HP-UX
CE Handbook

for

Series 800 HP Precision Architecture-RISC Computer Systems

HEWLETT PACKARD

HP Part No. 5961-8364
Printed in USA May 1993

HP Internal Use Only
E0593

Notice

Hewlett-Packard Company
Systems Support Division
Mountain View Training Center
Attn: George Taft, MS 36UM
100 Mayfield Avenue
Mountain View, California 94043

Printing History

New editions are complete revisions of the manual. Update packages, which are issued between editions, contain additional and replacement pages to be merged into the manual by the customer. The dates on the title page change only when a new edition or a new update is published. No information is incorporated into a reprinting unless it appears as a prior update; the edition does not change when an update is incorporated.

The software code printed alongside the date indicates the version level of the software product at the time the manual or update was issued. Many product updates and fixes do not require manual changes and, conversely, manual corrections may be done without accompanying product changes. Therefore, do not expect a one-to-one correspondence between product updates and manual updates.

Edition 1 . . . . June 1991

Edition 2 . . . . May 1993

Safety Considerations

Review the hardware documentation to become familiar with safety markings used on the product. The following list shows some of the safety symbols used to indicate various safety considerations.

SAFETY SYMBOLS

Instruction manual symbol. The product will be marked with this symbol when it is necessary for the user to refer to the instruction manual in order to protect the product against damage.

Indicates hazardous voltages.

Indicates earth (ground) terminal (sometimes used in manual to indicate circuit common connected to grounded chassis).

Warning

The WARNING sign denotes a hazard. It calls attention to a procedure, practice, or the like, which, if not done correctly or adhered to, could result in injury. Do not proceed beyond a WARNING sign until the indicated conditions are fully understood and met.

Caution

The CAUTION sign denotes a hazard. It calls attention to an operating procedure, practice, or the like, which, if not done correctly or adhered to, could damage or destroy part or all of the product. Do not proceed beyond a CAUTION sign until the indicated conditions are fully understood and met.

Contents

7. Cookbook Procedures

A. Other HP-UX Information Sources

Figures

1

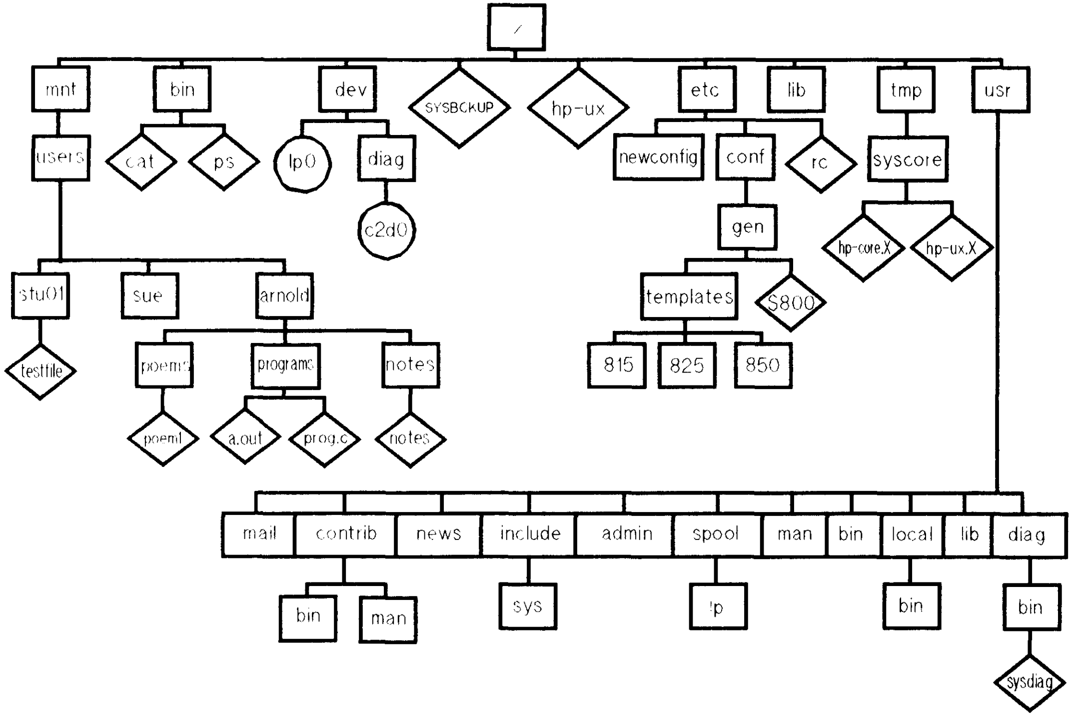
HP-UX Basics

Introduction

This handbook provides fundamental reference information about the HP-UX operating system.

The information presented here is in abbreviated format, and is, largely drawn from course instruction materials in HP-UX System Administration as presented by the Systems Support Training Department of Systems Support Division.

Users of this handbook are encouraged to seek other sources of information to acquire a broader understanding of HP-UX. The handbook is not a substitute for formal training, or self-initiated learning about HP-UX. Rather, it is assumed that to use this handbook, you will already be knowledgeable of basic HP-UX operating system principles.



UNXA0111

11/89

Figure 1-1. HP-UX File Hierarchy

HP-UX Directory Structure

HP-UX Directory Structure

Directory Name	Contents and Use
/	Root
/bin	Public commands
/dev	Special device files
/etc	Commands and files for system administration
/etc/conf	Contains object code for driver generation and system configuration
/etc/conf/gen	Contains the S800 file
/etc/newconfig	Contains new versions of configuration files and scripts after an update
/lib	Contains object code libraries and related utilities
/hp-ux	Contains the HP-UX operating system (kernel)
/tmp	Contains temporary files and system panic information
/mnt	User home directories
/usr	Contains less frequently used commands and miscellaneous files
/usr/lib	Overflow for /lib
/usr/mail	Mail directory used for depositing mail files
/usr/man /man1..man8	Contains unformatted man pages
/usr/man/cat1.. cat8,cat1m	Contains formatted man pages
/usr/spool /uucppublic	Used for free access of files by other systems (uucp and LAN)

HP-UX Directory Structure (continued)

Directory Name	Contents and Use
/usr/spool	Spooled (queued) files for various programs
/usr/spool/uucp	Queued work files, lock files, log files, etc. for uucp
/usr/tmp	Alternate place for temporary files
/usr/contrib	Contains contributed files and commands
/usr/contrib/bin	Contains user contributed commands
/usr/contrib/lib	Contains contributed object libraries
/usr/contrib /man	On-line documentation for any contributed files
/usr/news	Contains news items about customer's system.
/usr/diag/bin	On-line diagnostics
/usr/include/sys	Low level (kernel related) C language header files
/usr/lib/uucp	Configuration files for uucp
/usr/adm	System administrative data files

Command Structure

$ command [options] [parameters]

- White space is used by HP-UX as the delimiter between the command, any options, or parameters.
- Most commands have several options.
- Most commands require at least one parameter.
- Example:

  ls -a /mnt/users/stu01

Full Path Names

- Full path names always begin from the root (/) directory.
- Full path names ALWAYS start with a / (slash).
- Examples:

  /etc/conf/gen/S800

  /etc/mount

  /mnt/users/stu01

Relative Path Names

- The path specified is with reference to the directory where you currently reside in the hierarchical file system.
- Use the HP-UX *pwd* command if you are unsure of your location in the file system.
- Relative path names must begin with one of the following:

Metacharacter or Name	Meaning
.	Path begins with current directory.
..	Path begins with parent directory.
filename or subname	Path begins at the current directory with the file *filename*, or the subdirectory named *subname*.

HP-UX Commands

File Commands

HP-UX File Commands

Command	Description
more file2	Displays the contents of file2 on screen
q	Quits display and returns to command line when using *more* command
RETURN	Displays one more line when using *more* command
SPACE	Displays another screen when using *more* command
h	Displays *help* menu when using *more* command
cat file1	Displays the contents of file1 on screen
cat > newtest	Takes whatever is typed at the terminal and puts it into the new file *newtest* until Ctrl-d is typed
cat >> oldtest	Takes whatever is typed at the terminal and adds it to the end of the existing file *oldtest* until Ctrl-d is typed
cat file1 file2 > file3	Combines *file1* and *file2* and puts them in *file3* with *file1* first
grep pattern file4	Displays the lines in which the string *pattern* occurs in *file4*
cp file5 filenew	Makes a copy of the file *file5* in *filenew*. If *filenew* is a directory, a copy of *file5* is put in that directory.
mv blue green	Changes the name of the file *blue* to *green*. If *green* is a directory, the file *blue* is moved into it.

HP-UX File Commands (continued)

Command	Description
rm useless1 useless2	Deletes the files *useless1* and *useless2*
lp file6	Sends the file *file6* to the default system line printer
vi file7	Creates or edits the file *file7* with the vi screen editor
cmp file1 file2	Does a binary compare of *file1* and *file2*
diff file1 file2	Displays the difference between ASCII *file1* and ASCII *file2* on screen
chown stuxx file1	Changes ownership of file *file1* to stuXX
chgrp ces files1	Changes the group ID of *file1* to ces
chmod 755 file1	Changes the (r)ead-(w)rite-(e)xecute file permissions, for owner-group-other, of file *file1* to rwxr-xr-x

Directory Commands

HP-UX Directory Commands

Command	Description
ls	Lists the files and subdirectories of the current directory
lsf	Lists the files and subdirectories of the current directory; flags directories with a (/) and executable files with a (*)
ll	Long listing of the current directory; shows file type, permissions, ownership, size, etc.
lssf /dev/*	Displays all device files in */dev*; shows physical addresses and file descriptions (S800 only)
file *	Lists all files in current directory and attempts to show file type
pwd	Displays the name of the working directory on screen
cd	Returns you to your home directory
cd /user/stu01	Moves you to the directory */user/stu01*
cd ..	Moves you to your working directory's parent directory
mkdir servicenotes	Creates a new subdirectory in your current directory named *servicenotes*
rmdir letters	Deletes the directory *letters*, if the directory contains no files
rm -r *	Recursively removes all files in the current directory (know what directory you are in before typing this command).
find / -name "*cat*"	Searches all mounted file systems for a file name which includes the pattern *cat* anywhere in it. Once found, *find* displays the path name associated with the file

System Commands

HP-UX System Commands

Command	Description
who	Displays the users currently logged onto the system and the ports used
who -r	Displays the current system run-state
ps -ef	Displays all processes executing on the system; shows PIDs, PPIDs, etc.
man ls	Displays information about the *ls* command and its options
man -k mail	Lists the HP-UX commands that relate to the keyword *mail*
kill 4507	Terminates the process associated with process ID number 4507
exit	logout
bdf	Shows disk usage and percentage full
diskinfo /dev/rdsk/c1d0s2	Displays model number of disk associated with disk special file */dev/rdsk/c1d0s2*
lpstat -t	Shows status of spooler
write	Writes to users already logged on to system
wall	Broadcasts system wide announcement to all users
echo message	Echoes ASCII message *message* on screen

HP-UX System Commands (continued)

Command	Description
init s	Changes run-state from multiuser to single user
init 2	Changes run-state from single user to multiuser
mount	Lists what file systems are mounted
mount -a	Attempts to mount all file systems listed in /etc/checklist
umount /dev/dsk/c1d0s11	Manually unmounts /dev/dsk/c1d0s11 file system
cd /dev; insf	Assigns logical unit number(s) to new device(s) found by *ioscan*. Makes device files for new device(s).
cd /dev; insf -e	Remakes device files for existing devices
ioscan -f	Probes hardware. Binds drivers to hardware if the drivers are present in the kernel. Lists resulting hardware and path information.
ioscan -kf	Does not scan hardware. List hardware and path information for devices known to the kernel.
rmsf -H BC/X.Y.Z.U	Removes device file(s) and logical unit number assignment in the kernel for a device on path *BC/X.Y.Z.U*
cd /dev; insf -H BC.X.Y.Z.U -l lu	Assigns logical unit number *lu* to device at path *BC.X.Y.Z.U*. Makes device file for this path.
tset hp2392	Initialize hp2392 terminal. Sets backspace, tabs, etc.
suplicen	Installs a diagnostic password. Valid until user logs off.
swapinfo	Displays where system can swap

Command Keys

HP-UX Command Keys

Command	Description
Ctrl-c or DEL	Interrupt. Stops a command from being executed
Ctrl-d	Removes you from the current environment. At the $, #, or % prompts, these keys log you off the system (if you are in your primary shell)
Ctrl-s	Temporarily stops output to the display
Ctrl-q	Resumes the output that was halted by Ctrl-s

Wild Card Characters

HP-UX Wild Card Characters

Command	Description
*	Designates all files in the current directory
s*	Designates all files beginning with s in the current directory
*.c	Designates all files ending with .c in the current directory
????	Designates any 4-character filename in the current directory
s???	Designates any 4-character filename beginning with s in the current directory

- Lists the contents of a directory or file in long format

Example listing for ordinary or directory files:

 # ll /

mode	# of links	owner	group	size in bytes	time of last modification	file name
-rwxr-xr-x	1	root	root	1509376	Feb 3 18:06	hp-ux
drwxr-xr-x	2	root	other	4096	Feb 3 17:10	bin
drwxr-xr-x	103	root	other	4096	Feb 3 17:36	system

Example listing for special device files:

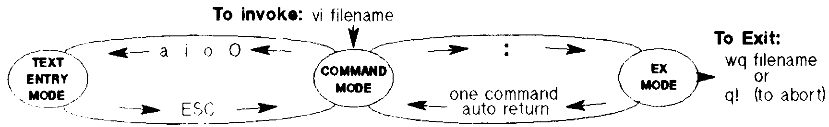
 # ll /dev/tty1p0

mode	# of links	owner	group	major number	minor number	time of last modification	file name
crw--w--w-	1	root	root	1	0x000100	Feb 3 17:47	/dev/tty1p0

UNXA0124

3/89

Figure 1-2. ll Command



TEXT ENTRY MODE

a - Append after cursor
i - Insert before cursor
o - Open new line after cursor
O - Open new line before cursor
ESC – Ends text entry mode

COMMAND MODE

U - Restore line to original state
u - Undo last command

Deleting Text:
dd - Delete line at cursor (ndd deletes 'n' lines)
x - Delete character at cursor
dw - Delete word at cursor
r - Replace character at cursor
('r' deletes char; next char typed replaces)

Cursor Movement:
h - left j - down k - up l - right

Search:
/pattern - Forward search for 'pattern'
n - Search forward again for 'pattern'
N - Search backward again for 'pattern'

Copy Line(s):
nY - Yank 'n' lines into buffer (start at cursor)
P - Put yanked lines above current cursor position

EX MODE

Global Search/Replace:
g/str1/s/str1/str2/ - Find line(s) containing 'str', replace first pattern 'str1' on that line with 'str2'

Save Text and Exit:
wq or x - Save file then exit

Write Text to File:
w filename - Write to 'filename'

Overriding Protection:
cmd! - ! following ex command overrides vi or file protect (you must be superuser)
examples:
w! - writes to protected file
q! - quits without forcing write (aborts vi)

Position Cursor to Line Number
- '#' is desired integer line number

UNXA0130 3/93

Figure 1-3. Using vi

ed in Review

- Modifying Text Within a Line

s/old pattern/new pattern/	Substitutes first occurrence of *old pattern* in a line with *new pattern*.
s/old pattern/new pattern/g	Substitutes all occurrence of *old pattern* in a line with *new pattern*.

- Moving Lines

General format: *x,ymz*

x	Beginning line argument.
y	Ending line argument.
m	Move command.
z	Destination argument (followed by the number of lines to move).

Example: *3,10m50*

Lines 3 through 10 move after line 50 (lines are automatically renumbered after the move).

- Searching a File for a Pattern

/pattern/	Forward Search.
?pattern?	Backward Search.
/^pattern/	Search forward for a *pattern* beginning a line.
/pattern$/	Search forward for a *pattern* ending a line.

- Making Commands Effective Globally

General format: *g/pattern/command list/g*

x	Beginning line argument.
y	Ending line argument.
g	Global command informs ed to perform *command list* on every line containing *pattern* in the file.
command list	List of ed commands to be performed on first *pattern* match on each line.
g	Informs ed that the *command list* is to be performed on all *pattern* matches on line.

- Line Arguments

General format: *x,y command list*

x	Beginning line argument.
y	Ending line argument.
command list	List of ed commands to be performed.

- Common Arguments

.	Current line.
+	Move forward one line.
-	Move back one line.
$	Move to last line in file.
1	Move to first line in file.
/pattern/	Forward search for a pattern.

Examples:

*1,$*command list	Perform command list on all lines in a file (same as *g* argument in *Making Commands Effective Globally*).
*1,$-3*command list	Perform command on all lines except last four.
/HP/,/products/ command list	Performs commands on all lines beginning with pattern *HP* and ending with line containing *products*.

- Invoking ed

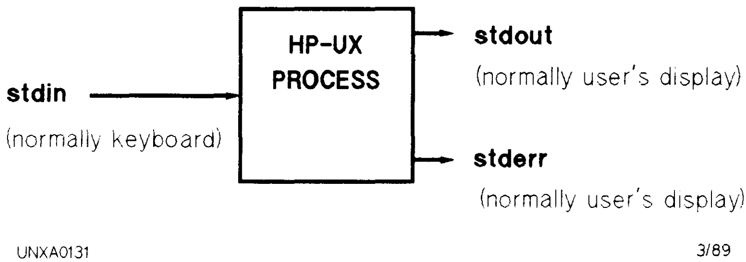
$ *ed*

$ *ed* oldfile

- Basic Commands

a	Append lines of text after current line (single period <.> at beginning of a line ends append mode).
i	Insert lines before current line (single period <.> at beginning of a line ends insert mode).
d	Delete line of text at cursor.
p	Print line.
g	Perform command list on selected lines of entire file.
s	Modify text on a line by substitution.
m	Move line(s) to new position in file (lines are renumbered after move.
n	Print line number of current line.
r	Read specified file into current line after the addressed line.
h	Terse help message.
u	Undo last command.
w	Write ed buffer to specified file (for example *w newfile*).
q	Quit ed.
qq	Abort ed without writing.

HP-UX Processes



- Work in HP-UX is accomplished within a process. A unique process is created for each command or program executed.
- Process I/O.
  - Input is taken from the standard input file (stdin).
  - Output is directed to the standard output file (stdout).
  - Any errors generated are directed to standard error file (stderr).
- Each process has a unique Process ID (PID) number to identify the process to the HP-UX kernel.

