```

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

4. Execute `drvconfig(1M)` to rebuild the entities in the devices directory for `ap_dmd(7)`:

```
# drvconfig -i ap_dmd
```

5. Execute the following `ap_dmd(7)` command to view the AP devices, to verify the results:

```
# ls /devices/pseudo/ap_dmd*
```

6. Execute `apconfig(1M)` with its `-R` option to create symbolic links from the devices directory to the meta-disk slices:

```
# apconfig -R
```

7. Execute the following command to view the symbolic links, to verify the results:

```
# ls -l /dev/ap/dsk
```

- 8. Modify every reference that uses a physical device node (i.e., a path that begins with `/dev/dsk` or `/dev/rdisk`) to use the corresponding meta-disk device node (i.e., 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 meta-disk path name.

For more information, refer to “Creating Disk Pathgroups and Meta-Disks” on page 4-2.

▼ To Place Boot Disk Under AP Control

- 1. Create a disk pathgroup for the boot disk, as described in the previous subsection.**
- 2. Run the `apboot(1M)` with the meta-disk name of the boot disk, for example:**

```
# apboot mc2t0d0
```

For more information, see Chapter 5, “Using AP Boot Devices”.

▼ To Remove a Boot Disk From AP Control

- ◆ Execute `apboot(1M)`, specifying the physical device name of an alternate of the boot meta-disk currently specified in `/etc/vfstab`:**

```
# apboot c2t0d0
```

For more information, see Chapter 5, “Using AP Boot Devices”.

▼ To Switch from One Disk Pathgroup To Another

1. Use the `apconfig(1M)` command with `-P` and `-a`:

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

`-P` specifies the primary adapter of the pathgroup for which you wish to change the active alternate. `-a` specifies the alternate that you wish to make active.

2. You may wish to verify the results by running `apconfig(1M)` with its `-S` option to view the committed meta-disks in the database.
Note that you do not have to commit a switch operation.

For more information, see “Switching Disk Pathgroups” on page 4-6.

▼ To Delete a Disk Pathgroup

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

```
# apdisk -d pln0
```

2. Execute `apconfig(1M)` with `-S` option to view the committed SCSI disk entries in the database, to verify the operation.

If the pathgroup was previously committed, it is still committed and fully functional at this point.

3. Execute the following `apdb(1M)` command to commit the database entries, thereby completing the deletion:

```
# apdb -C
```

For more information, see “Deleting Disk Pathgroups” on page 4-8.

Working with Network Pathgroups

▼ To Create Network Pathgroup

1. **Execute `apnet(1M)` to create a meta-network name.**

Use the `-c` option to specify the primary network and the `-p` option to specify the alternate network:

```
# apnet -c -p 1e0 -a 1e2
```

2. **Execute `apconfig(1M)` with its `-N` and `-u` options to view the uncommitted network entries in the database, to verify the operation.**
3. **Execute `apdb(1M)` to commit the network pathgroup in the database:**

```
# apdb -C
```

4. **Execute `apconfig(1M)` with its `-N` option to view the committed network entries in the database, to verify the results.**
5. **Before you can use a meta-network, you must remove all direct usage of the physical devices that make up the pathgroup.**
 - a. **You may have to unplumb the physical interface.**
 - b. **Use `ifconfig(1M)` to bring up the meta-network in the usual manner, but use the meta-network name instead of the physical network name.**
 - c. **Remove or rename the `/etc/hostname.xxx` for all interfaces that have been made AP alternates, since direct usage of the physical paths must not occur.**

For more information, see “Creating Network Pathgroups and Meta-Networks” on page 6-2.

▼ To Switch a Network Pathgroup (Ethernet or FDDI)

1. Execute `apconfig(1M)` with its `-P` and `-a` options to switch from one alternate path to another within an Ethernet or FDDI network pathgroup:

```
# apconfig -P m1e0 -a 1e2
```

`-P` specifies the name of the meta-network. `-a` specifies the alternate path that you wish to make active.

2. Execute `apconfig(1M)` with its `-P` option to view the committed meta-networks in the database, to verify the results.
Note that you do not have to commit a switch operation.

For more information, see “Switch Ethernet or FDDI Pathgroup” on page 6-5.

▼ To Delete a Network Pathgroup

1. Execute `apnet(1M)` with its `-d` option to specify the meta-network name:

