Alternate Pathing 2.0 Reference Manual

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

Part No: 805-0083-10
Revision A, March 1997

NAME

Intro – AP special files

DESCRIPTION

This section describes AP files for your Sun server.

ap(7)	alternate pathing librarian driver, **/dev/ap**
ap_dmd(7)	AP disk meta-driver
ap_nmd(7)	AP network meta-driver group
mbf(7)	FDDI 2.0 special character device; see **ap_nmd**(7)
mhi(7)	HiPPI special character device; see **ap_nmd**(7)
mle(7)	Lance Ethernet special character device; see **ap_nmd**(7)
mnf(7)	FDDI 3.0.x and 4.x special character device; see **ap_nmd**(7)
mqe(7)	Quad Ethernet special character device; see **ap_nmd**(7)

NAME

ap – alternate pathing librarian driver, / dev / ap

SYNOPSIS

ap

DESCRIPTION

The AP driver provides a pseudo-driver interface to the kernel Alternate Pathing (AP) Librarian features.

FILES

/kernel/drv/ap	AP driver module
/kernel/drv/ap.conf	AP driver configuration file

SEE ALSO

Alternate Pathing 2.0 User's Guide

ap(1M), **ap_daemon**(1M)

NAME

ap_dmd – AP disk meta-driver

SYNOPSIS

ap_dmd@*target,lun:partition*

DESCRIPTION

The **ap_dmd** driver works with the AP software to support Alternate Pathing for physical devices handled by the ssd SCSI disk driver. See **ssd**(7) in *man Pages(7): Device and Network Interfaces* of the *SunOS Reference Manual*.

The AP feature lets you configure alternate SCSI paths to a physical device. These paths are associated with a *meta-disk device*, which is one of the file system special nodes associated with a particular meta-driver.

ap_dmd allows the AP Librarian, **ap**(7), to configure or unconfigure physical paths to a SCSI device via an interface that allows SET, UNSET, and SWITCH commands. These commands are issued by **ap**(7) at the behest of the user-invoked AP commands and AP daemon. To change the SCSI path information associated with a particular **ap_dmd** device, use **apconfig**(1M), **apdb**(1M) and **apdisk**(1M). For more information, see the *Alternate Pathing 2.0 User's Guide*

All device operations supported by the ssd driver are also valid on **ap_dmd** devices that have been created via AP commands. See the other AP commands for details regarding other components of AP software, and **ssd**(7) man page for information about block/character file accesses, I/O requests, disk partitioning schemes, CD-ROM support, and ioctls.

ERRORS

ENXIO	No physical SCSI path to the target device exists.
Other	For information on other errors, see **sd**(7).

FILES

apdmd.conf	driver configuration file
/dev/ap/dsk/m*n***c***n***t***n***d***n***s***n*	block files
/dev/ap/rdsk/m*n***c***n***t***n***d***n***s***n*	raw files

where **m** identifies the device as a meta-device and:

c*n*	Controller number
t*n*	Target number
d*n*	Logical unit number
s*n*	Slice (partition) number

DIAGNOSTICS

See **ssd**(7) in *man Pages(7): Device and Network Interfaces* of the *SunOS Reference Manual*.

SEE ALSO

Alternate Pathing 2.0 User's Guide

apconfig(1M), **apdb**(1M), **apdisk**(1M), **apnet**(1M), **ap_daemon**(1M) in *man Pages(1M): Alternate Pathing Administration Commands*

ap(7), **ap_nmd**(7) in *man Pages(7): Alternate Pathing Special Files*
ssd(7) in *man Pages(7): Device and Network Interfaces* of the *SunOS Reference Manual*

NAME

ap_nmd, mbf, mhi, mhme, mle, mnf, mqe – AP network meta-driver group

SYNOPSIS

/devices/pseudo/clone@0:mbf – CS6400 server only

/devices/pseudo/clone@0:mhi – CS6400 server only

/devices/pseudo/clone@0:mhme – Enterprise 10000 server only

/devices/pseudo/clone@0:mle – Enterprise 10000 and CS6400 servers

/devices/pseudo/clone@0:mnf – FDDI 3 or 5 on CS6400 servers; FDDI 5 on Enterprise 10000 servers