Observing Process Status

- Use *ps -ef* to Display all System Processes

$*ps -ef*

```
UID     PID  PPID  C     STIME  TTY       TIME COMMAND
root     89     1  0    Nov 18  console   0:00 -sh
root      3     0  0    Nov 18  ?         0:01 statdaemon
root      2     0  0    Nov 18  ?         0:00 pagedaemon
root      1     0  0    Nov 18  ?         1:40 init
root      0     0  0    Nov 18  ?         0:04 swapper
root   5965    89  0  09:28:30  console   1:00 ps ef
root   5964    89  0  09:27:58  console   0:00 sleep 100
```

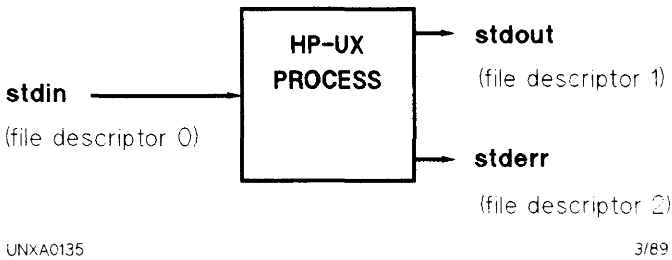
Field	Meaning
UID	User ID
PID	Process ID
PPID	PID of parent process that spawned this process
C	Processor utilization for scheduling
STIME	Starting time of process. Displays date if >24 hours
STTY	Sets or displays terminal I/O options
TIME	Cumulative CPU execution time in minutes and seconds
COMMAND	Nearest approximation of command typed that fits field

- Use *ps* to Display all User Processes $*ps*

```
 PID TTY      TIME COMMAND
  89 tty0p2   0:00 sh
5960 tty0p2   0:00 ps
```

Field	Meaning
PID	Process ID
TTY	Terminal process started on
TIME	Cumulative CPU execution time in minutes and seconds
COMMAND	Nearest approximation of command typed that fits field

Redirecting I/O



- Redirecting Standard Output.
  - Use > or *1>* to create or overwrite the specified output file. For example: *$cat file > newfile*
  - Use >> or *1>>* to append output to the end of the specified output file. For example: *$cat file >> appendedfile*
- Redirecting Standard Error.
  - Use *2>* to create or overwrite the specified error file.

    For example: *$cat file 2> errorfile*
  - Use *2>>* to append error output to the end of the specified error file. For example: *$cat file 2>> errorlogfile*
- Redirecting Standard Input.
  - Use < or *0<* to accept input from the specified input file. For example: *$mail < formletter*

Running a Process in the Background

- Use the *&* (ampersand) character following the command, options, and arguments to place a process in background.

  $sleep 90 &
  5964
  $
  $*ps*

PID	TTY	TIME	COMMAND
89	tty0p2	0:00	sh
5968	tty0p2	0:00	ps
5964	tty0p2	0:00	sleep 90

- Use *kill* along with the Process ID number to terminate a process before it completes execution.

  $*kill 5964*

Connecting Processes with Pipelines

- Pipelines connect the standard output (stdout) of one process to the standard input (stdin) of another process.
- The symbol | (vertical bar) is the pipe symbol. The standard output of the process to the left of | becomes standard input to the process on the right of |.
- The HP-UX kernel handles necessary buffering.
- Examples:

$*who | wc -l*	Counts the number of users on the system.
$*ls -a1R / | wc -l*	Counts the number of files on the system.
$*cat /etc/passwd | lp*	Directs */etc/passwd* to the default line printer.

Filters

- Filters take input, perform some filtering action, and finally output data according to the filtering criteria.
- Examples:
  - □ *$grep stu01 /etc/passwd*
    stu01:aq3jpzX:201:200::/mnt/users/stu01:/bin/sh

    Grep examines all lines in */etc/passwd*. Lines not containing *stu01* are filtered out.
  - □ *$who | sort*

stu01	tty0p1	Jul	20	15:48
stu02	tty0p5	Jul	20	11:32
stu04	tty1p0	Jul	20	08:31

    Sort filtered *who* output into alphabetical order by login name.

HP-UX Login Process

1. /etc/getty
   - Prints */etc/issue* at logoff.
   - Issues login prompt (uses */etc/gettydefs*).
   - Reads login name.
   - Invokes */bin/login* command.
2. /bin/login
   - Checks login name and password (uses */etc/passwd*).
   - Updates accounting files.
   - Sets working directory.
   - Invokes command given in last field of user entry in */etc/passwd*.
3. Initialize Shell
   - Bourne Shell (*/bin/sh*).
   - C Shell (*/bin/csh*).
   - Korn Shell (*/bin/ksh*).
   - Restricted Shell (*/bin/rsh*).

The */etc/gettydefs* File

- Fields: *label # initial-flags # final-flags # login-prompt # next-label*

Where:

label	-	Identifies the entry.
	-	Matches against "getty" speed argument.
initial-flags	-	Initial line and terminal settings (speed must be specified).
final-flags	-	Final line and terminal settings (speed must be specified).
login-prompt	-	Initial login prompt printed on the terminal.
next-label	-	Entry to try next if "break" is typed.

The */etc/passwd* File

- Fields:

user_name:passwd:user_id:group_id:comment:login_dir:cmd

Where:

user_name	-	User's login name.
password	-	User's password in encrypted form.
	-	Optional password aging sub-field.
user_id	-	Unique integer value between 1 and 6000.
	-	If user_id is zero, user has superuser capabilities.
group_id	-	Integer value identifying the group.
comment	-	User's full name and other ID information.
login_dir	-	Full path to user's login directory.
cmd	-	Command to execute at login.
	-	Usually a shell is invoked.
	-	Default is */bin/sh*.

The */etc/group* File

- Fields:

group_name:password:group_id:members

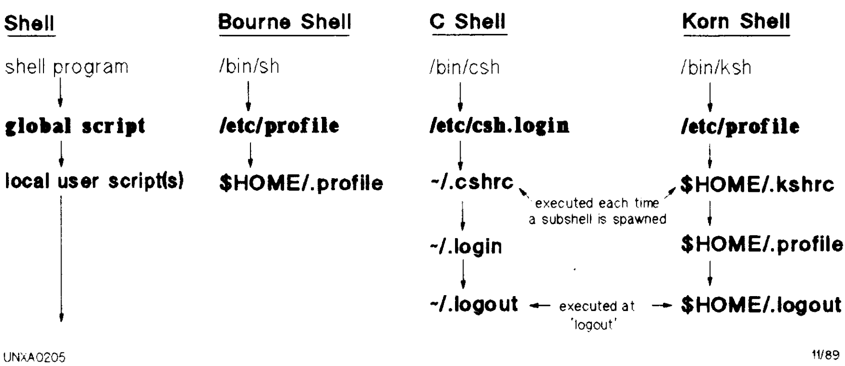
Where:

group_name	-	Contains the name of the group.
password	-	Generally not used and usually remains null.
group_id	-	Unique integer value identifying the group.
members	-	Comma separated list of members in the group.
	-	List of users who can change to the group with *newgrp*.

Shell Initialization

- */bin/login* invokes the user's shell. The three shells most frequently used are the Bourne Shell (*/bin/sh*), C Shell (*/bin/csh*), and Korn Shell (*/bin/ksh*).
- Once invoked, each shell executes customized scripts to set up the global user environment, and any user-created local environment.

  The initialization flow is illustrated below:



Bourne Shell

The */etc/profile* File

Example:

```
# @(#) $Revision: 66.6 $
# Default (example of) system-wide profile file (/bin/sh initialization).
# This should be kept to the bare minimum every user needs.
          trap "" 1 2 3                                       # ignore HUP, INT, QUIT now.

          PATH=/bin:/usr/bin:/user/contrib/bin:/usr/local/bin   # default path
          MANPATH=/usr/man:/usr/contrib/man:/usr/local/man      # default path

          if [ -r /etc/src.sh ]
          then
               . /etc/src.sh                                  # set the timezone
               unset SYSTEM NAME
          else
               TZ=MST7MDT                                     # change this for local time.
               export TZ
          fi

          if [ "$TERM" = "" ]                                 # if term is not set,
          then                                                #
               TERM=hp                                        # default the terminal type
          fi

          export PATH MANPATH TERM

# Set erase to ^H

          stty erase

# Set up shell environment:

          trap "echo logout" 0

# This is to meet legal requirements . . .
          cat /etc/copyright
          if [ -r /etc/motd ]
          then
               cat /etc/motd                                  # message of the day
          fi

          if [ -f /bin/mail ]
          then
               if mail -e                                     # notify if mail.
               then echo "You have mail."
               fi
          fi

          if [ -f /usr/bin/news ]
          then news -n                                        # notify if new news.
          fi

          if [ -r /tmp/changetape ]                           # might wish to delete this.
          then   echo "\007\nYou are the first to log in since backup:"
                 echo "Please change the backup tape.\n"
                 rm-f /tmp/changetape
          fi
          trap 1 2 3                                          # leave defaults in user environment.
```

UNXA0206 3/93

Figure 1-4. */etc/profile* **Example**

Default Variable Settings

- Variables set by */bin/login*:

HOME	-	Contains default argument (login directory) for *cd* command.
MAIL	-	Set to the name of the user's mail file. If mail arrives in the specified file, the shell notifies the user of its presence.
SHELL	-	Set to last field of user's */etc/passwd* entry (normally Shell).
LOGNAME	-	First field of user's */etc/passwd* entry.

- Variables Set by the Bourne Shell:

PATH	-	Contains the search path for commands.
PS1	-	Contains primary prompt string. Default is *$* (dollar sign).
PS2	-	Contains secondary prompt string. Default is > (greater than sign).
MAILCHECK	-	A colon (:) separated list of file names. If mail arrives in any of these files, the user is notified. Overrides MAIL if set.
IFS	-	Internal field separators, normally space, tab, and newline.

Examining Variables with *set*

```
$ set
HOME=/mnt/users/stu13
IFS=

LOGNAME=stu13
MAIL=/usr/mail/stu13
MAILCHECK=600
PATH=/bin:/usr/bin:/usr/contrib/bin:/usr/local/bin
PS1=$
PS2=>
SHELL=/bin/sh
TERM=hp
TZ=PST8PDT
$
```

A *.profile* Example

```
.
.
# To search your HOME directory:
.
PATH=$PATH:/mnt/users/stu13
.
# To change your primary system prompt:
.
PS1='MY NAME: '
```

Note

- Rather than construct a *.profile* file, the system administrator is encouraged to copy */etc/d.profile* into the user's HOME directory. The user can customize this script to meet login needs.
- To copy the file, type the following:

  $ *cp /etc/d.profile $HOME/.profile*

The */etc/csh.login* File

Example:

```
# Default (example of) system-wide profile file (/bin/csh initialization
# This should be kept to the bare minimum every user needs.

        # default path for all users
        set path=(/bin /usr/bin /usr/contrib/bin /usr/local/bin)
        set prompt="[\!] % "

        # default MANPATH
        setenv MANPATH /usr/man:/usr/contrib/man:/usr/local/man.

        if ( -r /etc/src.csh ) then
              source /etc/src.csh                        # set the TZ variable
        else
              setenv TZ MST7MDT                          # change this for local time.
        endif

        if ( ! $?TERM ) then                             # if TERM is not set,
              setenv TERM HP                             #    use the default
        endif

# This is to meet legal requirements . . .
        cat /etc/copyright                               # copyright message

# Miscellaneous shell-only actions

        if ( -f /etc/motd ) then
              cat /etc/motd                              # message of the day.
        endif

        if ( -f /bin/mail ) then
              mail -e                                    # notify if mail.

              if ( $status == 0 ) echo  "You have mail."
        endif

        if ( -f /usr/bin/news ) then
              news -n                                    # notify if new news.
        endif

        if ( -r /tmp/changetape ) then                   # might wish to delete this :
              echo
              echo "You are the first to log in since backup:"
              echo "Please change the backup tape.\n"

              rm -f /tmp/changetape
        endif

   UNXA0231                                                3/90
```

Figure 1-5. */etc/csh.login* **Example**

The ~/*.cshrc* File

```
# Default user .cshrc file ( /bin/csh initialization )
# Usage:  Copy this file to a user's home directory and edit it to
# customize it to taste.  It is run by csh each time it starts up.
# Set up default command search path:

    set path=( /bin /usr/bin )

# Set up C shell environment:

    if ( $?prompt ) then                # shell is interactive.
      set history=20                    # previous commands to remember.
      set savehist=20                   # number to save across sessions.
      set system=`hostname`             # name of this system.
      set prompt = "$system  \!: "      # command prompt.

      # Sample alias:

      alias        h        history

      # More sample aliases, commented out by default:

      # alias      d        dirs
      # alias      pd       pushd
      # alias      pd2      pushd +2
      # alias      po       popd
      # alias      m        more
    endif
UNXA0232                                                    6/89
```

Figure 1-6. The ~/*.cshrc* File

The ~/.login File

```
# Default user .login file ( /bin/csh initialization )

# Set up the default search paths:
set path=(. /bin /usr/bin /usr/contrib/bin /usr/local/bin)

# set up the terminal
eval `tset -s -Q -m ':?hp' `
stty  erase "^H"  kill "^U"  intr "^C"  eof "^D"
stty  hupcl  ixon  ixoff
tabs
hp9000s800                  # Job control available on hp9000s800 only
if ( "$status" == "0" ) then
      stty  sup "^Z"  tostop
endif
# Set up shell environment:
set  noclobber
set  history=20
UNXA0233                                                    6/89
```

Figure 1-7. The ~/.login File

Setting up C Shell *history* Command

- The following sets up the C Shell *history* command:
  - To set the size of the history event buffer (this is normally done by the *~/.cshrc* file at login):

    % set history = listsize
    (where *listsize* is an integer between 10 and 20)
  - To set the number of events to be saved at logout and restored on login (this is normally done by the *~/.cshrc* file at login):

    % set savehist = listsize
    (where *listsize* is an integer between 10 and 20)
  - Set the login prompt to reflect history event numbers (this is normally done by the *~/.cshrc* file at login):

    % set prompt = "[\!] %"
  - To save typing, alias the history command to h (this is normally done by the *~/.cshrc* file at login):

    % alias h history

Examining the *history* Event Buffer

- The user's command sequence:

  [1]% ls
  testfile file1 file2
  [2]% pwd
  /mnt/users/stu01
  [3]% mv tesfile testfile.bak
  mv: tesfile : Cannot access : No such file or directory
  [4]%

- To view the event buffer:

  [4]% history
  1 ls
  2 pwd
  3 mv tesfile testfile.bak
  [5]%

Reusing Commands in the Event Buffer

- An explanation mark (!) is used to reference a command for reuse.
- The examples shown below assume the following event buffer contents:

  [4]% history
  1 ls
  2 pwd
  3 mv tesfile testfile.bak
  4 history
  [5]%

- Referencing and executing the last command using two explanation marks (!!):

  [5]% !!
  history
  1 ls
  2 pwd
  3 mv tesfile testfile.bak
  4 history
  5 history
  [6]%

- Referencing and executing a buffer event using an explanation mark (!) and the event number:

  [6]% !1
  ls
  testfile file1 file2
  [7]%

- Referencing and executing a buffer event by using an explanation mark (!) and a relative location number:

  [7]% !-3
  ls
  testfile file1 file2
  [8]%

- Referencing and executing a buffer event by using and explanation mark (!) and event text:

  [8]% !p
  pwd
  /mnt/users/stu01
  [9]%

Modifying Previous Events

- Correcting the previous event:

  Syntax: ^oldstring^newstring^ (to replace *oldstring* with *newstring*; omit the final circumflex (^) when a carriage return is used).

  Example:

  [1]% ls
  testfile file1 file2 file30
  [2]% mv tesfile testfile.bak
  mv : tesfile : Cannot access : No such file or directory
  [3]% ^tesfile^testfile.bak
  [4]% mv testfile.bak
  [5]%

- Correcting an event by string substitution:

  Syntax: !listnumber:s/oldstring/newstring (to substitute the first occurrence of *oldstring* with *newstring*).

  Example:

  [1]% ls
  testfile file1 file2 file3
  [2]% mv tesfile testfile.bak
  mv : tesfile : Cannot access : No such file or directory
  [3]% pwd
  /mnt/users/stu01
  [4]% !2:s/tes/test
  mv testfile.bak
  [5]%

- Correcting an event using global string substitution:

  Syntax: !listnumber:gs/oldstring/newstring (to substitute all occurrences of *oldstring* for *newstring*).

  Example:

  [1]% ls
  testfile file1 file2 file3
  [2]% mv tesfile testfile.bak
  mv : tesfile : Cannot access : No such file or directory
  [3]% pwd
  /mnt/users/stu01
  [4]% !2:gs/tes/test
  cp testfile.bak
  [5]%

Modify an Event and Print Without Execution

- Printing a modified event without execution:

  Syntax: !listnumber:p:modification_command(s)

  Example:

  [4]% history
  1 ls
  2 pwd
  3 mv tesfile testfile.bak
  4 history
  [5]% !3:p:s/esf/estf
  mv testfile.bak
  [6]% history
  1 ls
  2 pwd
  3 mv tesfile testfile.bak
  4 history
  5 mv testfile.bak
  6 history
  [7]% !5
  mv testfile.bak
  [8]%

Booting

Initial System Load (ISL)

- Must reside in Section 6 of boot device.
- Operating system independent.
- Used to set default boot paths.
- Used to load the HP-UX kernel.
- Diagnostic information:
  - Error messages on console.
  - Error codes on display panel.
  - See ISL(1M) for message detail.
  - See hpux_800(1M).

ISL> help

```
?            Help facility
HELP         Help facility
LISTF        List ISL Utilities
LS           List ISL Utilities
AUTOBOOT     Set or clear autoboot flag in stable storage
AUTOSEARCH   Set or clear autosearch flag in stable storage
PRIMPATH     Modify primary boot path in stable storage
ALTPATH      Modify alternate boot path in stable storage
CONSPATH     Modify system console path in stable storage
DISPLAY      Display boot and console path in stable storage
LSAUTOFL     List contents of autoboot file
LISTAUTOFL   List contents of autoboot file
FASTSIZE     Sets or displays FASTSIZE
SUPPORT      Boot the Support Tape from the boot device
SUPPORTCD    Boot the Support Tape from the CDROM
READNVM      Display contents of one word of NVM
READSS       Display contents of one word of stable storage
LSBATCH      List the contents of batch file
LISTBATCH    List the contents of batch file
BATCH        Execute commands in the batch file
LSEST        List the contents of EST (Extended Self Test) file
LISTEST      List contents of EST (Extended Self Test) file
EST          Execute commands in EST (Extended Self Test) file

Utilities on this system are:
HPUX
RDB
IOMAP

Utilities on the Support Tools Media are:
RECOVERY     SS_CONFIG
CLKUTIL      BCDIAG
A1002AI      A1002AM
A1002AP      A1100AI
A1100AM      A1100AP
EDBC         EDPROC
MPROC        MULTIDIAG
TDIAG        UNIPROC
```

ISL> *hpux* Utility

ISL>hpux (BC/X.Y.Z.U;S)kernelfile

Use for normal system start-up.