```
# apnet -d m1e0
```

2. Execute `apconfig(1M)` with its `-N` option to view the committed network entries in the database, to verify the operation.

If the pathgroup was previously committed, it is still committed and fully functional at this point.

3. Execute `apdb(1M)` with its `-C` option to commit the database entries, thereby completing the deletion:

```
# apdb -C
```


Managing the AP Database



AP maintains a database that contains information about all defined meta-disks and meta-networks, and their corresponding alternate paths and properties. Conceptually, a single AP database is maintained on a given Sun server system. However, you can and 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 300KB, 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:

- Sun Microsystems recommends that you set up three to five database copies.
- As configured at the factory, partition 4 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.
- The database copies should have no I/O adapters in common with each other. Following this rule allows maximum availability in case of adapter failure.
- If you have configured your system to make use of Dynamic Reconfiguration (DR), the database copies should be hosted by I/O adapters on different system boards so that a database copy is accessible if one of the system boards is detached.

A subset of the information in the AP database is automatically maintained on the SSP for use at boot time. This database contains alternate pathing information for your Sun server's 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 5-4.

Creating a Database

To create AP database copies, use the `apdb(1M)` command. To create an AP database copy, use `apdb -c`:

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

When you create a database copy, you must specify the raw disk path (`/dev/rdisk`) to the slice where you wish to create the database. You must use the force option (`-f`) when creating the first database copy (i.e., when your Sun server currently has no AP databases).

To view information about the 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
```

The above example has only one AP database copy and 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.

The AP database contains information about SCSI and network device class pathgroups. When a pathgroup is initially defined (as described in Chapter 4, “Using Meta-Disks and Disk Pathgroups” and Chapter 6, “Using Meta-Networks and Network Pathgroups”), its pathgroup definition is an *uncommitted* database entry. The meta-disk or meta-network associated with an uncommitted entry is not created and cannot be accessed 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) allow 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.

Deleting a Database

To delete a database copy, use `apdb -d`, as follows:

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

When you delete a database copy, you must specify the raw disk (`/dev/rdisk`) path to the slice where the database copy resides. You must use the force option (`-f`) to delete the second-to-last database copy and the last database copy. In the example above, the last database copy was deleted. (This must be the case since the subsequent `apconfig -D` command returns no results.)

Note – 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 meta-disk 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.

Using Meta-Disks and Disk Pathgroups



You can only create meta-disks and disk pathgroups for disks that are accessible via two paths. You should generally use two separate SSA adapters 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 via its physical device name.

Device Nodes for Meta-Disks

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

```
/dev/dsk/c0t0d0s0
```

or

```
/dev/rdisk/c0t0d0s0
```

where *c* references the *p1n* port, *t* is the SCSI bus within the SSA, *d* is the SCSI target ID of the disk on that bus, and *s* is the slice number on the disk. These physical device nodes represent a particular physical path to a partition on a disk in the SSA.

Each SSA adapter port has both a port number (such as `c0`) and a port name (such as `pln2`). The port name consists of the port type and *instance* number. See `/etc/path_to_inst`. When an SSA is connected to two `pln` 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 meta-disk is derived from the physical device node of the primary path for a pathgroup. Here are two examples of meta-disk device nodes:

```
/dev/ap/dsk/mc0t0d0s0 and /dev/ap/rdsk/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 meta-disk has the ability to access the underlying physical disk drive from multiple paths.

Creating Disk Pathgroups and Meta-Disks

Before you can create a disk pathgroup, you must decide which two `pln` ports are to make up the alternate paths for the pathgroup. You can use the `apinst(1M)` command to display all `pln` ports (e.g., `pln0` and `pln1`) and their disk device nodes (e.g., `/dev/dsk/c1t0d0` etc.). However, you must know your system hardware configuration in order to recognize when two `pln` ports are connected to the same SSA. You must specify the `pln` port names, not the device nodes, when you create and delete disk pathgroups. Here is an `apinst` example:

```
# 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
```

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`.

To create a meta-disk name that is associated with two physical device paths, use the `apdisk(1M)` command:

```
# apdisk -c -p pln0 -a pln1
# apconfig -S -u

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

This `apdisk(1M)` command in the listing above creates a meta-disk name, as well as all of the necessary information in the AP database for maintaining the two alternate paths for all six SSA disks. `-c` specifies that this information is to be created, while `-p` specifies the alternate path that will be the primary path and `-a` specifies the other alternate path. Note that you use the `pln` port name (such as `pln0`) rather than the `pln` port number (such as `c1`) for these options.