/devices/pseudo/clone@0:mqe – Enterprise 10000 and CS6400 servers

DESCRIPTION

ap_nmd is a group of multi-threaded, loadable, clonable, STREAMS meta-network device drivers that support the connectionless Data Link Provider Interface, **dlpi**(7), for **bf(7)** (FDDI 2.0), **hi**(7) (HiPPI), **hme**(7) (Sun FastEthernet 2.0), **le**(7) (Lance Ethernet), **nf**(7) (FDDI 3.0.x or 5.x) and **qe**(7) (Quad Ethernet).

NOTE: The SunOS man pages **bf**(7) and **nf**(7) are available only on systems that run Sun's optional FDDI packages; **hme**(7) is available (by mid-1997) on systems that run Sun FastEthernet; **hi**(7) is available only on systems that run the optional HiPPI package.

ap_nmd works with the AP software to support Alternate Pathing for physical network devices.

Device operations of **ap_nmd** are an extension of the operations of the underlying network drivers. **ap_nmd** normally operates as a transparent pass-through module; it neither interprets nor modifies any of the STREAMS DLPI type messages. However, it does intercept and modify the DL_ATTACH_REQ and DL_INFO_ACK messages.

DL_ATTACH_REQ messages are captured and used to drive the initial connection between logical and physical devices. DL_INFO_ACK messages are captured and responded to with a prebuilt response to eliminate the possibility of the message response timing out due to induced message delays.

The cloning character-special device **/dev/m***xx* is used to access all device-specific instances of the **ap_nmd** within the system.

ap_nmd and AP

The **ap_nmd** driver provides an interface to support Alternate Pathing. The APSET interface allows a user to provide a mapping between physical path and logical path. The APUNSET provides an interface to remove a physical-to-logical path mapping and APSWITCH provide a mechanism to switch a logical path from its existing physical path to a new physical path. For a more complete description of the AP capability, see the *Alternate Pathing 2.0 User's Guide*.

ap_nmd and DLPI

The **ap_nmd** driver is a "style 2" Data Link Service provider. All DLPI processing is handled by the underlying physical device driver. See the man page that corresponds to each underlying driver.

ERRORS

EBUSY	An attempt was made to unload a busy device, or to APUNSET an active device
EEXIST	An attempt was made to APSET an existing logical-to-physical mapping and a logical path when the system was out of memory
EIO	An attempt to switch between physical devices failed
ENODEV	No physical mapping exists
ENOMEM	System memory was exhausted during an attempt to create a mapping between a physical path and a logical path

FILES

mbf.conf	Driver configuration file
mhi.conf	Driver configuration file
mhme.conf	Driver configuration file
mle.conf	Driver configuration file
mnf.conf	Driver configuration file
mqe.conf	Driver configuration file
/dev/mbf	bf special character device
/dev/mhi	hi special character device
/dev/mhme	hme special character device
/dev/mle	le special character device
/dev/mnf	nf special character device
/dev/mqe	qe special character device

DIAGNOSTICS

See **hi**(7) in the *HiPPI Reference* (available only with those systems that support the HiPPI option)

le(7), **qe**(7) in *man Pages(7): Device and Network Interfaces* of the *SunOS Reference Manual*.

SEE ALSO

Alternate Pathing 2.0 User's Guide

ap_daemon(1M), **apconfig**(1M), **apdb**(1M), **apnet**(1M) in *man Pages(1M): Alternate Pathing Administration Commands*

ap(7), **ap_dmd**(7), in *man Pages(7): Alternate Pathing Special Files*

hi(7) in the *HiPPI Reference* (provided only with those systems that support HiPPI)

driver.conf(4) in *man Pages(4): File Formats* of the *SunOS Reference Manual*

dlpi(7), **le**(7), **qe**(7) in *man Pages(7): Device and Network Interfaces* of the *SunOS Reference Manual*

bf(7), **nf**(7) in the **FDDI Reference Manual** (provided only with those systems that support FDDI)