ISL>hpux set autofile "hpux (BC/X.Y.Z.U;S)kernelfile"

Use to change the boot string in autofile.

ISL>hpux -lm (BC/X.Y.Z.U;0)kernelfile

Use to perform maintenance (will not boot) boot on LVM boot disk.

ISL>hpux -lq (BC/X.Y.Z.U;0)kernelfile

Use to override root volume group quorum requirements when booting an LVM bootable disk.

ISL>hpux -a(C|R|S|D)devicefile (BC/X.Y.Z.U;S)kernelfile

Use to override kernel device(s) (console, root, swap, and dump).

ISL>hpux -is (BC/X.Y.Z.U;S)kernelfile

Use if root password lost or start-up files are suspect.

ISL>hpux ls (BC/X.Y.Z.U;S)/[path]

Use to look for a kernelfile on the specified path.

ISL>hpux -m[p|s|x] (BC/X.Y.Z.U;S)kernelfile

Use to boot from opposite side of mirrored root.

ISL>hpux -F

Use with SWITCHOVER/UX to ignore locks.

Note

Parameters for the ISL *hpux* utility are described on the following two pages.

Where:

Key	Description
kernelfile	Name of kernel to be loaded (typically *hp-ux* or *SYSBCKUP*).
set autofile	Sets autofile to the string enclosed in double quotes.
lm	Boot LVM boot disk from known offset of *LVOL1* without using information in the Boot Data Reserve Area (BDRA). Useful if normal boot fails because of BDRA corruption.
lq	Boot LVM bootable disk without meeting quorum requirements.
devicefile	Hardware path of kernel device.
-a(C|R|S|D)	Configure the device as console, root, swap, or dump device, for example the boot string: *ISL> hpux -aR(BC/X.Y.Z.U;r) -aS(BC/X.Y.Z.U;s) -aD(BC/X.Y.Z.U;d) (BC/X.Y.Z;r)kernelfile* overrides the existing kernel's kernel devices information specifying root on section *r*, swap on section *s*, dump on section *d*. The final argument specifies the location of the kernelfile.
-is	Load kernel in single-user mode. Gives superuser capabilities on the system console without going through */etc/passwd* or using */etc/inittab*.
ls	Lists files on the specified path. Executable files names are appended with an asterisk (*).
path	Path to specified file or directory.
BC/	Bus converter address (PMB or SMB). *BC/* is required on models 85X/860/87X/890 only. *BC/ = 2/* or *6/* on 85X/860/87X. *BC/ = [(4 x PMB slot#) + (interface module#)]* on 890.
X	MIDBUS or HPPB module number. *X* = [(4 x interface slot number) + *interface* module number].
Y	On CIO machines *Y* = CIO interface slot number; on HPPB machines *Y* = HPIB, SCSI, or HPFL device address.
Z	HPIB, SCSI, or HP-FL device address
U	Integer unit number. Used only for SCSI disk and disk arrays.

Key	Description
S	An integer disk section number which is to be searched for *kernelfile*.
-m[-p|-s|-x]	Boot from opposite side of mirrored root (*-mp* = boot from primary; *-ms* = boot from secondary).
-F	Use with SWITCHOVER/UX to ignore any locks on boot disk.

Booting HP-UX on HP 9000 Series 600/800

- From Reset or Transfer of Control

Boot from primary boot path (Y or N)?> n
Boot from alternate path (Y or N)?> n
Enter boot path or ?> BC/X.Y.Z.U (the path to System Disk)

.

Booting.

.

Console IO Dependent Code (IODC) revision 4
Boot IO Dependent Code (IODC) revision 4

.

Interact with IPL (Y or N)?> y

.

Hard Booted.

.

ISL Revision A.00.08 June 22, 1989

.

ISL> hpux (BC/X.Y.Z.U;S)hp-ux

Where:

Key	Operation
BC/X.Y.Z.U	The path to the system disk. See addressing by model number for details.
S	Integer disk section number that contains the kernel file. For Logical Volume Manager (LVM) disks, *S* is always zero (0). For non-LVM file systems, *S* ranges from 0 to 15 (using default non-LVM root disk sectioning, *S* is often 13).
hp-ux	The name of the kernel file (usually *hp-ux* or *SYSBCKUP*).