The `apconfig -S -u` command in the listing above lists the uncommitted meta-disks. `-S` lists SCSI devices only (that is, disks rather than networks). `-u` lists uncommitted devices only. The `U` next to each meta-disk name indicates that the meta-disk entry is uncommitted. The `R` indicates the root disk (which contains the root file system found under the `/` directory.)

In the listing above, 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 meta-disk is named, and it is used to identify the meta-disk. In this case, `c1t0d0` from the primary path name becomes part of `mc1t0d0` in the meta-disk name.

Assuming that you are satisfied with the resulting meta-disk, the next step is to commit it to all copies of the AP database by using `apdb -C`:

```
# apdb -C
# apconfig -S

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

The `apdb -C` command commits all uncommitted entries in the AP database. `apconfig -S` lists the committed SCSI entries in the database. As you can see, the listing appears exactly the same as the previous listing, except the `U` no longer appears after the meta-disk names, indicating that the meta-disk is no longer uncommitted.

The next step is to create the disk nodes with the `drvconfig` command:

```
# drvconfig -i ap_dmd
# 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
...
```

`drvconfig` rebuilds the devices directory, which represents the device tree in the kernel. The AP disk meta-driver is a pseudo-device. As you can see from the listing, `drvconfig` created minor nodes for the alternately pathed device.

The next step is to execute `apconfig -R` as shown below to create symbolic links from `/dev/ap/dsk` and `/dev/ap/rdisk` to the meta-disk special files under `/devices/pseudo`:

```
# apconfig -R
# cd /dev/ap/dsk
# ls -l
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. Every reference that uses a physical device node (i.e., a path that begins with `/dev/dsk` or `/dev/rdisk`) should now be modified to use the corresponding meta-disk device node. You should check `/etc/vfstab` for such references. If a partition is currently mounted under a physical device path, you must first unmount it, and then mount it again using the meta-disk device node. If you don't wish 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 5, “Using AP Boot Devices”.)

If you are using the VERITAS Volume Manager (VxVM) in conjunction with AP, see Chapter 8, “VxVM / AP Interaction” for additional steps that you must perform.

Switching Disk Pathgroups

Switching from one path to another is easy. First, 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
```

As you can see, `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`.)

To perform the switch, use the `apconfig(1M)` command with `-P` and `-a`:

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

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

`-P` specifies the primary path and thereby *identifies the pathgroup* for which you wish 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 wish to make active. As you can see from the listing produced by `apconfig -S`, the active alternate has been switched to `pln1`. (As mentioned earlier, switching the active alternate does not change the primary path.)

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.

You could perform a switch back to the original configuration as follows:

```
# 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, says to switch the active alternate to `pln0` for the pathgroup whose primary adapter is `pln0`.

Warning – 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 wish to verify the status of the path before switching to it by performing an I/O operation such as `prtvtoc(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.

Deleting Disk Pathgroups

To delete a disk pathgroup, use `apdisk -d`, as shown below. Note that all usage of the meta-disks in the pathgroup must be discontinued (e.g., the file systems must be unmounted) before executing this command.

```
# apdisk -d pln0
# 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 that you delete has not been 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 a `D` indicates that it is marked for deletion. To commit the entries in the database, use `apdb -C`:

```
# apdb -C
# apconfig -S
```

You can undo a deletion if the deletion is uncommitted. To undo a deletion, run `apdisk -z`, specifying the same `pln` port that you previously specified.

If you are using the VERITAS Volume Manager (VxVM) in conjunction with AP, see Chapter 8, “VxVM / AP Interaction” for more information related to deleting disk pathgroups.

Placing the Boot Disk Under AP Control

To allow for unattended system boot, even if the adapter for the boot disk fails, you can place your Sun server 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 4, “Using Meta-Disks and Disk Pathgroups”.

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

`apboot` modifies `/etc/vfstab` and `/etc/system`. Here is an example:

```
# apboot mc2t0d0
```

where `mc2t0d0` is the meta-disk name of the boot disk. `apboot` examines `/etc/vfstab` and replaces the physical device name of the disk (e.g., `/dev/dsk/c2t0d0*` or `/dev/dsk/c1t0d0*`) with the meta-disk name (e.g., `/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 meta-disks for the boot disk. Instead, use `apboot` to ensure that all required changes are made.

3. Set the Open Boot Prom (OBP) devalias variable `boot-device` to the physical path most likely to be used for booting.

For more information about OBP, refer to the “SSP Internals” chapter of the *Ultra Enterprise 10000 SSP User’s Guide*.

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.

Note – If you wish 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 `pln` 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 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.

Note – 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 (via `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 wish to place under AP control.

Warning – If you place the boot disk under AP control, and later decide to remove the AP package (via `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 your Sun server is booted on an alternately pathed boot disk. 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 your Sun server is fully booted, you use the AP command versions located in `/usr/sbin`. However, if your Sun 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 single user:

- 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 (i.e., 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 Meta-Networks and Network Pathgroups



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 adapters should be connected to the same subnet.

While multiple physical network connections exist, only one adapter at a time is active. The adapters 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.

Meta-Network Interfaces

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

- LE Ethernet meta-network names have the form `mle#`, where `#` is the instance number. For example, assume the network adapters `le0` and `le1` connect to the same Ethernet network. A meta-network `mle0` could

encompass these two adapters (if the primary adapter is `le0`). Similarly, QE Ethernet meta-network names have the form `mqe#`. Note that you cannot mix `le` and `qe` networks within the same pathgroup.

- FDDI meta-network names have the form `mbf#` and `mnf#`. `nf` networks can be either SAS or DAS; `bf` networks can only be SAS. Note that you cannot mix `nf` and `bf` networks within the same pathgroup.

Creating Network Pathgroups and Meta-Networks

To create a network pathgroup, use the `apnet(1M)` command. An example follows:

```
# 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 meta-network interface name `mle0` for the two physical devices `le0` and `le2`. The meta-network interface name is derived from the primary adapter 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.

If you are satisfied with the network pathgroup, the next step is to 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`.

Before you can use a meta-network, you must remove all direct usage of both members of the pathgroup (see `ifconfig(1M)`). You may have to unplumb the physical interface; for example:

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

Note – 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 execute the commands to configure the meta-network, follow one of the procedures in “Alternately Pathing the Primary Network Interface” on page 6-7.

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.

You should create an `/etc/hostname.mxxx` file (such as `/etc/hostname.mle0`) for any meta-networks that you wish to configure at system reboot. This file should contain the meta-network’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 via `/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 meta-networks in this state during AP network configuration. A network meta-device may be deleted only if it and all other network meta-devices 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, metadvice returned error
16
```

Note – If you are using FDDI, you must specify a unique MACID for the meta-network. The MACID is set via 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 Media Access Control Identifiers (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, 345 East 47th Street, New York, N.Y. 10017. However, it is possible to “create” a number by transposing digits on an existing MACid of one of the meta-interface’s 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 meta-network MACID will be used to configure the active physical interface of the meta-network. 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 meta-network 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.

Finally, bring up the meta-network in the usual manner, but use the meta-network name instead of the physical network name. This may be done 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)`.

Switch Ethernet or FDDI Pathgroup

To perform a switch on an Ethernet or FDDI pathgroup, use the `apconfig(1M)` command:

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

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

`-P` specifies the pathgroup and `-a` specifies the alternate that you want to become active. As you can see in the listing produced by `apconfig -N`, the active alternate has been switched to `le2`, as indicated by the `A` following `le2`. You can perform a switch like this even when the network contains traffic.

Warning – 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 wish 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.

Deleting Network Pathgroups

To delete a network pathgroup, remove all usage of the corresponding meta-network, 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. To complete the deletion, commit the entries in the database with `apdb -C`:

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

As you can see, the `apconfig -N` command produces no listing, indicating that the network pathgroup (the only one that existed previously) has been deleted.

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

Note – When an `apnet -m -r` or `apnet -m -a` command is executed, AP marks the current pathgroup configuration as deleted and creates a new uncommitted pathgroup definition. Once the database change is committed via `apdb -C`, the new definition replaces the old.

Alternately Pathing the Primary Network Interface

The primary network interface between your Sun 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 (e.g., `qe0`) is the primary network.

You can alternately path the primary network, if you wish. 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 server is used for this purpose.

Recall that 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 meta-network 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 server. To perform this transition, you should use one of the following procedures, which are shown in order of preference:

- Create the appropriate AP database entries, create a new `/etc/hostname.mxxx` file, remove (or rename) the corresponding `/etc/hostname.xxx` file, and then reboot your Sun server.
- Set up a script file to perform the transition on your Sun server.
- Log into your Sun server via another network interface to allow commands to be issued when the network service is lost on the network interface that you are bringing up under AP.

The following example shows how to reconfigure the primary network using all three of these approaches. This example assumes you have a Sun server named `hmb` with a primary network interface on `qe0`, and that you want to have a meta-network interface composed of `qe0` and `qe4`.

First, verify that the primary network interface is `qe0`:

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

Next, create the new network pathgroup and commit the changes:

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

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

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

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
```