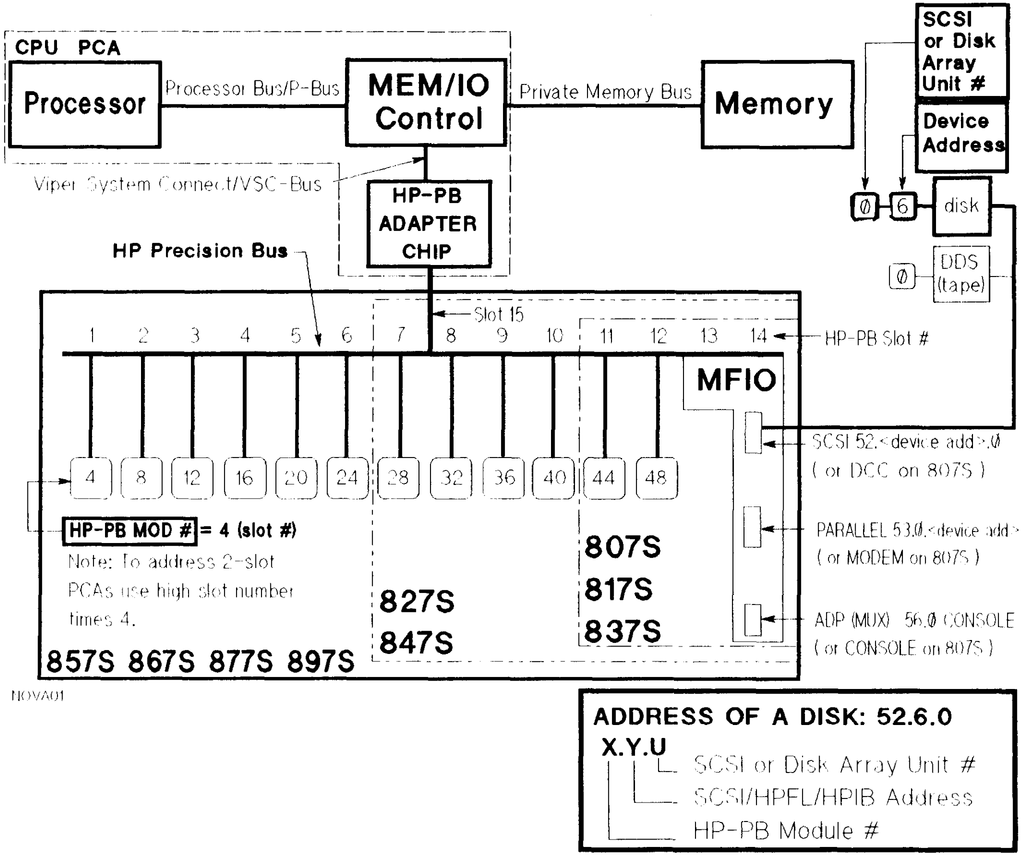


Figure 1-8. 8x7 Addressing

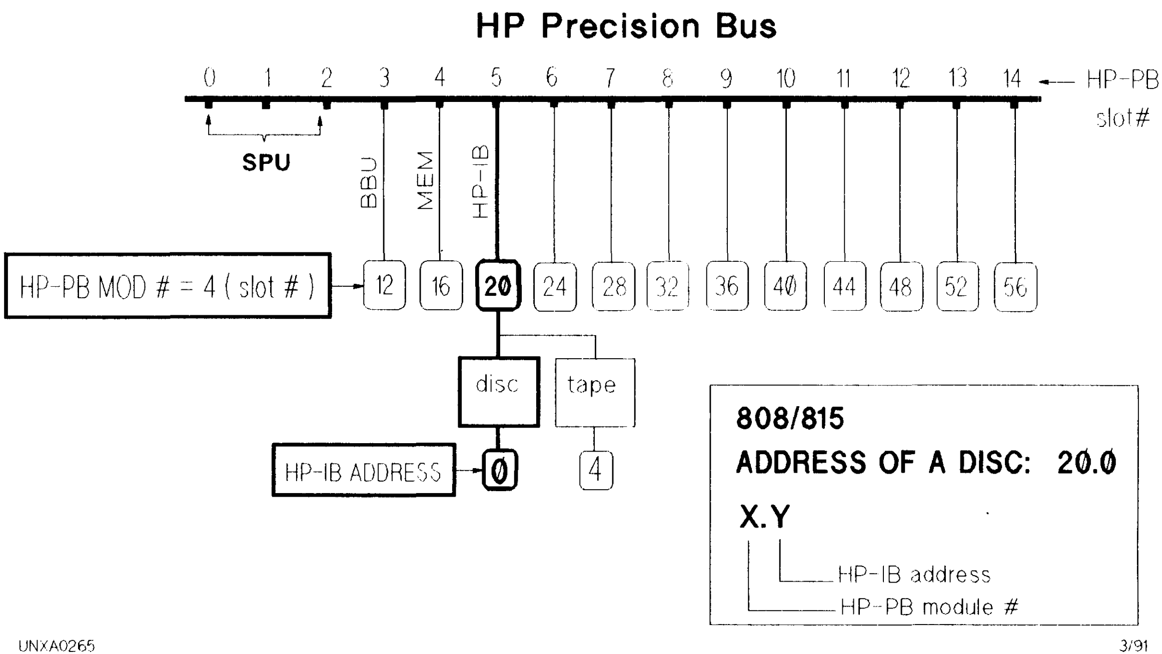


Figure 1-9. 808/815 Addressing

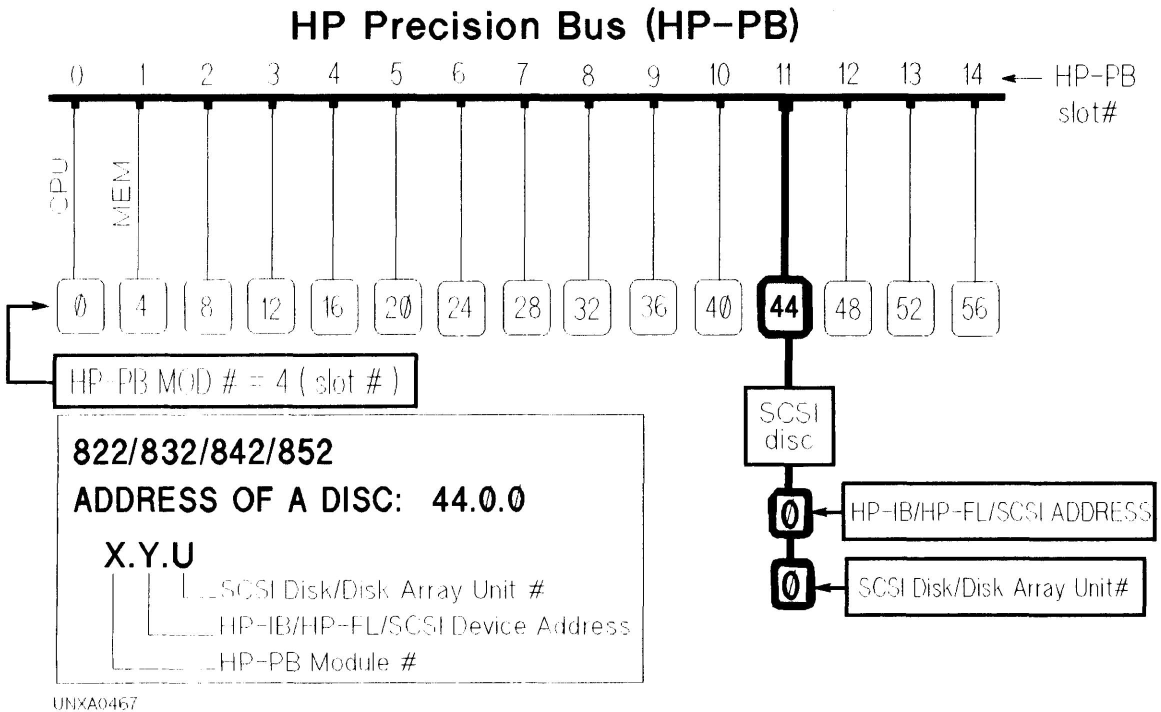


Figure 1-10. 822/832/842/852/642/652 Addressing

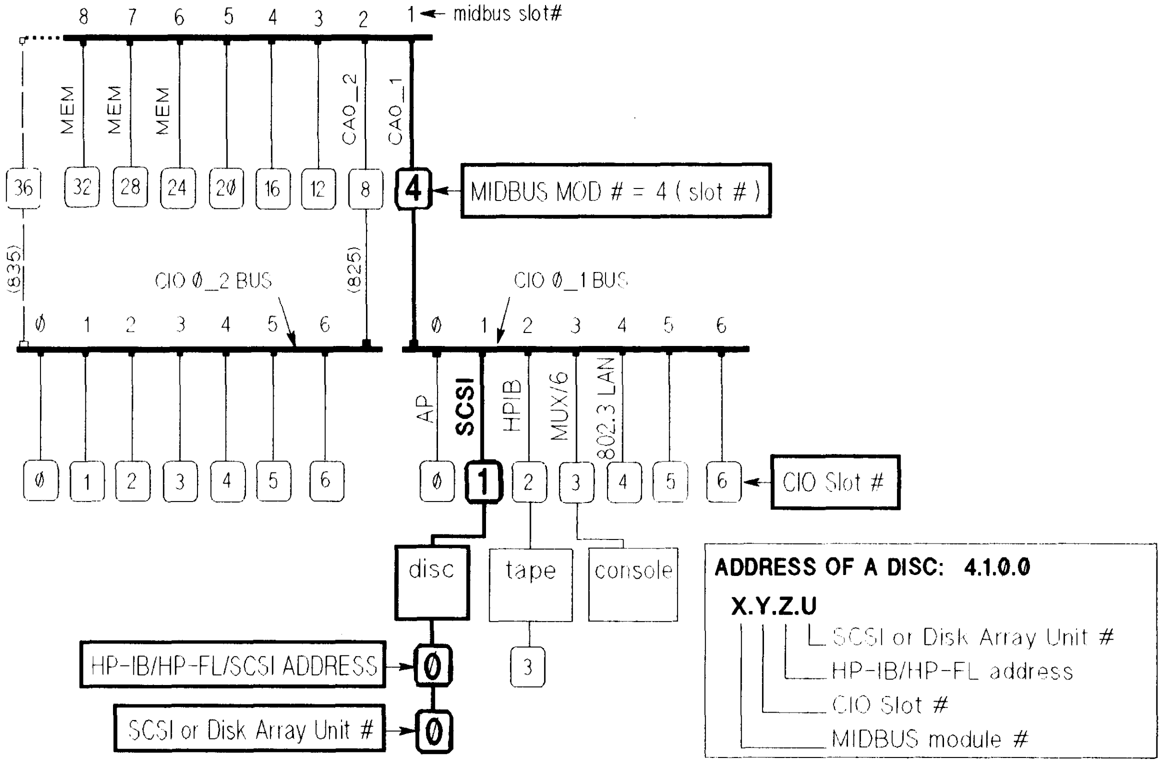


Figure 1-11. 825/835/845 Addressing

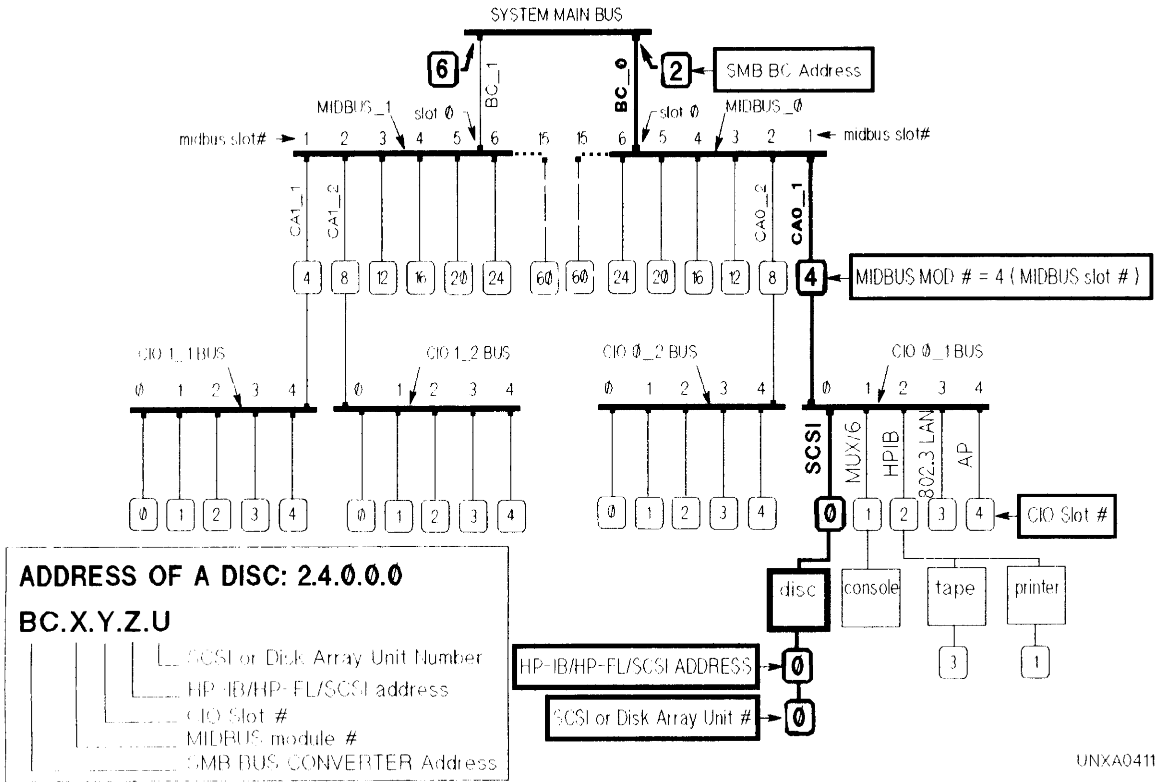


Figure 1-12. 850/855/860/865/870 Addressing

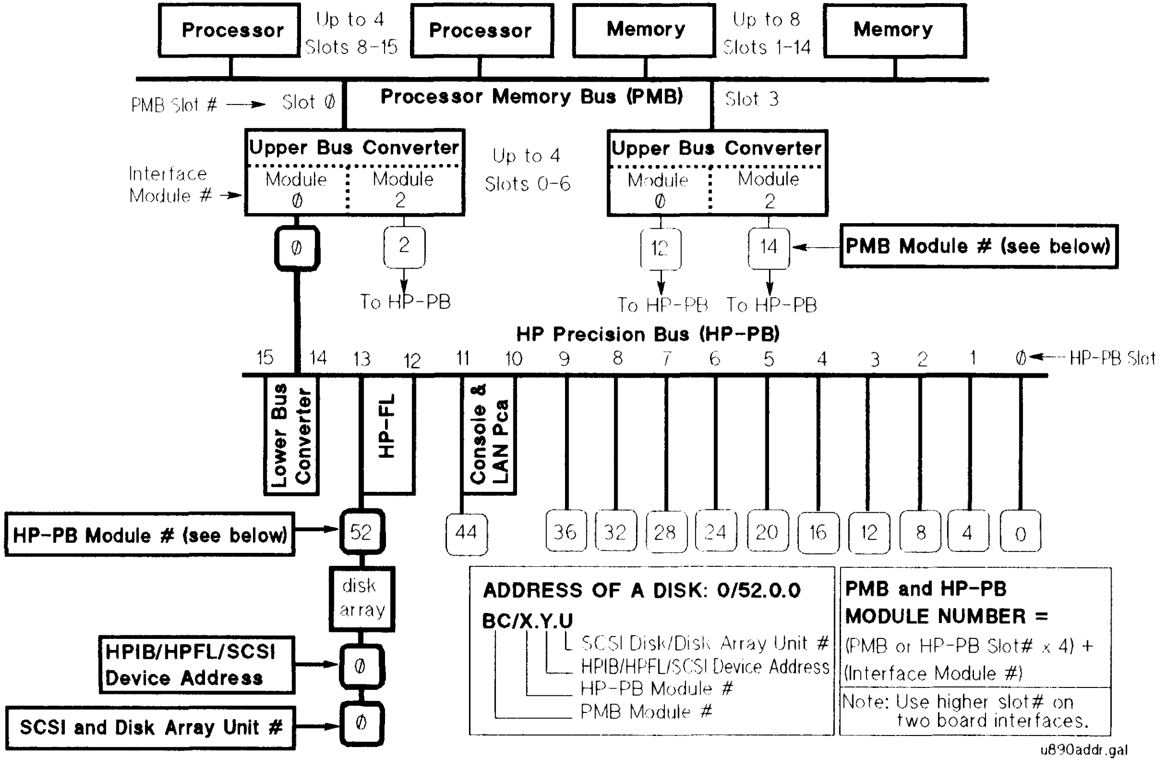


Figure 1-13. 890 Addressing

System Startup

HP-UX System Startup

ISL>hpux (BC/X.Y.Z;S)hp-ux

↓

Loads *hp-ux* into memory and then begins execution.

↓

Kernel invokes */etc/init* command.

↓

init spawns processes in the order that are listed in */etc/inittab*. All lines with action fields related to start-up and the default run-level are spawned.

↓

The system completes the startup process. Users may login.

/etc/inittab Example

```
init:2:initdefault:
ioin::sysinit:/etc/ioinit -i>/dev/console > /dev/console 2>&1
muxi::sysinit:/etc/dasetup </dev/console > /dev/console 2>&1
brc1::bootwait:/etc/bcheckrc </dev/console > /dev/console 2>&1
slib::bootwait:/etc/recoversl </dev/console > /dev/console 2>&1
brc2::bootwait:/etc/brc >/dev/console 2>&1
link::wait:/bin/sh -c "/rm -f dev/syscon; ln /dev/systty /dev/syscon"
>/dev/console 2>&1 [1]
cwrt::bootwait:cat /etc/copyright >/dev/syscon
rc::/etc/rc </dev/console >dev/console 2>&1
powf::powerwait:/etc/powerfail >dev/console 2>&1
vue :34:respawn:/etc/vuerc
cons::respawn:/etc/getty console
ttp1:2:respawn:/etc/getty -h tty0p1 9600
ttp2:2:respawn:/etc/getty -h tty0p2 9600
ttp3:2:respawn:/etc/getty -h tty0p3 9600
ttp4:2:respawn:/etc/getty -h tty0p4 9600
```

1 Prints as a single line on the console; no linewrap.

/etc/inittab Fields

inittab format: *id:run-level:action:process*

Where:

Key	Operation
id	Unique 1 to 4 character identification.
run-level	Defines when a process is allowed to run. If the system run-level matches a process run-level, the process will be spawned or allowed to continue running. If the system run-level does not match the process run-level, the process will be killed. Valid run-levels are *0* through *6*, *s*, or *S*. Processes can have multiple run-levels, for example, *id:234:respawn* ... initiates the process to run-levels 2, 3, or 4.
action	The options listed below: *respawn*—if process does not exist, start; if process dies, restart. *wait*—start the process waiting to complete. *once*—start once, do not wait for completion; if it dies, do not restart. *boot*—start once at boot-up; do not wait for completion; do not restart. *bootwait*—start once at boot-up; wait for completion; do not restart. *powerwait*—execute only when *init* receives a power fail signal (SIGPWR). *off*—if process associated with this entry is running, send a 20 second warning signal, then kill the process. *initdefault*—invoke first time *init* is run. Sets default run level. *sysinit*—execute before *init* tries to access console. Used to initialize devices that *init* could receive run-level information from.
process	This is the command to be executed.

/etc/inittab Start-up Process Flow

- Sets the system run-level as indicated by *initdefault*.

  init:2:initdefault:

- Runs */etc/ioinit* command. Assigns output and error messages to the console. *ioinit* initializes Kernel I/O System data structures using information from the */etc/ioconfig* file. The *-i* option causes *ioinit* to run */etc/insf* in order to assign logical unit numbers (lu) and create special device files for all the new devices on the system.

  ioin::sysinit:/etc/ioinit -i >/dev/console 2>&1

- Turns on all multiplexer cards. Assigns input from console, output to console, and directs errors to console.

  muxi::sysinit:/etc/dasetup < /dev/console >/dev/console 2>&1

- Run */etc/bcheckrc* shell program. Assign input, output, and error messages to the console. *bcheckrc* exits if running on a diskless client, starts mirror disks if configured, and invokes *fsck* on dirty file systems.

  brc1::bootwait:/etc/bcheckrc </dev/console >/dev/console 2>&1

- Run */etc/recoversl* shell program. Assign input, output, and error to the console). *recoversl* checks for the existence of shared libraries that are critical to the system. If any critical shared library is missing or damaged, *recoversl* assists the system administrator in recovering the shared library from update media.

  slib::bootwait:/etc/recoversl </dev/console >/dev/console 2>&1

- Run */etc/brc* shell program. Assigns output and error messages to */dev/syscon*. */dev/syscon* is used by HP-UX as a virtual system console. *brc* removes the file */etc/mnttab* that contains old file system mount information. If not running on a diskless client, *brc* removes */etc/rcflag*.

  brc2::bootwait:/etc/brc >/dev/console 2>&1

- Remove old */dev/syscon* device file. Create new */dev/syscon* file and link it to */dev/systty* (the physical console). Direct output and errors to the console.

  *link::wait:/bin/sh -c "/bin/rm -f /dev/syscon; *
  ln /dev/systty /dev/syscon" >/dev/console 2>&1

- Display a copyright message on the console for legal purposes.

```
cwrt::bootwait:cat /etc/copyright >/dev/syscon
```

- Run */etc/rc* shell program. Direct output and errors to the console. *rc* contains run commands that set date and time, mount file systems, and perform other housekeeping chores.

```
rc::wait:/etc/rc </dev/console >/dev/console 2>&1
```

- Run */etc/powerfail* shell program. Assign output and error messages to the console.

```
powf::powerwait:/etc/powerfail >/dev/console 2>&1 \
#power fail routine
```

- Create a *getty* process for the system console. Set up communication protocol and issue the first login prompt.

```
cons::respawn:/etc/getty console console
```

- Run HP *vue* if the system is in run-state 3 or 4. Only X11 window terminals that have been properly configured for *vue* will be enabled. Processes that are not configured to run in run-state 3 or 4 will be killed.

```
vue :34:respawn:/etc/vuerc
```

- Create a *getty* process for each terminal that is to be enabled. Set up communication protocol and issue the first login prompt.

```
ttp1:2:respawn:/etc/getty -h tty0p1 9600
ttp2:2:respawn:/etc/getty -h tty0p2 9600
ttp3:2:respawn:/etc/getty -h tty0p3 9600
ttp4:2:respawn:/etc/getty -h tty0p4 9600
```

Changing Run Levels with *init*

- The */etc/init* command can be used to change the system's run-level.
- *init* reads */etc/inittab* only when the run-level changes, or when forced to by a *Q* or *q* option.
- */etc/init* command and parameters, for example:

  \# *init [0123456SsQq]*

Where:

Key	Operation
S | *s*	Single user run-level. (Only the virtual console, */dev/syscon*, is enabled.).
2	By convention, used as a multi-user run-level. Normally used to enable user terminals and other serial devices.
013456	Assigned by system administrator to create other run-levels.
Q | *q*	Forces re-examination of */etc/inittab* even if setting same run-level.

Note

The default run-level, double colons (::), matches run-levels 0-6.

Example: *co::respawn:/etc/getty console console* (console will remain enabled in run-levels 0-6)

The */etc/bcheckrc* Script

- The *bcheckrc* script performs the following tasks:
  - *bcheckrc* determines the type of system invoking this script. If the calling system is a diskless cluster, the script is exited.
  - If applicable, mirror disks are configured and *fsck* is run on them.
  - Check if the mountable file systems listed in */etc/checklist* were cleanly unmounted on last shutdown. If a file system was not cleanly unmounted, *fsck* is run in *preen* mode to check for corruption.
  - If corruption is detected by *fsck* and can be repaired without data loss, the file system is automatically corrected. If a correction could result in data loss, the operator is prompted to run *fsck* interactively to repair the damage; then, to reboot the system using the */etc/reboot -n* command.

The */etc/brc* Script

- The *brc* script performs the following tasks:
  - The */etc/rcflag* flag file is removed if present. This flag is used by the */etc/rc* script to test if system start-up is occurring. If this flag file is absent */etc/rc* assumes that the system is in the start-up process.
  - *brc* removes the */etc/mnttab* file unless the system being started is a diskless client. The */etc/mnttab* file contains the file system mounting information; */etc/mnttab* is updated whenever the */etc/rc* script is run at system start-up.

The */etc/rc* Script

- The *rc* script is divided into two sections: Functions and Function Calls.
  - Functions—an example:

```
initialize ( )
{
  RBOOTD_DEVICES="" # device file used by /etc/rbootd
  if [ "$SYSTEM_NAME"= "" ] # system's network name
  then
    SYSTEM_NAME=unknown
    export SYSTEM_NAME
  fi
  TIMEOUT=20 # timeout length for date setting
  vtgateway="" # name of system acting as the gateway
  vtgopts="" # vtdaemon options
  vtginterfaces="" # gateway devices
}
localrc ( )
{
  # This function is intended for adding local initialization
  functions to rc.
  : # do nothing instruction (a function must contain some
  command)
}
```

  - Function Call—an example:

```
initialize
:
localrc
```

/etc/rc Function Calls by System Type

/etc/rc **Function Calls by System Type**

Function Call	System Type			Basic Tasks Performed
	SA[1]	DS[2]	DC[3]	
hfsmount	√	√	√	Mounts high performance file systems
setparms	√	√	√	Set system configuration values
initialize	√	√	√	Set time zone, system name, etc.
switch_over	√	√	√	Starts SwitchOver/UX
set_state	√	√	√	Determine if standalone, server, or diskless client
set_privgrp	√	√	√	Associate a kernel capability with a group id
set_date	√	√		Set system date, time, and year
save_core	√	√	√	Saves memory core dump to file system files
swap_start	√	√	√	Turn ON paging and swapping
syncer_start	√	√		Start Synchronizer; flush file system memory to disk every 30 seconds
lp_start	√	√		Start lp scheduler if configured
clean_ex	√	√		Save editor files open during power failure
clean_uucp	√	√		Scan spool directories; delete old *uucp* files
net_start	√	√	√	Start networking
csp_start		√	√	Start the cluster server
rbootd_start		√	√	Start remote boot daemon for diskless cluster

1 SA = Standalone System.

2 DS = Diskless Server.

3 DC = Diskless Client.

/etc/rc Function Calls by System Type, cont.

Function Call	System Type			Basic Tasks Performed
	SA[1]	DS[2]	DC[3]	
cron_start	√	√	√	Start cron and make new *cronlog*
pty_start	√	√	√	Start pseudo-terminal daemon
vt_start	√	√	√	Start vtdaemon
list_tmps	√	√	√	Display files in */tmp*, */usr/tmp*, and */lost+found*
clean_adm	√	√	√	Save oldlogs; make *sulog*, *diaglog*, and *messages*
diag_start	√	√	√	Start diagnostic event logger
syslogd_start	√	√	√	Start system error message logger; make *syslog*
envd_start	√	√	√	Start environmental daemon
audit_start	√	√		Start audit subsystem
localrc	√	√	√	Run Sys. Admin. created initialization commands

1 SA = Standalone System.

2 DS = Diskless Server.

3 DC = Diskless Client.

The */etc/powerfail* Script

- The *powerfail* script performs the following tasks:
  - Runs */etc/src.sh* shell program which sets system configuration variables such as timezone (TZ) and the system's network name (*SYSTEM_NAME*).
  - Starts multiplexer cards using */etc/dasetup*.
  - Logs the fact that a power failure occurred on system console and user terminals.
  - If required, the system administrator can add commands to reload any programmable I/O card or device needing post failure attention.

System Shutdown

The *shutdown* and *reboot* Commands

- *shutdown* is the recommended command for halting and rebooting the system. The */etc/shutdown* script performs the following:
  - Changes to the root directory (/).
  - Warns users of impending shutdown and waits a specified delay time (default wait is 60 seconds).
  - Stops non-essential system and user processes.
  - Changes the run-level to single-user (30 second delay).
  - Unmounts file systems other than *root*.
  - Executes *sync* to flush the system cache buffers to disk.
  - If *-r* or *-h* was used, reboots or halts the system.

shutdown Syntax

\# */etc/shutdown [-r | -h] [grace]*

Key	Operation
-r	Automatic reboot following shutdown.
-h	Halts the system following shutdown.
grace	Optional number of seconds to wait before killing processes.

reboot Syntax

\# */etc/reboot [-h | -r] [-n] [-s]*

Key	Operation
-h	Halts the system.
-r	Reboots the system automatically (default).
-n	No *sync* before halt or reboot.
-s	*sync* before halt or reboot (default).

Backup/Restore

Recovering from a Catastrophic Data Loss

To recover, perform the following:

1. Install HP-UX.
2. Perform applicable updates.
3. Create file systems on non-system disk drives.
4. Restore applicable incremental backup(s).
5. Restore last complete backup.

tar Tape File Archiver

- Saves and restores files on magnetic tape of flexible disk.
- When a directory is to be saved, *tar* recursively searches down the tree to save all related subdirectories and files.
- Files saved with full path names will be restored to the same locations. Directories are created as required.
- Files saved with relative path names (./path) can be restored on any directory.

Caution

- Default option is *N* (see following pages for full explanation).
- Due to internal limitations in the header structure, not all file names of fewer than 256 characters fit when using the *N* key. If a file name does not fit, *tar* prints an error message and does not archive the file.
- Symbolic link names are limited to 100 characters when using the *N* key.
- Tape errors are handled ungracefully.

tar Tape File Archiver Command Syntax

The *tar* command and common options are shown below:

 # tar key [arg ...] [file | -C directory]

Where:

Key	Operation
c	Creates an archive (save).
x	Extracts from an existing archive (restore).
v	Verbosely shows *tar* operations and displays each file accessed.
f fsname	Use device specified in *fsname* for save or restore (default special device file: */dev/rmt/0m*).
H	Cause all entries in hidden directories (context-dependent files) to be written in archive.
m	Restore files from archive with current system date and time stamp.
N	Default. Write a new (POSIX) format archive. New format allows file names up to 256 characters; archives and restores special files and symbolic links.
O	Write in old (pre-POSIX) format.
u	Update existing file in archive (very slow).
t	List all filenames in archive (with *v* option adds file mode and owner).
r	Append file(s) to end of existing archive.
-C directory	*tar* will change to *directory* and continue save or restore.
w	Wait for user confirmation of *tar* operations.
h	Treats symbolic links as normal files or directories.

tar Examples

- Create an archive of all files associated with your home directory (and subdirectories) to the default tape drive verbosely:

  $ cd
  *$ tar cv **

- Create an archive of */full/path/file1* and */full/path/name/file2* to a high-density tape drive verbosely:

  # tar cvf /dev/rmt/0h /full/path/file1 /full/path/name/file2

- Create an archive of all files on */directory* using relative pathnames so it can be restored on any directory:

  # *tar cv ./directory*

- Create an archive of all files on two unrelated directories to a high-density tape drive:

  # *tar cvf /dev/rmt/0h /full/path/directory1 /full/path/directory2*

- List all archived files on the default tape drive to the printer:

  # *tar tv | lp*

- Restore the entire archive from the default tape drive verbosely:

  # *tar xv*

- Restore a single file from a high density tape drive:

  $ *tar xvf /dev/rmt/0h /full/path/name/file1*

- Restore all files saved by relative pathing to */my/directory*:

  # *cd /my/directory*
  # *tar xv*

- Restore a file saved by relative pathing to */my/directory*:

  # *cd /my/directory*
  # *tar xv ./subpath/file1*

Note

tar will not accept HP-UX wild card characters in pathnames when restoring files on a system.