Remove the configuration files for the physical network interfaces:

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

The next step is to bring down the physical network interfaces and bring up the meta-network interface. As mentioned earlier, the underlying physical network interfaces must be taken down before you bring up the meta-network. You could follow any of the following three procedures:

1. Reboot the machine. The `qe0` and `qe4` network interfaces are automatically taken down, and `/etc/hostname.mqe0` causes the meta-network interfaces to be started automatically on reboot. The individual interfaces `qe0` and `qe4` will not be automatically started, since their `/etc/hostname` files have been deleted.
2. Generate a script to configure the `qe0` and `qe4` interfaces down, then configure up the meta-network interface. You can use `ifconfig -a` to determine the parameters that you need to enter into the script:

```
# ifconfig -a
lo0: flags=849<UP,LOOPBACK,RUNNING,MULTICAST> mtu 8232
inet 127.0.0.1 netmask ff000000
qe0: flags=863<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST> mtu
1500
inet 136.162.22.45 netmask ffffffff broadcast 136.162.22.255
ether 0:0:be:0:8:c5
# cat > /tmp/hmb.restart
ifconfig qe0 down unplumb
ifconfig qe4 down unplumb
ifconfig mqe0 plumb
ifconfig mqe0 inet 136.162.22.45 broadcast + netmask + up
^D
# chmod 755 /tmp/hmb.restart
# nohup /tmp/hmb.restart &
```

3. If the machine supports another active interface, log into your Sun server via one of the alternate interfaces and issue the following commands:

```
# ifconfig qe0 down unplumb
# ifconfig qe4 down unplumb
# ifconfig mqe0 plumb
# ifconfig mqe0 inet 136.162.22.45 broadcast + netmask + up
```


Interaction Between AP and DR



Dynamic Reconfiguration (DR) and Alternate Pathing (AP) are designed to work closely together. DR allows you to attach and detach system boards without bringing down your Sun server, as described in the *Dynamic Reconfiguration User's Guide*. AP allows you to switch usage *away from* an adapter on a board that is being detached or possibly *to* an adapter on a board that has been attached. You must perform the switch manually via `apconfig(1M)` with its `-P` option, as shown in the previous two chapters.

DR notifies the AP Daemon of the current status of a DR operation. (The AP Daemon is described in Chapter 10, "AP Components".) The AP Daemon uses this information to restrict certain operations. If an adapter is on a board that is being drained, AP marks it with a DR flag; if it is on a board that is detached, AP marks it with a DE flag. AP does not let you switch any adapters that have DR or DE flags. When a board is in the Attached state, AP does allow you to switch to any alternates on that board.

The following AP command shows that the pln0 controller is detached and, thus, unavailable. AP would disallow any attempt to switch to pln0 while the board it is on is in this detached state.

```
# ./apconfig -S

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

When you plan to detach a board, you can switch away from adapters on that board at any time prior to a detach operation, or even during the Init Detach and drain states of a detach operation. However, you *must* perform any necessary switching before you enter the Complete Detach state, or the detach operation will fail. In this case, you can perform the switch and retry the detach operation. For more information about DR, see the *Dynamic Reconfiguration User's Guide*.

Notes

If you attach a board that has an SSA path you must execute `apconfig -F` to clear the DR Detach flag for that board.

If you DR Detach a board that has a network device, 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.

Output of the `apconfig -N` command may incorrectly show a network as DR Detached or Present if its controlling AP meta-driver has not been loaded. Run `apconfig -F` to ensure correct information.

VxVM/AP Interaction



If you use the VERITAS Volume Manager (VxVM) in conjunction with AP, you must perform extra steps when creating and deleting disk pathgroups.

Creating Disk Pathgroups with VxVM

After you create a disk pathgroup as described in “Creating Disk Pathgroups and Meta-Disks” on page 4-2, you must run the following command:

```
# vxctl enable
```

This allows VxVM to rediscover the new configuration. Be sure to run this command after you run the other required commands (in the proper order), i.e.:

1. **apdisk -c**
2. **apdb -C**
3. **drvconfig -i ap_dmd**
4. **apconfig -R**
5. **vxctl enable**

Deleting Disk Pathgroups with VxVM

The process of deleting a disk pathgroup is described in “Deleting Disk Pathgroups” on page 4-8. If you are using VxVM, however, you must instruct VxVM to stop using a meta-disk *before* you delete the corresponding disk pathgroup. For more information, see the VxVM documentation.

The AP man pages are in the *Alternate Pathing 2.0 Reference Manual* portion of your Sun 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 your Sun 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 adapter
- `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

≡ 9

- `ap(7)` – AP driver
- `ap_dmd(7)` – AP disk meta-driver
- `ap_nmd(7)` – AP network meta-driver

AP consists of the following components:

AP Daemon

`ap_daemon(1M)` runs on your Sun 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 meta-drivers as necessary. It receives requests via `ioctl`s and acts on them by updating the database or calling entry points in the meta-drivers.

AP Meta-Drivers

The low-level capability for re-routing I/O accesses to alternate paths is implemented in the meta-drivers. The meta-drivers are: `ap_dmd(7)`, `ap_nmd(7)`, `mle`, `mqe`, `mnf`, `mhi`, and `mbf`

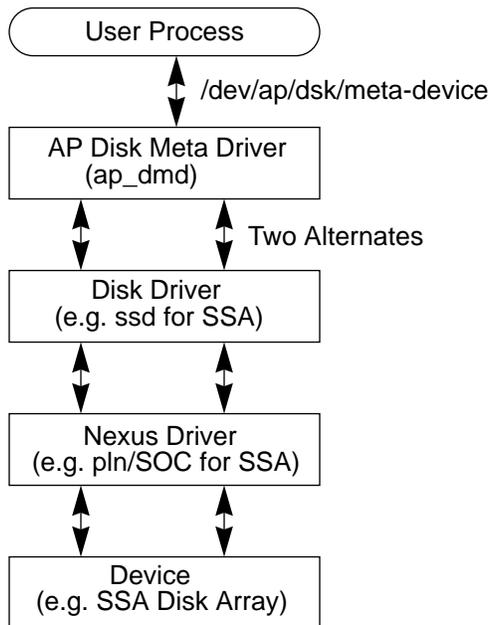