---

cpio Copy File Archives In and Out

cpio copies files in or out of an archive. An archive may be a file or a *raw* physical device.

cpio -o	Reads STDIN to obtain a list of path names and copies those files to STDOUT along with path names and status information. Normally, redirection is used to make an archive file or tape. For example: *# ls | cpio -o > /dev/rmt/0m*
cpio -i [pattern]	Extracts from STDIN (an archive file) those filenames that match zero or more patterns. The files are recreated based on the paths placed in the archive by *cpio -o*. For example: *# cpio -i chapter[1-5]* < /dev/rmt/0m*
cpio -p directory	Reads STDIN to obtain a list of path names and copies those files to a previously created target directory. For example: *# ls | cpio -p archivedir*

cpio Command Syntax

The *cpio* command and common options are shown below:

\# *cpio -o [Bcxvh]*
\# *cpio -i [Bdctuxv] [patterns]*
\# *cpio -p [duxv] directory*

Where:

Key	**Operation**
c	Write header information in ASCII for portability.
d	Create directories as needed.
t	Print only table of contents from the input.
x	Save or restore special device files. *mknod* recreates these files on restore (superuser only). Restoring special device files onto a different system can be dangerous; use only on HP-UX machines. Backups made for one system should not be restored on another.
v	Verbose; causes list of file names to be printed.
patterns	Only files that match *patterns* according to the rules of Pattern Matching Notation are selected. The default for *patterns* is *.
h	Follow symbolic links as if they were normal files or directories. Normally *cpio* archives the link.
B	I/O is blocked to 5120 bytes per record (meaningful only when data is directed to devices which support variable length data such as magnetic tape).
u	Copy unconditionally (older file will replace newer file).
directory	Destination directory tree.

cpio Examples

- Save all files in current directory in blocks with ASCII headers onto medium speed magnetic tape:

  # ls | cpio -ocBx > /dev/rmt/0m

- Use *find* command to save all files on your working directory recursively with ASCII headers onto high-speed magnetic tape:

  # find . -print | cpio -ocBx > /dev/rmt/0h

- Use *find* command to save all files on your working directory recursively with ASCII headers onto an archive directory:

  # find . -print | cpio -pdxv /archive_directory

- Print a listing of an archive to the printer:

  # cpio -icBtxv < /dev/rmt/0m | lp

- Restore */path/myfile* from tape:

  # cpio -icBxv /path/myfile < /dev/rmt/0m

tcio Tape Cartridge Formatter

tcio reads or writes data to a character Command Set 80 Cartridge Tape Unit special device file specified in *filename*.

tcio -o	Reads STDIN and writes data to the Command Set 80 Tape Unit.
tcio -i	Reads Command Set 80 Tape Unit and writes data to STDOUT.

tcio Tape Cartridge Formatter Command Syntax

The *tcio* command and common options are shown below:

\# *tcio -o[drvV] [-S buffersize] devicefile*
\# *tcio -i[dvr] [-S buffersize] devicefile*
\# *tcio -u[rV] devicefile*

Where:

Key	**Operation**
d	Prints checksum to STDERR (normally terminal). User may record this number for comparison of checksum value on restore. Checksum is not recorded on tape.
r	Release the tape from the mechanism, unlocking the door.
v	Verbose mode; prints information and errors to STDERR.
S buffersize	Forces memory of *buffersize* to be reserved for reading or writing the tape. Size of the buffer is 1014 times *buffersize* (range: 32-512). If *buffersize* is not specified, *tcio* defaults to the maximum size available.
devicefile	Special device file of tape unit.
V	Power OFF tape verification.
u	Perform utility function. Used with *r* to unlock tape cartridge door.

tcio Examples

- Save all files in current directory in blocks with ASCII headers onto an integrated disk and cartridge tape on single controller:

  # ls | cpio -ocBx | tcio -o /dev/rct/c1d1s2

- Use *find* command to save all files on your working directory with ASCII headers onto cartridge tape:

  # find . -print | cpio -ocBx | tcio -o /dev/rct/c1d0s2

- Print a listing of an archive to the printer:

  # tcio -iv /dev/rct/c1d1s2 | cpio -ict | lp

- Restore */path/myfile* from cartridge tape:

  # tcio -iv /dev/rct/c1d0s2 | cpio -icBxv /path/myfile

- To unlock tape cartridge door:

  # tcio -urV /dev/rct/c1d0s2

/etc/fbackup Backups

- The HP Recommended System Backup Utility
   - Syntax: *# fbackup -f devicefile [-0-9] [options]*

Common Options

Key	Operation
-f devicefile	Devicefile identifies the backup device.
[-0-9]	Backup level (0 = full backup; 1-9 indicates an incremental backup).
-i / e	Include/exclude file tree from backup (cannot be used with *-u*).
-I f_name	Create index of backup files in *f_name*.
-g f_name	Use *f_name* as graph file (must be used with *-u* option).
-u	Update */usr/adm/fbackupfiles/dates* (cannot be used with *-i* | *-e*).
-v	Verbose.
-H	Search hidden subdirectories (context-dependent files).

Note

- The user of *fbackup* need not be superuser. If the user does not have access to a given file, the file is NOT backed up.
- *fbackup* does not backup network special files.
- *fbackup* may not work correctly with NFS mounted file systems.

/etc/fbackup Backup Examples

- To backup all mounted files (assumes superuser capability):

  # fbackup -f /dev/rmt/0h -0vHi /

- To backup all files residing on */extra* tree except */extra/users*:

  # fbackup -f /dev/rmt/0h -0vHi /extra -e /extra/users

- To backup */users* file tree to cartridge tape:

  # fbackup -0vHi /users -f - | tcio -ov /dev/rct/c1d0s2

/etc/frecover Restores

- Works in Conjunction with *fbackup* Archives

o Syntax:

> \# *frecover -R path [-f devicefile]*
> \# *frecover -r [ovX] [-f devicefile]*
> \# *frecover -x [ovX] [-g f_name] [-i path] [-e path] [-f devicefile]*
> \# *frecover -I path [-f devicefile] [-c config]*
> \# *frecover -V path [-vy] [-f devicefile] [-c config]*

Common Options

Key	**Operation**
-r	Recover all files on backup tape.
-f devicefile	Devicefile identifies the backup device (default is */dev/rmt/0h*).
-x	Files identified by *-i*, *-e*, or *-g* are extracted/not extracted.
-g f_name	Use *f_name* as graph file.
-i / e	Include/exclude file tree from backup.
-I path	Create index of backup tape and place it in the *path*.
-o	Recover file regardless of age.
-v	Verbose.
-X	Recover files relative to the current working directory.
-V path	The volume header from the backup is written to the *path*.
-R path	Restart interrupted *frecover*.
-c config	Use configuration file for error actions.

Note

- The user of *frecover* need not be superuser. If the user does not have access to a given file, the file is not restored.
- *frecover* does not work with network special files.

Recovering Files with *frecover*

- To recover all files from an fbackup archive tape (assumes superuser capability):

  # frecover [-r] -o -f /dev/rmt/0h

  The *-r* option is the default value. The *-o* option causes *frecover* to restore a file in the archive that is older than one existing within the file system.

- To restore all files archived on */extra* tree except */extra/users*:

  # frecover -x -i /extra -e /extra/users -f /dev/rmt/0h

- To restore the */users* file tree from cartridge tape:

  # tcio -iv /dev/rct/c1d0s2 | frecover -x -vi /users -f -

dd Device-to-Device Copy

- *dd* is an all purpose utility that copies the specified input file to the specified output file. Using *dd* options, data conversions are possible.
- Often used to save or restore information disk-to-disk on hard disk partitions.

Caution

- □ In the past you may have used the *dd* command to recreate a boot area that has been corrupted (by copying the file */usr/lib/uxbootlf* to the beginning of the raw device file of a disk). DO NOT DO THIS ANY LONGER! The LIF area at the beginning of an LVM volume is located in a slightly different area than for a traditional disk. You can destroy all of the LVM information on a disk if you do this. Use the */etc/mkboot* command to create a boot area on an LVM disk. For details on making an LVM disk a bootable disk, see *Managing Logical Volumes* in the *System Administration Manual*, HP part number B3108-90005.
- □ Never use *dd* to copy a section of a non-LVM disk directly to an LVM physical volume without going through the LVM mechanism. That is, if a disk represented by the device file */dev/[r]dsk/c1d0s2* is an LVM disk (a physical volume), DO NOT use */dev/[r]dsk/c1d0s2* as the output file (or input file) in the *dd* command. If you do, you will overwrite the LVM data structures at the beginning of your LVM disk or overwrite a non-LVM disk with LVM disk structures. If you no longer want to use the destination physical volume as an LVM disk, you must properly remove it from the volume group using the appropriate LVM commands.

  DO NOT:

  dd if=non-LVM-disk-section-device-file of=LVM-physical-volume-device-file

You can, however, use a logical volume as a destination.

YOU CAN:

dd if=non-LVM-disk-section-device-file of=logical-volume-name-device file

- □ Just as when working with traditional disk sections, the destination must be as big as the source, and you must take care not to overwrite something critical!

- The following is an example of the *dd* command and syntax to copy a file system from one drive to another:

  # dd if=/dev/rdsk/c1d0s2 of=/dev/rdsk/c2d0s2 bs=1024k

- The following is an example of the *dd* command and syntax to copy a non-LVM disk section to a logical volume. Note that the size of the destination must be as large as the source.

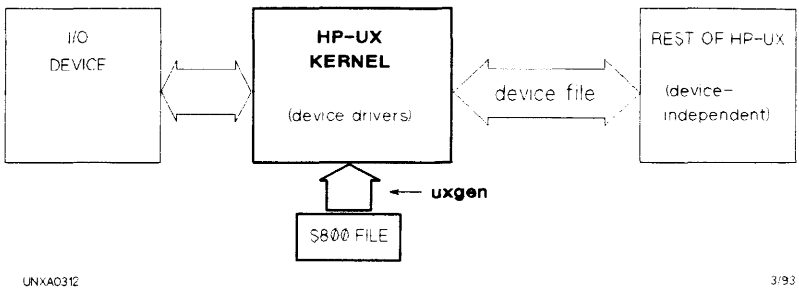
  # dd if=/dev/rdsk/c5d0s5 of=/dev/vg01/rlvol2 bs=1024k

Where:

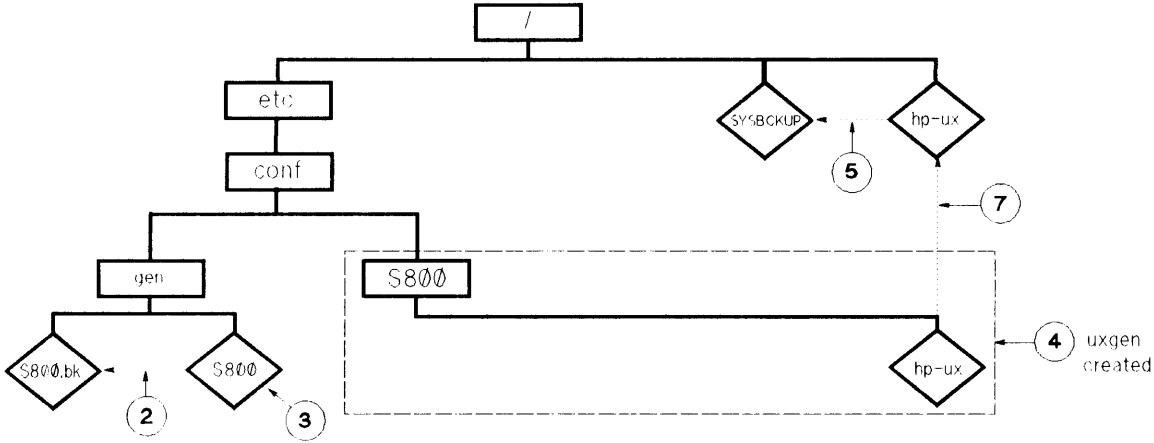
Key	**Operation**
if=	Specifies the input file name (default STDIN).
of=	Specifies the output file name (default STDOUT).
bs=	Specifies the I/O block size.

UXGEN

Adding Kernel Drivers



- A new kernel (hp-ux) must be generated using *uxgen* only when adding a device whose associated driver was not previously included in the existing Kernel with uxgen (1m).



The following steps assume that the gen file is named S800:

1. # cd /etc/conf/gen
2. # cp S8ØØ S8ØØ.bk
3. # vi S8ØØ
4. # /etc/uxgen S8ØØ
5. # cp /hp-ux /SYSBCKUP
6. # cd /etc/conf/S8ØØ
7. # mv hp-ux /hp-ux
8. # shutdown -h Ø
9. Turn the system off. Install hardware.
10. Turn the power on. Reboot the system.
11. # cp /hp-ux /SYSBCKUP
12. If boot string changed, modify LIF's 'AUTO' file

S800—UXGEN Input File

The S800 file consists of the following sections:

- Include statements.
  - □ # include */etc/master*.
    - Contains connectivity rules for I/O configuration.
    - Defines major numbers for all devices.
    - Contains a list of kernel tunable parameters and defaults.
  - □ Used to add device drivers and optional subsystems to the Kernel (hp-ux).
- Kernel devices.
  - □ Specifies the location of console, root, primary swap, and dump(s) devices.
- Tunable parameters.
  - □ Used to specify non-default kernel parameters.
- I/O statement.
  - □ Used to configure devices whose drivers do not support autoconfiguration (for example: *instr0*, *gpio0/gpio1*, *psi0*, *pdn0*, *rti0/rti1*).

S800—What Drivers are Kernel Resident?

- To list kernel resident drivers, type the following:

  # *lsdev*

S800—Adding Drivers and Subsystems

- Determine the correct driver(s) to use for the peripheral or subsystem by consulting the list of drivers later in this section; *Installing Peripherals*; or product installation manuals for the desired subsystem (LVM, X.25, etc.).
- Edit the */etc/conf/gen/S800* file adding the appropriate driver(s) or subsystem include statement(s) to the end of the existing include statements in the S800 file.
- Example:

  To add an HP 2563B line printer driver to the kernel, add the following line at the end of the *include* statements in the S800 file:

  ```
  include lpr0;
  ```

  Once this has been accomplished, generate a new kernel using the *UXGEN Process*.

S800—Kernel Devices

- The HP-UX kernel must know the location of the root file system, primary swap device, dumps device (if present), and the console.
- Failure to specify the correct location of these necessary devices may result in a hung boot, or worse, damage to a file system should the kernel be told to swap over the top of a data disk.
- The following are examples of typical kernel devices specifications:
  - HP-PB Model 815:

```
console on mux1;
root on hpib1.disc1 at 16.0 section 13;
swap on hpib1.disc1 at 16.0 section 15;
dumps on default;
```

  - HP-PB Model 827:

```
console on mux2 at 56;
root on lvol;
swap on lvol;
dumps on default;
```

  - HP-PB Model 890:

```
console on bus_converter.lanmux0.mux4 at 0.44.0
root on lvol;
swap on lvol;
dumps on default;
```

Kernel Device—Console

- Syntax:

  console on <driver_path> at <hdw_address>;

  or

  console on default;

- Examples:
  - Default console path

    ```
    console on default;
    ```

  - CIO system console on machines without System Main Bus (SMB) 825/835/625/845/645:

    ```
    console on cio_ca0.mux0 at 8.0;
    ```

  - CIO system console on machines with System Main Bus (SMB) 850/855/860/865/870:

    ```
    console on bus_converter.cio_ca0.mux0 at 2.4.1;
    ```

  - HP-PB system console on 808/815/822/832/842/852:

    ```
    console on mux1 at 16;
    ```

  - HP-PB system console on 8x7:

    ```
    console on mux2 at 56;
    ```

  - HP-PB non-LAN system console 890:

    ```
    console on bus_converter.mux4 at 0.44;
    ```

  - HP-PB LAN system console 890:

    ```
    console on bus_converter.lanmux0.mux4 at 0.44.0;
    ```

Kernel Device—Root

- Syntax:
  - Non-LVM Root Device

    root on default [section <integer>];

    or

    root on <driver_path> at <hdw_address> [section <integer>];

  - LVM Root Device

    root on lvol;

    or

    root on default [section <integer>];

- Non-LVM Root Device Examples:
  - Use default section as root device:

    ```
    root on default;
    ```

  - HP-PB root device on 808/815/822/832/842/852:

    ```
    root on hpib1.disc1 at 16.0 section 13;
    ```

  - CIO root device on machines without System Main Bus (SMB) 825/835/625/845/645:

    ```
    root on cio_ca0.hpib1.disc1 at 8.0.0 section 13;
    ```

  - CIO root device on machines with System Main Bus (SMB) 850/855/860/865/870:

    ```
    root on bus_converter.cio_ca0.hpib1.disc1 at
    2.4.0.0 section 13;
    ```

- LVM Root Device Example:
  - Use root section as defined in the Boot Data Reserve Area (BDRA) and LIF's LABEL file:

    ```
    root on lvol;
    ```

Kernel Device—Swap

- Syntax:
  - Non-LVM Swap Device

    swap on default [section <integer>];

    or

    swap on <driver_path> at <hdw_address> [section <integer>]

    ⋮

    <driver_path> at <hdw_address> [section <integer>];
  - LVM Swap Device

    swap on lvol;

    or

    swap on default [section <integer>];
- Non-LVM Swap Device Examples:
  - Use Boot Disk section 15 as primary swap device:

    ```
    swap on default;
    ```
  - HP-PB primary swap on 808/815/822/832/842/852:

    ```
    swap on hpib1.disc1 at 16.0 section 15;
    ```
  - CIO swap devices on machines with System Main Bus (SMB) 850/855/860/865/870:

    ```
    swap on bus_converter.cio_ca0.hpib0.disc1 at
            2.4.0.0 section 15
            bus_converter.cio_ca0.hpib0.disc1 at
            6.8.0.5 section 11;
    ```
- LVM Swap Device Example:
  - Use default as primary swap device as defined in Boot Data Reserve Area (BDRA) and LIF's LABEL file:

    ```
    swap on lvol;
    ```

Kernel Device—Dumps

- Syntax:
  - Non-LVM Dumps Device(s)

    dumps on default [section <integer>];

    or

    dumps on <driver_path> at <hdw_address> [section <integer>]

    ⋮

    <driver_path> at <hdw_address> [section <integer>];

  - LVM Dumps Device(s)

    dumps on default [section <integer>];

    or

    dumps on lvol;

- Non-LVM Dumps Device Examples:
  - Use default section (primary swap) as dumps device:

    ```
    dumps on default;
    ```

  - HP-PB dumps device on 808/815/822/832/842/852:

    ```
    dumps on hpib1.disc1 at 16.2 section 15;
    ```

  - CIO dumps device on machines without System Main Bus (SMB) 825/835/625/845/645:

    ```
    dumps on cio_ca0.hpib0.disc1 at 8.0.2 section 0;
    ```

  - CIO multiple dumps devices on machines with SMB Main Bus (SMB) 850/855/860/865/870:

    ```
    dumps on bus_converter.cio_ca0.hpib0.disc1 at
             2.8.0.1 section 15
             bus_converter.cio_ca0.hpib0.disc1 at
             6.8.0.5 section 15;
    ```

- LVM Dumps Device Examples:
  - Use default (primary swap) as dumps device:

    ```
    dumps on default;
    ```

  - Use one or more LVM dumps devices (information resides in the Boot Data Reserve Area (BDRA) and LIF's LABEL file):

    ```
    dumps on lvol;
    ```

I/O Statement Syntax

- The I/O statement in the *S800* file must begin with *io* { and end with }. See examples below.
- Curly braces pairs must enclose driver I/O statements unless the driver is the only or final driver in a driver path. See examples below.
- Semicolons must be used to end I/O statements at the end of a driver chain. See examples below.
- Comments begin with /* and end with */. See examples below.

Non-Automatically Configurable Devices

Device controlled by drivers that do not automatically configure (for example: *instr0*, *gpio0/gpio1*, *psi0*, *pdn0*, *rti0/rti1*) MUST be specified in the I/O statement.

Some examples:

- To add an HP-IB instrument or HP-IB plotter on a CIO system at physical path 4.2.7:

```
io {

        cio_ca0 address 4 {
           hpib0 address 2 {
                  instr0 address 7;
           } /* ends hpib0 driver statement */
        } /* ends cio_ca0 driver statement */

} /* ends io statement */
```

- To add a General Purpose I/O card on a HP-PB system at physical path 48 (slot 12):

```
io {
        gpio1 address 48;
} /* ends io statement */
```

CIO (Mid-bus) Architecture Drivers

- System Main Bus Driver

Name	Supported Devices
bus_converter	Bus Converter Interface (required on 850/855/860/865/870 only)

- Mid-Bus Driver

Name	Supported Devices
cio_ca0	CIO Channel Adapter Interface

- CIO Cardcage Bus Master Device Drivers

Name	Supported Devices
hpib0	HP-IB Interface (27110 card). Supports up to 8 HP-IB peripherals
hpfl0	HP-IB Interface (27111 card). Supports up to 8 HP-FL peripherals
scsi2	CIO SCSI (27147 card). Supports up to 7 SCSI

- CIO Cardcage Device Adapter Drivers

Name	**Supported Devices**
disc1	CS/80 and SS/80 devices (disks and cartridge tapes) connected via HP-IB interface (27110 card)
disc2	CS/80 devices (disks) connected via HP-FL interface (27111 card)
disc3	Disk devices connected via SCSI interface
tape1	1/2 inch Magnetic tape drives and HP C1511 HP-IB DDS- Format Tape Drive
tape2	SCSI tape drives such as the HP C1512A SCSI DDS-Format Tape Drive
lpr0	256X line printers (using CIPER protocol)
lpr1	2932/2934 and 2235 line printers (using Amigo protocol)
lpr3	C2753A Model F100 SCSI printer protocol
mux0	RS-232 (serial) peripherals (27140 card); includes terminals, printers, and plotters (8 ports available)
mux0_16	Same as above with 16 ports
instr0	All other HP-IB peripherals (plotters, other printers, general HP-IB instruments)
lan0	Networking (27125 card)
gpio0	General-purpose parallel I/O interface (27114 card)

- CIO Cardcage Device Adapter Drivers, Cont.

Name	Supported Devices
osi0	OSI Express Card interface (32124A or 32125A cards) connected via HP A1126 bus converter supporting HP MAP 3.0 networking
autoch	Optical Library Systems
autox0	650 Mbyte magneto (rewritable) optical disk

HP-PB Architecture Drivers

- HP-PB Bus-Master Drivers

Name	Supported Devices
cio_ca0	HP-FL disks that are connected to the HP A1749A HP-PB Fiber-Optic Link (PBA-FL) Precision Bus Device Adapter
hpib1	HP-IB interface (28650 card). Supports up to 8 HP-IB peripherals
hpfl0	HP-FL Disks that are connected to the HP A1749A HP-PB Fiber-Optic Link (PBA-FL) Precision Bus Device Adapter
hpfl1	HP-FL disks that are connected to an HP 28615A card
scsi1	SCSI port on the 28655A HP-PB SCSI/Parallel Adapter card

- HP-PB Device Adapter Drivers

Name	Supported Devices
disc1	CS/80 and SS/80 devices (disks and cartridge tapes) connected via HP-IB interface (28650 card)
disk2	HP-FL disks connected to HP A1749A HP-PB Fiber-Optic Link (PBA-FL) Precision Bus Device Adapter
disc3	Disks such as HP 7957S connected via SCSI interface
disk4	HP-FL disks that are connected to the HP 28615A Fiber Optic Link Adapter
scc1	RS-232C for Serial Communications Controller (SCC) on motherboard or CPU of HP 9000 Models 808/815 only
tape1	1/2 inch Magnetic tape drives and HP C1511 HP-IB DDS- Format Tape Drive
tape2	SCSI tape drives such as the HP C1512A SCSI DDS-Format Tape Drive
autoch	Optical Library Systems
autox0	650 Mbyte magneto (rewritable) optical disk
lpr0	256X line printers (using CIPER protocol)
mux2	RS-232C (serial) peripherals (40299 card); includes terminals, printers, and plotters
mux4	RS-232C (serial) peripherals connected to the built in ports on HP 9000 Model 890 computer
instr0	All other HP-IB peripherals (plotters, other printers, general HP-IB instruments)

- HP-PB Device Adapter Drivers

Name	**Supported Devices**
lan1	Networking designed for HP 28562-60001 LANLINK card
gpio1	General-purpose parallel I/O interface (28651 card)
osi0	OSI Express Card interface (32122 or 32123 cards) supporting HP MAP 3.0 networking

lsdev

- Lists I/O drivers contained in the kernel (hp-ux). For each I/O driver, the character and block major numbers, and the device class are displayed.
- Example output:

Character	**Block**	**Driver**	**Class**
-1	-1	processor	processor
-1	-1	memory	memory
50	-1	lan0	lan
5	5	tape1	tape_drive
1	-1	mux0	tty
7	8	disc1	disk
-1	-1	cio_ca0	cio
-1	-1	hpib0	hpib
60	-1	nm	pseudo
56	-1	ni	pseudo

ioscan Syntax

- Probes and displays the I/O configuration of a running system.
- Syntax:

  ioscan [-k|-u] [-d dvr | -C class] [-l lu] [-H hdw_path] [-f [-n]] [devfile]

  or—

  ioscan [-M ...] [-H hdw_path]

- Options:

Key	Operation
-k	Displays kernel data structures; does not scan devices.
-u	Displays usable devices; does not scan devices.
-d	Selects devices associated with an I/O driver.
-C	Selects devices in a device class.
-l	Selects devices with a given logical unit number.
-H	Selects devices at a given hardware path.
-f	Full listing; includes software paths, status, etc.
-n	Lists device files associated with each device.
-M	Driver, or string of drivers separated by periods, that specify hardware device(s).
hwd_path	Physical address of device (for example *BC/X.Y.Z.U*).
none	Probes the devices and displays information for all the devices in the system.

ioscan Default Behavior

\# *ioscan*

Hardware Path	Description	Status
0	processor	ok
4	cio	ok
4.0	hpfl	ok
4.0.0	disk	ok
4.1	tty	ok
4.2	hpib	ok
4.2.0	disk	ok
4.2.4	tape_drive	ok
4.4	lan	ok
4.6	hpib	ok
8	memory	ok
12	memory	ok

ioscan Listing Device Files

\# *ioscan -fn -C tape_drive*

Class	LU	H/W Path	Driver	H/W Status	S/W Status
tape_drive	1	4.2.4	cio_ca0.hpib0.tape1	ok(0x178)	ok
			/dev/diag/mt/1	/dev/mt/1m	/dev/rmt/1l
			/dev/mt/1h	/dev/mt/1mn	/dev/rmt/1ln
			/dev/mt/1hn	/dev/rmt/1h	/dev/rmt/1lm
			/dev/mt/1l	/dev/rmt/1hc	/dev/rmt/1mn
			/dev/mt/1ln	/dev/rmt/1hn	

\# *ioscan -fn -H 4.1*

Class	LU	H/W Path	Driver	H/W Status	S/W Status
tty	0	4.1	cio_ca0.mux0	ok(0x7)	ok
			/dev/diag/mux0	/dev/tty0p1	/dev/tty0p4
			/dev/mux0	/dev/tty0p2	/dev/tty0p5
			/dev/tty0p0	/dev/tty0p3	

ioscan Full Listing

\# *ioscan -f*

Class	LU	H/W Path	Driver	H/W Status	S/W Status
processor	-	0	processor	ok(0x0)	ok
cio	-	4	cio_ca0	ok(0x1000)	ok
hpfl	-	4.0	cio_ca0.hpfl0	ok(0x8)	ok
disk	0	4.0.0	cio_ca0.hpfl0.disc2	ok(0x2)	ok
tty	0	4.1	cio_ca0.mux0	ok(0x7)	ok
hpib	-	4.2	cio_ca0.hpib0	ok(0x2)	ok
disk	0	4.2.0	cio_ca0.hpib0.disc1	ok(0x22b)	ok
tape_drive	1	4.2.4	cio_ca0.hpib0.tape1	ok(0x178)	ok
lan	0	4.4	cio_ca0.lan0	ok(0x6)	ok
memory	-	8	memory	ok(0x800)	ok

\# *ioscan -fu*

Class	LU	H/W Path	Driver	H/W Status	S/W Status
disk	0	4.0.0	cio_ca0.hpfl0.disc2	ok(0x2)	ok
tty	0	4.1	cio_ca0.mux0	ok(0x7)	ok
tape_drive	0	4.2.3	cio_ca0.hpib0.tape1	ok(0x178)	ok
tape_drive	1	4.2.4	cio_ca0.hpib0.tape1	ok(0x178)	ok
lan	0	4.4	cio_ca0.lan0	ok(0x6)	ok

Device Files

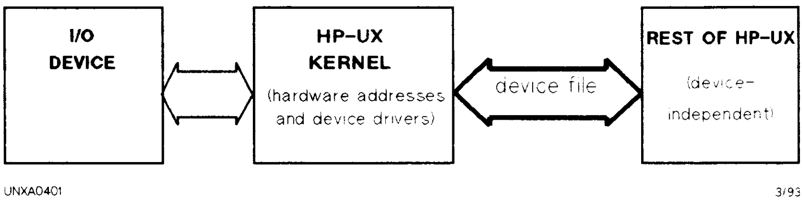


Figure 1-14. Device Files

- Special device files link the kernel to the rest of HP-UX.
- Special device files pass device dependent parameters (for example: driver name, tape density, modern configuration, etc.) to the kernel.
- Transfer data by character (raw), or in blocks (cooked).
- Must reside in the */dev* directory.
- Created by *insf*, *mksf*, or *mknod* commands after system generation.

Block and Character Devices

Block Devices

- I/O operations are done in blocks buffered by the kernel's file system cache.
- Disks with MOUNTED file systems use block I/O. Standard block size transfer size is 8192 bytes. Block sizes of 8Kb and 4Kb are supported.

Character Devices

- I/O operations on character devices are done in character data streams using minimal buffering by the kernel.
- All devices support character I/O.
- UNMOUNTED disk file systems, use character data streams for I/O.

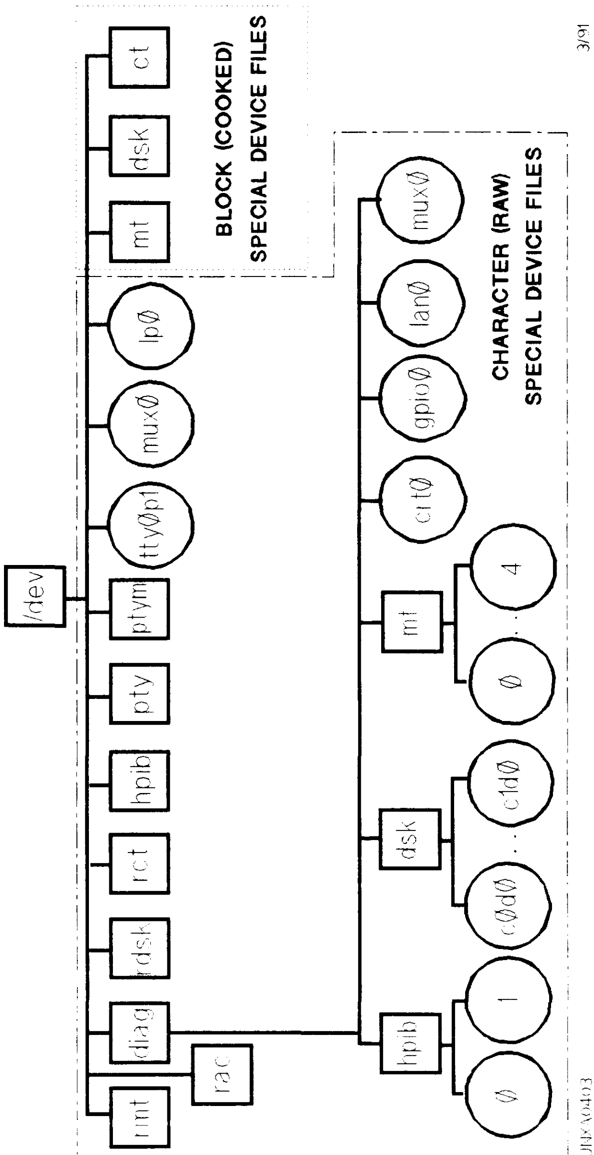


Figure 1-15. */dev* **Directory: Peripheral Device Files**

Device File Naming Conventions

Terminal Device Files

/dev/tty0p0 Where:

Key	Operation
tty0	0 is a logical unit number (lu) assigned by *insf*.
p0	0 is the multiplexer port number (0-5).

MODEM Device Files

Dial-in MODEM Device File Names.

/dev/ttyd2p3 Where:

Key	Operation
d	Designates a dial-in modem.
2	A logical unit number (lu) assigned by *insf*.
3	The multiplexer port number (0-5).

UUCP MODEM for Automatic Dial-Out.

/dev/cua2p3 Where:

Key	Operation
2	A logical unit number (lu) assigned by *insf*.
3	The multiplexer port number (0-5).

UUCP MODEM for Manual Dial-Out.

/dev/cul2p3 Where:

Key	Operation
2	A logical unit number (lu) assigned by *insf*.
3	The multiplexer port number (0-5).

Magnetic Tape Device Files

/dev/[r]mt/udn Where:

Key	Operation
[r]	When present, indicates that file can be found on the character (raw) directory for this device.
u	The logical unit (lu) assigned by *insf*.
d	Identifies tape density: *h* (high density, 6250 bpi), *m* (medium density, 1600 bpi), or *l* (low density, 800 bpi).
n	When present, indicates no rewind after write.

Examples:

/dev/rmt/0h	Logical unit 0, high density, character device file.
/dev/rmt/1hn	Logical unit 1, high density, character device file, no rewind after write.

Disk Device Files

Conventional High-Performance File System (HFS) Device Files.

/dev/[r]dsk/cXd0sZ Where:

Key	Operation
r	When present, indicates that file can be found on the character (raw) directory for this device.
X	Logical unit number (lu) *X* assigned by *insf*.
Z	Section number *Z* addressed by this device file.

Examples:

/dev/rdsk/c0d0s4	Logical unit 0, section 4, character device file.
/dev/dsk/c1d0s2	Logical unit 0, section 4, block device file.

Logical Volume Manager (LVM) File System Device Files.

/dev/vgXX/[r]lvolNN Where:

Key	**Operation**
XX	Integer volume group number to which the logical volume belongs.
r	When present, specifies use of the character device file associated with the logical volume specified. If absent, the block device file will be used.
NN	Integer logical volume number.

Examples:

/dev/vg00/lvol1	Volume group 00, logical volume 1, block device file.
/dev/vg01/rlvol3	Volume group 01, logical volume 3, character device file.

Printer Device Files

/dev/lpX Where:

Key	Operation
X	Logical unit number (lu) *X* assigned by *insf*.

Cartridge Tape Device File Names

/dev/[r]ct/cXdYsZ Where:

Key	Operation
r	When present, indicates that file can be found on the character (raw) directory for this device.
X	Logical unit number (lu) *X* assigned by *insf*.
Y	Unit number *Y* (0 = does not share same controller as disk; 1 = shares same controller as disk drive, for example: 7914CT).
Z	Section number *Z* addressed by this device file (typically *2*; section 2 accesses all sections of tape).

Device Files Needed by HP-UX

Filename	**Use**	**Recreated by:**
syscon	Access system console	}
systty	Access system console	} *insf -d cn*
console	Access system console	}
diag0/	HP-UX diagnostics	*insf -d diag0*
dmem	HP-UX diagnostics	*insf -d dmem*
ktest	HP-UX diagnostics	*insf -d ktest*
kmem	Virtual memory	}
mem	Physical memory	} *insf -d mm*
null	Bit bucket	}
config	Access I/O configuration	*insf -d devconfig*
root	Access root device during	*insf -d root*
rroot	system start-up	
tty	Access user terminal	*insf -d sy*

Note

Remember to change directories to */dev* before executing the *insf* command.

/etc/lssf

- Lists characteristics of special files:
  - Driver name
  - lu number
  - Driver options
  - HP-UX software address
  - Device file path

Note *lssf* will NOT list Logical Volume Manager (LVM) device files. Use *ll* instead.

Examples:

# *lssf /dev/dsk/c7d0s3*	disk2 lu 7 unit 0 section 3 address 4.0.0 /dev/dsk/c7d0s3.
# *lssf /dev/tty1p5*	mux0 lu 1 port 5 hardwired address 4.5 tty1p5.

Commands to Make Device Files

MKNOD Example

This example makes a dial-in modem special device file for mux lu 2, port 0.

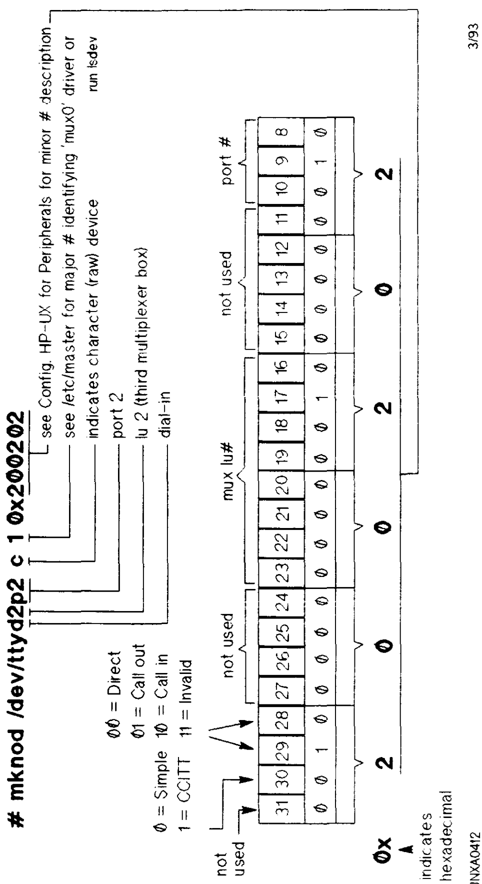


Figure 1-16. MKNOD Example

mksf

- Creates one or more special device files for the specified driver.
- Extracts the major number from the kernel (hp-ux).
- Command syntax varies depending on the driver type (mux0, disc0, etc.).