All application I/O requests that use the appropriate meta-disk go through a meta-driver that passes them to the physical device drivers. As a result, the meta-drivers can decide which physical path to use, whether a given path is no longer functioning, and so forth. The information on which the meta-drivers 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. This daemon is responsible for maintaining a file on the SSP containing alternate pathing information for booting.

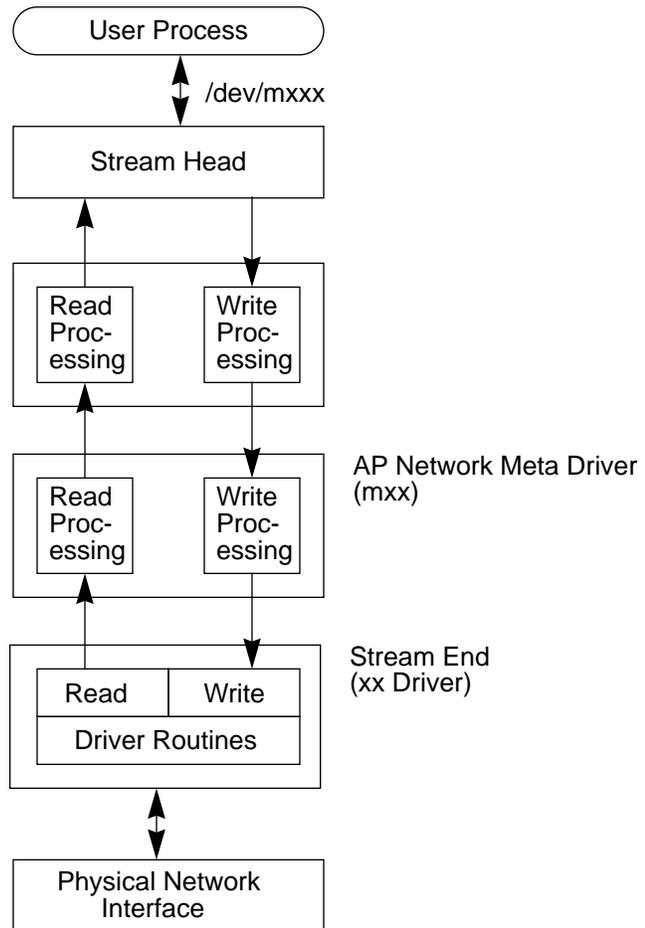
Driver Layers

The following figure illustrates the driver layers that are used when AP controls disk devices:



A user process references a meta-disk, 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 adapter driver). The nexus driver controls the physical device.

The following figure illustrates the driver layers that are used when AP controls networks:



A user process references a meta-network, 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.

Index

A

- active alternate, 1-8
 - alternate path, 1-8
 - alternate pathing, 9-1
 - active alternate, 1-8
 - alternate path, 1-8
 - AP Daemon, 10-1
 - AP database, 1-8
 - AP Librarian, 10-1
 - AP Meta-Drivers, 10-1
 - AP SSP Daemon, 10-2
 - apconfig, 3-2, 4-4
 - apdb, 3-2, 4-4
 - apdisk, 4-3
 - apinst, 4-2
 - apnet, 2-8
 - boot disk, 5-1
 - boot-device devalias, 5-2
 - creating AP database, 3-2
 - creating meta-devices, 4-2
 - creating meta-networks, 6-2
 - creating symbolic links, 4-5
 - database partition, 2-1, 3-1
 - delete database, 3-3
 - delete meta-device, 4-8
 - delete meta-network, 6-6
 - DR / AP interaction, 7-1
 - driver layers, 11-1
 - drvconfig, 4-4
 - meta-device, 1-9, 4-1
 - meta-network, 1-9
 - meta-network names, 6-1
 - path group, 1-9
 - physical path, 1-9
 - primary network considerations, 6-7
 - primary path, 1-9
 - single user mode, 5-4
 - switch, 1-9
 - switching meta-devices, 4-6
 - switching meta-networks, 6-5
 - viewing database information, 3-2
- AP Daemon, 10-1
 - AP database, 1-8
 - AP database, deleting, 3-3
 - AP database, viewing information, 3-2
 - AP Librarian, 10-1
 - AP Meta-Driver, 10-1
 - AP SSP Daemon, 10-2
 - apconfig, 3-2, 4-4
 - apdb, 3-2, 4-4
 - apdisk, 4-3
 - apinst, 4-2
 - apnet, 2-8

B

boot disk and AP, 5-1
boot-device and AP, 5-2

C

commands, list of, 9-1
creating AP database, 3-2
creating meta-devices, 4-2
creating meta-networks, 6-2

D

delete AP database, 3-3
delete meta-device, 4-8
delete meta-network, 6-6
devalias boot-device and AP, 5-2
disk, boot under AP control, 5-1
DR / AP interaction, 7-1
drvconfig, 4-4

E

Ethernet meta-network names, 6-1

F

fault tolerance, 1-1
FDDI meta-network names, 6-2

L

LE meta-network names, 6-1

M

meta-device, 1-9, 4-1
meta-network, 1-9

P

path group, 1-9
physical path, 1-9
primary network and AP, 6-7

primary path, 1-9

S

single user mode and AP, 5-4
switch, 1-9
switching meta-devices, 4-6
switching meta-networks, 6-5
symbolic links, creating for AP, 4-5

V

view AP database information, 3-2