  Syntax example for a mux0 driver:

  # mksf -d mux0 [-l lu] [-p port] [-h | -i | -o] [-c] [path]

Key	**Operation**
-d mux0	Specifies driver to use for special device file(s) created.
-l lu	Logical unit number (lu) of a device as assigned by *insf*.
path	Default path name is *tty*<lu>*p*<port>.
-c	CCITT (European Standard).
-h	Hardwired (direct connect).
-i	Dial-in modem.
-o	Dial-out modem.
-p	Multiplexer port number (0-5).

- This example makes a dial-in modem special device file for mux lu 2, port 2.

  # cd /dev
  # mksf -d mux0 -l 2 -p 2 -i ttyd2p2

Caution

The *insf* command should only be run in single user mode. To accomplish this, type the following from the system console:

 # cd / ; shutdown 0

Note

insf builds device files in the working directory, therefore, *cd* to the target directory before invoking the command. The target directory is often */dev*.

- Reads information about devices directly from the kernel (hp-ux).
- Assigns a logical unit number to each *new* device.
- Creates all default special files for each supported device.
- Sets appropriate file permissions.
- If needed, set appropriate file ownership.
- Creates diagnostic special files.
- Uses standard path naming conventions.

Syntax

 # /etc/insf [-d driver | -C class] [-H hdw_path] [-l lu] [-f] [-k] [-e]

- No options: Assigns logical unit (lu) numbers and creates default device files for all *new* devices in the kernel (hp-ux).
- Options:

Key	Operation
-d	Select device associated with an I/O driver (*disc0*, etc.).
-C	Select devices in a device class (disk, printer, etc.).
-H	Select device at a hardware address (8.0.0, etc.).
-l	Select device with a particular lu or assign a specific lu.
-f	Force a specific lu to map to a device.
-k	Assign an lu to the device without creating device files (cannot be used with the *-e* option).
-e	Create device files for devices having lu numbers (cannot be used with the *-k* option).

- Examples:

  # cd /dev ; insf
  Assigns logical unit (LU) numbers to new devices and creates default device files for those devices.

  # cd /dev; insf -e
  Rebuilds device files for existing (as opposed to new) devices.

  # cd /dev ; insf -H BC.X.Y.Z.U -l lu
  Assigns logical unit number *lu* to the device indicated by path *BC.X.Y.Z.U*. This will fail if the logical unit number is currently assigned to another device. Use *rmsf -k* or *rmsf -H* to remove an LU number.

rmsf

- Removes device files and logical unit (LU) number(s) assigned to hardware path(s) in the kernel.
- Syntax:

  \# *rmsf [-a | -k] devfile*
  \# *rmsf [-k] [-d driver |-C class] -H hdw_path*

- Options:

Key	Operation
-a	Removes device information in kernel and the device file(s).
-k	Removes device information in the kernel only.
-d	Selects devices controlled by an I/O driver.
-C	Selects devices in a device class.
-H	Selects device at a hardware address.

- Examples:

  \# *rmsf tty2p0*
  Removes the device file. Device information in kernel is retained.
  \# *rmsf -H BC.X.Y.Z.U*
  Removes device information in the kernel and device file(s) associated with path *BC.X.Y.Z.U*

Line Printer Spooler System

User Capabilities

- Queue files to printers.
- Obtain status of lp system.
- Cancel any print job.
- Mark printers in and out of service.

LP Administrator Functions

- Change configuration of system.
- Mark printers in and out of service.
- Start and stop the system.

Spooling System Directory Overview

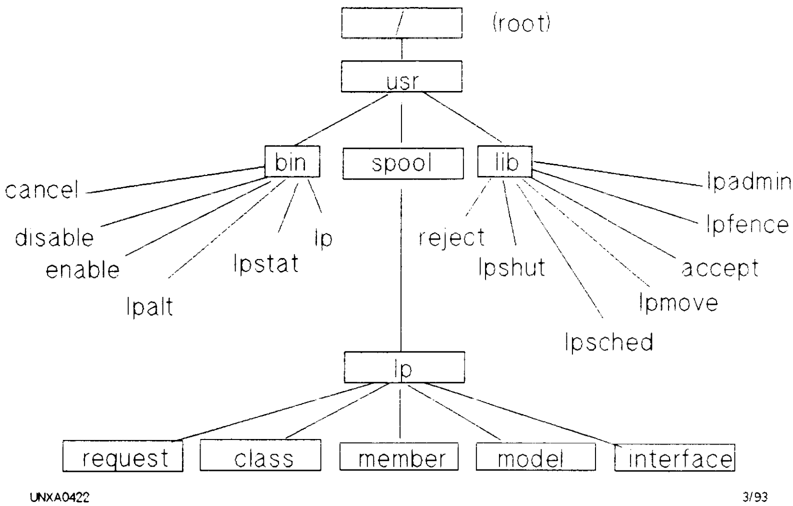


Figure 1-17. Spooling System Directory Overview

Spooler System Terminology

Term	**Meaning**
Class	- A grouping of printers. - Must have at least one printer. - Usually contains more than one printer. - Printers may be assigned to more than one class.
Device	- Port on system where printer is connected. - Accessed through the device file */dev/lp*(n), or other device file on */dev*.
Printer	- Logical name of a physical device. - Name used for actual printer.
Destination	- Place where the files will be queued. - Destination can be a queue for a particular printer, or class of printers.
Scheduler	- Runs when system is multi-user. - Routes requests on a FIFO or priority basis. - Enables files to be printed on a specific printer or printer class.
Log	- Log file located in */usr/spool/lp/log*. - Maintains a record of each LP spooler system request, including: request ID, user name, printer name, time, error messages, and reprints due to failure.

User Commands

Common LP Spooler User Commands

Command	Example
lp	$ *lp /etc/passwd*
	$ *lp -dprinter_name file1* (select printer by name)
	$ *lp -p4 /etc/passwd* (set priority to 4)
lpstat	$ *lpstat -t*
cancel	$ *cancel printer_name-117* (cancel job in queue)
	$ *cancel printer_name* (cancel job now printing)
enable	$ *enable printer_name*
disable	$ *disable -r*"Changing Print Ribbon" *printer_name*

/usr/bin/lp

- The *lp* command arranges for the named files and associated information (collectively called a request) to be printed by a printer or plotter.
- The *lp* command and common options are shown below:

  $ *lp [-ddest] [-m] [-nnumber] [-ppriority] [-s] [-ttitle] [-w] [files]*

Key	Operation
-ddest	Files will be printed on *dest* (the name of a printer or printer class). If this key is omitted, file(s) will be printed on the default system printer.
-m	Send mail after the file(s) have been printed.
nnumber	Print *number* of copies of the specified file(s).
ppriority	Give *priority* to the print request (specified as a integer from 0 through 7 for the lowest through the highest priority). The file(s) will be printed if equal to or greater than the printer fence value; if below the fence, the printout will be deferred.
-s	Suppress messages from *lp* such as *request ID is . . . .*
-ttitle	Print *title* on the banner page of the output.
-w	Write a message on the user's terminal after *files* have been printed.
files	File(s) to be printed or plotted (default file is STDIN).

/usr/bin/lpstat

- The *lpstat* command prints information about the status of the LP spooling system. It reports the status of the scheduler, printers, printer classes, and the default system printer.
- If *lpstat* is used with no arguments, it reports the status of any requests made by the user.
- The *lpstat* command and common options are shown below:

  $ *lpstat [-c[list]] [-d] [-p[list]] [-r] [-t] [-v[list]]*

Key	**Operation**
-c[list]	Print class names and their member, where *list* is a listing of intermixed printer names and class names.
-d	Print the system default printer destination.
-p[list]	Print the status of printers, where *list* is a listing of printer names.
-r	Print the status of the scheduler.
-t	Print all status information.
-v[list]	Print the names of printers and the path names of the devices associated with them, where *list* is a listing of printer names.

/usr/bin/cancel

- The *cancel* command cancels printing of the file currently being printed, or spooled files when request ID numbers are specified.
- The *cancel* command and common options are shown below:

  $ *cancel [ids] [printers] [-a] [-e] [-i] [-uuser]*

Key	**Operation**
ids	Specify the request IDs to be canceled in *ids*.
printers	Specify the name(s) of printer(s) or printer classes where the *cancel* request is be performed.
-a	Remove all requests a user owns as specified in the *printers* queue.
-e	Empty the spool queue of all requests for the *printers* specified. Only the superuser can use the *-e* option.
-i	Cancel only local requests.
-uuser	Remove any requests queued belonging to *user*. Multiple *-u* options are allowed; only the superuser can use this option.

/usr/bin/enable

- The *enable* command activates the named printer(s), enabling them to print requests taken by */usr/bin/lp*. This changes the status to allow the scheduler to send requests to the printer.
- The *enable* command is shown below:

  $ *enable <u>printers</u>*

Key	**Operation**
printers	Specify the name(s) of printer(s), or printer class or classes to be enabled.

/usr/bin/disable

- The *disable* command deactivates the named printer(s), disabling them from printing requests taken by */usr/bin/lp*. By default, any requests that were being printed on the designated printers are reprinted in entirety on the same printer, or on another printer in the same class.
- The *disable* command and options are shown below:

  $ *disable [-c] [-r[<u>reason</u>]] <u>printers</u>*

Key	**Operation**
-c	Cancel any requests that are currently printing on any of the designated *printers*.
-r[<u>reason</u>]	Associates *reason* with the deactivation of *printers*. The specified *reason* will be printed when status is requested via the */usr/bin/lpstat* command. *reason* must be enclosed in double quotes if the *reason* string contains white space.
<u>printers</u>	Name of printer(s) being deactivated.

Administrator Commands

/usr/lib/lpshut

- The *lpshut* command shuts down the printer scheduler.
- The printers that are printing when *lpshut* is invoked will stop printing. Any files that are interrupted will reprint in entirety after the scheduler is started again by the */usr/lib/lpsched* command.
- All LP commands perform their functions even when the scheduler is not running. Jobs can still be submitted to queues.
- The *lpshut* command has no arguments and is shown below:

  # *lpshut*

lpadmin

Configuration changes can be done with *lpadmin*:

- Add and remove printers.
- Change class members.
- Change the device associated with the printer.
- Assign an interface for a printer.
- Assign a system default destination.

Options to lpadmin		Examples
-d(dest)	Assign dest as the system default destination.	-dlaser
-x(dest)	Remove destination dest from the spooler.	-xlaser
-p(printer)	Selects a printer to which other options refer.	-plaser
-acluster_client	Specify a non-rootserver cnode printer.	

To be used when the -p(printer) option is selected:		
-g(priority)	Sets the default priority (0 - 7) for incoming print requests for printer; default is 0.	-g4
-c(class)	Insert printer as a member of a class.	-clp3
-v(device)	Associate device with printer.	-v/dev/lp0
-r(class)	Remove printer from a class.	-rlp3

Declaring an Interface Program		
-e(printer)	Use existing printer interface.	-elp
-i(interface)	Use a new script as interface.	-i(path/mk)
-m(model)	See */usr/spool/lp/model*.	-mhp2563a

/usr/lib/accept

- The *accept* command allows */usr/bin/lp* to accept request for the named printer or class of printers.
- The *accept* command is shown below:

  $ *accept destinations*

Key	Operation
destinations	Name of a spooled printer(s), printer class or classes that */usr/bin/lp* is to accept requests from.

/usr/lib/lpsched

- The *lpsched* command schedules request from */usr/bin/lp* for printing on printers. *lpsched* is typically invoked in the */etc/rc* script at system start-up.
- The *lpsched* command and options are shown below:

  \# *lpsched [-v] [-a]*

Key	Operation
-v	Write a verbose record of the *lpsched* process on */usr/spool/lp/log*.
-a	Write *lpana* (see *lpana*(1M)) logging data on */usr/spool/lp/lpana.log*.

lpmove

- The *lpmove* command moves requests from one printer to another. For example:
  - ☐ Move a specific request to another destination.

    \# */usr/lib/lpshut*
    \# *lpmove dp-115 printer_name*
    \# */usr/lib/lpsched*

  - ☐ Move all destination requests to another destination.

    \# */usr/lib/lpshut*
    \# */usr/lib/reject -r*"Down for Repair" *printer_name*
    \# */usr/lib/lpmove from_printer_name to_printer_name*
    \# */usr/lib/lpsched*

/usr/lib/lpmove

- The *lpmove* command moves requests that were queued by */usr/bin/lp* between printer destinations.
- This command may be used only when the scheduler is not running, so */usr/lib/lpshut* must be invoked prior to using the *lpmove* command.
- *lpmove* can move a single request to another printer or class, or all requests from one printer or class to another printer or class.
- The *lpmove* command and arguments are shown below:

  \# *lpmove* *dest1* *dest2*

Key	Operation
dest1	Request ID(s) to move to *dest2*. If a printer name is specified, all requests queued for that printer are moved to *dest2*. Following this, all subsequent requests to *dest1* will be rejected.
dest2	Name of printer or class of printers where requests are to be moved.

/usr/lib/reject

- The *reject* command prevents */usr/bin/lp* from accepting requests for the named printer or class of printers.
- The *reject* command and options are shown below:

  $ *reject [-r[reason]] destinations*

Key	Operation
-rreason	Associates *reason* with preventing */usr/bin/lp* from accepting requests. If the stated reason contains white space it must be enclosed in double quotes (for example: "Changing Ribbons"). Maximum length of a reason message is 80 characters; default is "Reason Unknown."
destinations	Name of printer or class of printers from where requests are rejected.

/usr/lib/lpfence

- The *lpfence* command defines the minimum priority for which a spooled file needs to be printed.
  - Fence values must be between 0 (lowest fence) and 7 (highest fence). Spooled files with a priority equal to or higher than the fence will be printed; files with priorities lower than the fence will be deferred.
  - Each printer has its own fence setting and is initialized to 0 (lowest fence) when configured into the spooling system by the */usr/lib/lpadmin* command.
  - The *lpfence* command may be used only when the scheduler is deactivated (using the */usr/lib/lpshut* command).
- The *lpfence* command syntax is shown below:

  \# */usr/lib/lpfence printer fence*

Key	**Operation**
printer	The individual printer or class name where fence is set.
fence	Fence priority value (0, lowest priority through 7, highest priority).

/usr/bin/lpalt

- The *lpalt* command alters a printer request that was made by the *lp* command. New unique ID is returned to standard output.
- The *lpalt* command and common options are shown below:

  $ *lpalt id [-ddest] [-m] [-nnumber] [-ppriority] [-s] [-ttitle] [-w]*

Key	**Operation**
id	Request ID returned by *lp*. This request will be altered if not printed.
-ddest	File(s) will be printed on *dest* (the name of a printer or printer class). If this key is omitted, file(s) will be printed on the default system printer.
-m	Send mail after the file(s) have been printed.
-nnumber	Print *number* of copies of the specified file(s).
-ppriority	Give *priority* to the print request. *priority* must be an integer between 0 (lowest priority) and 7 (highest priority) inclusive. The file(s) will be printed if equal to or greater than the printer fence value; if below the fence, the printout will be deferred.
-s	Suppress messages from *lp* such as *request ID is* ....
-ttitle	Print *title* on the banner page of the output.
-w	Write a message on the user's terminal after file(s) have been printed.

2

HP-UX Installation/Updating

New Installation of HP-UX—Initial Steps

- Read any *Read Me First* documents and follow directions therein. This information supersedes the released manual set.
- If you have had previous experience installing HP-UX version 9.0, this section should remind you of the key steps and decisions that must be made. If you have not had previous experience with the installation process please see *Installing and Updating HP-UX*, HP part number B3108-90006 (for HP-UX version 9.0) for detailed instructions.
- HP-UX installation consists of the following general steps:
  1. Install the hardware in card locations and verify peripheral addresses. Physically connect any additional peripheral devices.
  2. Boot to the ISL> prompt from the Support Tools Media.
  3. Run SS_CONFIG. Set stable storage parameters as required.
  4. Boot from the Install Media.
  5. From the *ISL>* prompt do the following:
     a. Set *primpath* to the path of the system disk.

        ISL> *primpath BC/X.Y.Z.U*

     b. Set *altpath* to the path of the tape device.

        ISL> *altpath BC/X.Y.Z.U*

     c. Set *autoboot* ON.

        ISL> *autoboot on*

d. Type the installation string required for your media.

For HP-UX version 9.0:

ISL> *hpux install to (BC/X.Y.Z.U)*

Where: *BC/X.Y.Z.U* is the physical path to the system disk.

For HP-UX version 8.0:

Consult *Installing and Updating HP-UX,* HP part number 92453-90035 (for HP-UX version 8.0) for correct installation string.

6. The subsections that follow assume that you have begun the installation process and must answer a series of system specification questions. The questions are presented in the form of a checklist.
7. See *Post Installation Guidelines* (later in this section) for the general steps required to configure the newly installed system.

Installation—System Specification Decisions

The installation process will prompt you with a series of questions regarding System Size, the Root File System, Root Size, Swap Size, and Type of Filenames. The following will assist you in answering these questions.

Checklist Item	Information and Tasks
System size	Continuing the installation, you will see the following prompt: `Do you wish to do a full install (y/n) [y]?` The default is `Y`; so typing (Return) does a full installation. If you type N, you respond to prompts and get a minimal system for which you will need to customize the boot area after installation.
Root file system	You get the following prompt: `Do you wish to install a Logical Volume system (y/n) [y]?` Accepting the default, `Y`, installs to a logical volume. Typing N installs to a hard partition.
Root size	You get the following prompt: `Enter Root size in Mega bytes, range 104-592 [104]:` Specifying a larger size (for example, 180) should be adequate for the root file system). Entering a size greater than 200 will subsequently prevent the Logical Volume Manager from creating logical volumes for file systems such as `/usr`.

Swap size

You get the following prompt:

```
Enter Swap size in Mega bytes,
range 48-136 [48]:
```

The 48 MB default is usually adequate. A safer rule of thumb for small standalone systems is to allocate three times the size of physical memory. Size of swap can be determined by determining the largest application (look at the manual supplies or ask the manufacturer). If the customer will be running several applications concurrently, you should add their swap space requirements together. The maximum amount of swap space you can configure (both device swap and file system swap combined) is approximately 537 Mbytes).

Type of filenames

You get the following prompt:

```
Enter file system filename type,
long or short: [s]:
```

The default, `s`, specifies short filenames. Unless you are sure you will be using long filename, take the default. It is easy to convert to long filenames after the system has been installed at a later time using the *convertfs(1M)*. Type l to specify long filenames.

At this point, the system loads install kernel files and reboots the computer.

System Reboot and Initial Loading

Do NOT interrupt the boot process unless the system hangs. While the computer boots, remove the Install Media and install Product Media on the source drive.

Logical Volume Installation Continued

If you have chosen to install HP-UX 9.0 on Logical Volumes, continue with this section. If you selected hard disk partitioning see *Root Hard Partition Installation Continued* with follows.

The following screen shows the default root volume group creation.

```
------------------------------------------------------------
Root Volume Group Creation

Root Volume Group /dev/vg00 will
be created with the following
configuration:

Logical Volume    Size (MB) File System

/dev/vg00/lvol1   504        /
/dev/vg00/lvol2   64         swap

Root VG Disk(s)          Total Size (MB): 640
/dev/dsk/c0d0s6   640    Available (MB):   72

(Some instructions appear here ............)

  [Modify Root Volume Group]      [Restart]

  [OK]              [CANCEL]              [HELP]
------------------------------------------------------------
```

If the values on the screen are acceptable, activate OK. As shown below, the install program displays messages about what happens and indicates how to continue. If you need to use the options, the following sections explain them (Modify Root Volume Group and Restart).

```
-------------------------------------
UPDATE ON ROOT VOLUME GROUP

The Physical Volumes will be created
...
(Other messages appear)
...

Use OK to run Update, and continue

   OK      SHELL      HELP
-------------------------------------
```

Option: Restart

Activating this option aborts the installation and lets you start over.

Option: Modify the Root Volume Group

The screen looks like this.

```
--------------------------------------------------
           Change Root Volume Group

             [CHANGE FILESYSTEMS]
             [CHANGE DISKS]

             [CHANGE VG PARAMS]

   OK              Cancel                    Help
--------------------------------------------------
```

Checklist Item	Information and Tasks
Related Information	Selecting an option takes you to another screen. These screens are not explained; use Help to get context-sensitive help within them.
CHANGE FILESYSTEMS	Provides a screen that lets you modify the specifications for file systems (for example, you could add /usr and /mnt). You can also modify the swap space. After you make changes, activate OK to return to the screen for Root Volume Group Creation.
CHANGE DISKS	Provides a screen that lets you add or delete disks from the specification of disks for the root volume group, but you must retain at least one disk. It is better to install the root file system to one logical volume and, if desired, use SAM to add more disks after the installation. After you make changes, activate OK to return to the screen for Root Volume Group Creation.
CHANGE VG PARAMS	Provides a screen that lets you view the current values for logical volume parameters and, if desired, modify those values. You can create or remove logical volumes in the root volume group, but you must retain logical volumes for the root file system and swap space. After you make changes, activate OK to return to the screen for Root Volume Group Creation.

Root Hard Partition Installation Continued

This section is to be used if you decided to install the root file system to a hard partition on a physical disk. Normally, the default settings displayed on the *Root Disk Partitioning* screen are adequate. If swap size is inadequate, changes will be necessary.

Partition the Root Disk If Necessary

The screen shows the default configuration for your root disk (values for parameters such as Section, Size, Section Name, and Section layout for `boot`, directories such as `/usr`, and `swap`). Exactly what you see depends, for example, on your type of disk and its default values. In general, the screen looks like this:

```
---------------------------------------------------------------
                 Root Disk Partitioning

Filename Type (long or short)  |s|   Disk Type: |7935|

Section    Size            Section Names           Section
           (Mb)   (e.g. swap, /, ..., /extra)   Layout


        (Information ..............)


Softkey labels appear along this line ...
---------------------------------------------------------------
```

Changing the Hard Partitions

You should read this entire section. Then make modifications as necessary. The following checklist items discuss changes you can make:

Checklist Item	Information and Tasks
Long/Short filenames?	The default is short filenames. To have long filenames, tab to the field and type l over the s.
Partition size?	The values you see for disk partitioning are usually adequate. Do not change them unless you have sufficient expertise to know what to specify.
Swap space?	During an installation, you can install a minimal system or more than a minimal system. For following items provide guidelines:

- Some disks default to 16 MBs on Section 15 for swap space. This allows only a minimal system, which contains the KERN-BLD, TOOL, UX-CORE, CORE-SHLIBS, CORE-DIAG, C-MIN, EDITORS, and CMD-MIN filesets.
- The *Installing Peripherals* document has information about the default swap space for disks. Be aware that the HP 7914, 7936, 7958, 7959, 7962B, 7963B, and 9263B disks have 16 MBs of swap space.
- You must have at least 24 MBs to install more than the minimum system.
- You can install a minimal system now. Then, as a system administrator, you can modify the swap space and use *update*(1M) to add other filesets to the system.

Root Disk Partitioning Screen Operation

The section explains how to use the *Root Disk Partitioning* screen to change the names and section locations of the *mount points* for directories such as `/`, `/usr`, and `/bin` into which files are loaded during an installation.

Sections and Directories.

- HP-UX divides the physical space on the disk into logical sections for the purpose of managing the space efficiently and flexibly.
- Directories have certain sections by default. The *Root Disk Partitioning* screen lets you change the default assignments.
- The section to which a directory is assigned determines the size of that directory. You can change the size of a directory by moving it to another section. The root directory is typically assigned to Section 13, which covers the entire disk except for `boot` and `swap`. This assignment causes *update*(1M) to load all files into the root directory.

Things to Consider If You Change the Default Configuration.

- Looking at the screen, you see that some sections are formed by merging smaller sections, which means the same physical space is defined twice. (For example, section 11 encompasses sections 10, 3, 4 and 5 (or 10, 3 and 9). If you use any of these smaller sections, you cannot also use section 11 (and conversely) because the update program prevents you from using a space twice. You get a warning if you have not used all the sections available (for example, if you use 3 and 4, but not 5). Make sure all the space on the disk is allocated.
- Do not leave holes. If you use sections 0 and 14, or 0 and 13, you waste disk space (24 MBs on an HP 7935). You get a warning if your choice of sections leaves wasted space.
- The `swap` section is not a directory. It is an area of raw storage used by the HP-UX memory manager. You *must* assign at least one section to `swap`, and you *can* assign multiple sections to `swap`.

  If you install more than a minimal system at HP-UX Release 9.0, you *must* have at least 24 MBs of swap space. If you have a disk with a default swap section of 16 MBs and want to install more than a minimum system, assign a swap section of at least 24 MBs.

- Having multiple sections for file systems has advantages:
  - Logically related files are stored in contiguous physical space. This reduces disk access time.
  - You can tune block/fragment sizes according to the types of file in a particular file system. The *System Administration Tasks* document has information about this.
  - Making the root directory self-contained reduces risk of having activity on other disks corrupt the root directory.
  - You are better protected against root overflow. When the root directory is full, the operating system will not work correctly. If the root is in a separate section from such directories as `/tmp`, the root is unaffected if one of these other directories fills up its section.
- Having multiple sections for file systems has an disadvantage; namely, you might run out of space in a particular section when there is ample space elsewhere on the disk. Merging everything into one large section eliminates this problem but loses the advantages listed above.
- Creating *symbolic links* can help if you run out of space in a particular section. The *System Administration Tasks* document has information about creating symbolic links.

Using the Root Disk Partitioning Screen

The following steps explain how to change the default configuration.

1. Use (Tab) to move from line to line on the screen.
2. Space through a default name you want to change, move or delete. The `boot`, `/` (root) and `swap` entities must remain in the list. You can have more than one swap area.
3. Type in a new name where you want it to appear. There are some restrictions:

   a. Do not enter duplicate names. Doing so leads to the following error:

```
ERROR:  File system names in Section [number] and
        Section [number] cannot be identical.
```

   If you have a duplication, tab to the duplicate, delete it, reenter an appropriate entry, tab back to where you were, and continue.

   b. Do not use a section that is overlapped by another already in use.

   The line for a given section remains protected (you cannot tab to it) when there is an entry for an overlapping section. Clean out overlapping sections by spacing through them.

   c. Except for `swap`, enter section names that begin with `/`.

   Do not name sections `/dev`, `/bin`, `/lib`, or `/etc`. These directories must be under the / (root) section.

   If you do, press the space bar and make the indicated correction.

   d. Do not enter section names with embedded spaces.

   If you do, press the space bar and retype the name without spaces.

e. If you specified short filenames, do not enter section names longer than 14 characters.

   If you do, press the spacebar and type a shorter name; or enter `l` in the `Filename Type` field and use long filenames. Conversion from short to long file names can be accomplished with the *convertfs* command after installation.

When you have made desired changes and you think the values on the screen are appropriate, you need to continue. Continue by pressing `Perform Task`, which is (f1).

The Main Menu Continues the Installation

```
-----------------------------------------------------------------------

INSTALL                     Main Menu

Highlight an item and then press "Return" or
"Select Item".  To refresh the screen, press CTRL-L.

Source:  Tape Device          Destination: Local System
         /dev/scr_device                   /


       Select ALL Filesets on the Source Media ->
       Select Filesets for a Minimum System ->
       Select/View Partitions and Filesets ->
       Enter Codeword ->

       How to Use Install

 Help        Shell    Select                         Exit
                       Item                         Update
-----------------------------------------------------------------------
```

Read This Before You Choose a Main Menu Option

Checklist Item	Information and Tasks
You must eventually load filesets	Regardless of which option you choose for loading filesets, and regardless of which subscreen you use from the Main Menu, you must eventually activate `Start Loading` to continue the installation.
Do not interrupt the loading	While the system loads filesets, messages appear on the screen. They are also recorded in `/tmp/update.log` so you can review them after the installation. The loading takes one to several hours.
Install runs customization scripts	When the loading completes, the install program runs customization scripts for individual filesets and builds a new kernel. Again, you should not interrupt the system during this time.

Checklist Item	Information and Tasks
Install prompts for time zone, time, and date	The customize scripts prompt you to enter a time zone, time, and date.
Install prompts about networking	The customize scripts let you specify a hostname and an internet protocol address. If you get to this point in the installation and do not know your hostname and internet protocol address, you can continue and set up networking later. The *System Administration Tasks* manual explains how to do this.
End of an installation	Getting a login prompt indicates the installation is complete.

Is Your Media on a CD-ROM?

If you install from CD-ROM and you install software other than the 2-user Runtime product, activate the `Enter Codeword ->` option. You get the following screen.

```
-----------------------------------------------------------
      From CD-ROM (directory) to Local System

Modify the desired fields and press "Done".


Source Directory: /UPDATE_CDROM

Destination Directory: /

Codeword Certificate:

Codeword :____ ____ ____ ____
            ----short form-----

 Verified Hardware ID:
-----------------------------------------------------------
```

Checklist Item	Information and Tasks
Hardware IDs	You can get the current hardware IDs on your system by highlighting the `Verified Hardware ID` and pressing `Help`. You are not allowed to enter anything, you only get the information.
Codeword field	Enter the appropriate codeword from your CD-ROM Certificate and continue.

Option 1: Select All Filesets on the Source Media ->

Checklist Item	Information and Tasks
What the option does	Selecting `Select All Filesets on the Source Media ->` automatically loads every fileset on the source media.
Start the loading	From the `Select All...` screen, activate `Start Loading`.

Option 2: Select Filesets for a Minimum System->

Activating this option selects the following minimum set of filesets (plus the appropriate user license):

```
UX-CORE          CORE-DIAG
KERN-BLD         C-MIN
TOOL             EDITORS
CORE-SHLIBS      CMDS-MIN
```

Checklist Item	Information and Tasks
Related Information	The `Select Filesets for a Minimum System` screen appears. You can install a minimal system now and then, after the installation, you can use *update*(1M) to load other filesets.
Load the filesets	Activate `Start Loading` from this screen.

Options continue on the next page.

Option 3: View/Select Partitions and Filesets->

The following screen lets you tailor the functionality you load onto the system.

```
---------------------------------------------------------
        View or Select Individual Partitions


Mark "y" or "n" to make a selection.
Press "Main" to return to the partition selection
screen.

Select   Partition        Partition Description      Size Kb

n      DIAGNOSTICS      Hardware Diagnostics        37663
p      NETWORKING       Networking Products          8919
y      NLS              Native Language Support       472
y      OS-ADMIN         Rec. Administration Cmds     2292
y      OS-CORE          Recommended System Core      5517
y      OS-FEATURES      Selectable OS Features       8176
y      PROG-LANGUAGES   Programming Languages        8542
y      REFERENCE-DOC    Reference Manual Pages        348
y      SHARED-LIBS      Runtime Shared Libraries     2757
y      WINDOWS          Windowing Products            102

Help      Shell    Start     Disk   View      Global Main
                   Loading   Space  Filesets  Select
---------------------------------------------------------
```

Checklist Item	Information and Tasks
Using Y, N, and P	Y selects and N deselects a partition. If you select or deselect filesets in a partition, a P appears to indicate a partial selection.
Update checks dependencies	The install program does not let you load (or not load) partitions without accommodating required dependencies.
You eventually start loading	After you select or deselect the partitions or filesets, activate `Start Loading`.

Post Installation Guidelines

Essential tasks are:

- Setup system security (root password, trusted system, etc.).
- Configure peripherals (printers, terminals, disk drives, etc.) into the system.
- Mount file systems.
- Test the LP spooler.
- Print */etc/sbtab* (# *lp /etc/sbtab*).
- Check/configure software subsystems (networking, X11 Windows, uucp, etc.).
- Back up the configured system (*fbackup*).

Note

For further information, refer to the section *After Installing HP-UX* in the manual *Installing and Updating HP-UX*, HP part number, B3108-90006.

F, G, H, I Model Standard Hardware Configuration

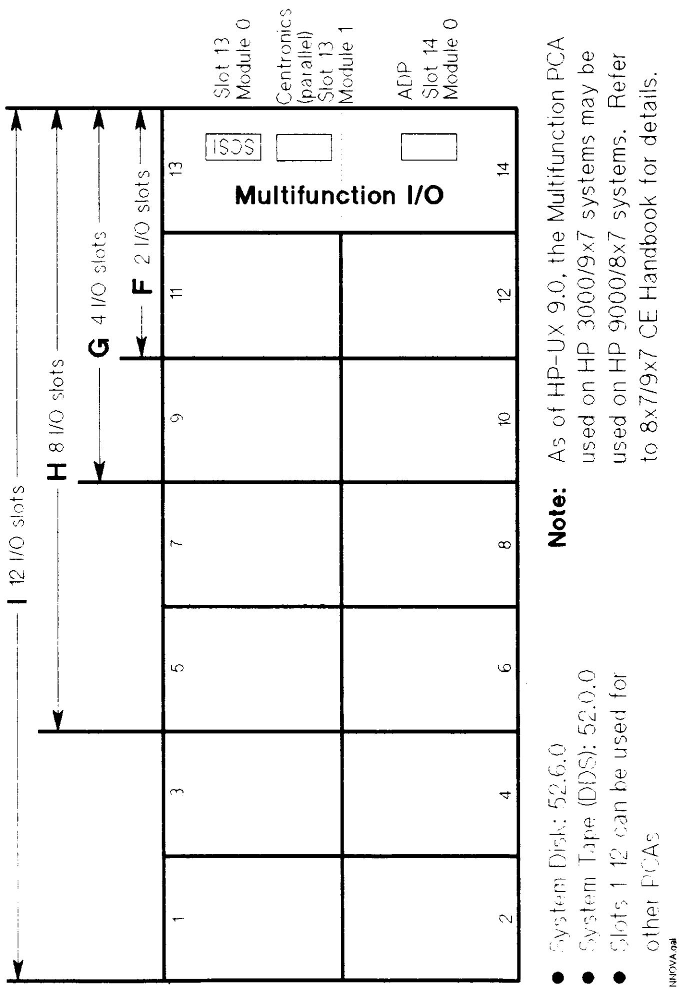
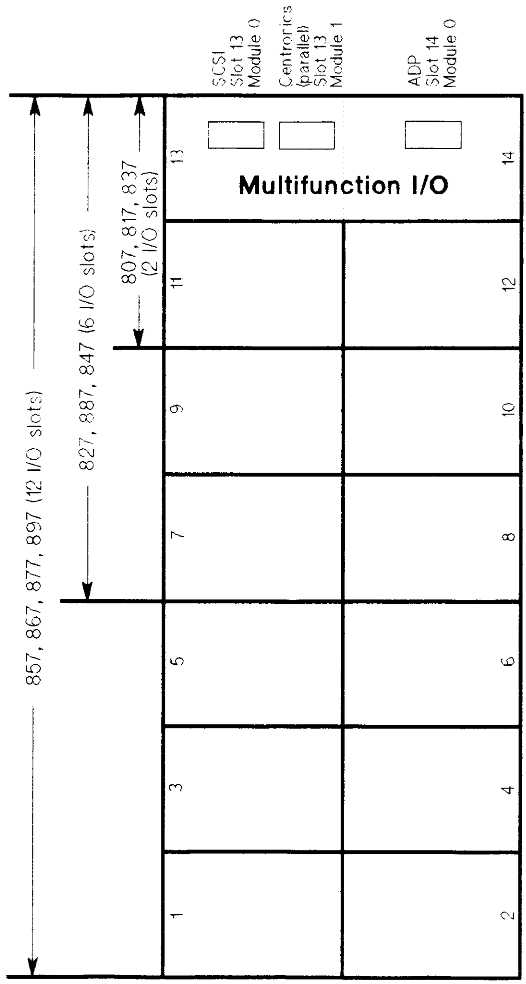


Figure 2-1. F, G, H, I Model Standard Hardware Configuration

8x7 Standard Hardware Configuration



- System Disk: 52.6.0
- System Tape (DDS): 52.0.0
- Slots 1-12 can be used for other PCAs

Note: As of HP-UX 9.0, the Multifunction I/O PCA used on HP 3000/9x7 systems may be used on HP 9000/8x7 systems. Refer to 8x7/9x7 CE Handbook for port details.

ONOVA.gal

Figure 2-2. 8x7 Standard Hardware Configuration

HP 9000 Model 808/815

815 Standard Hardware Configuration

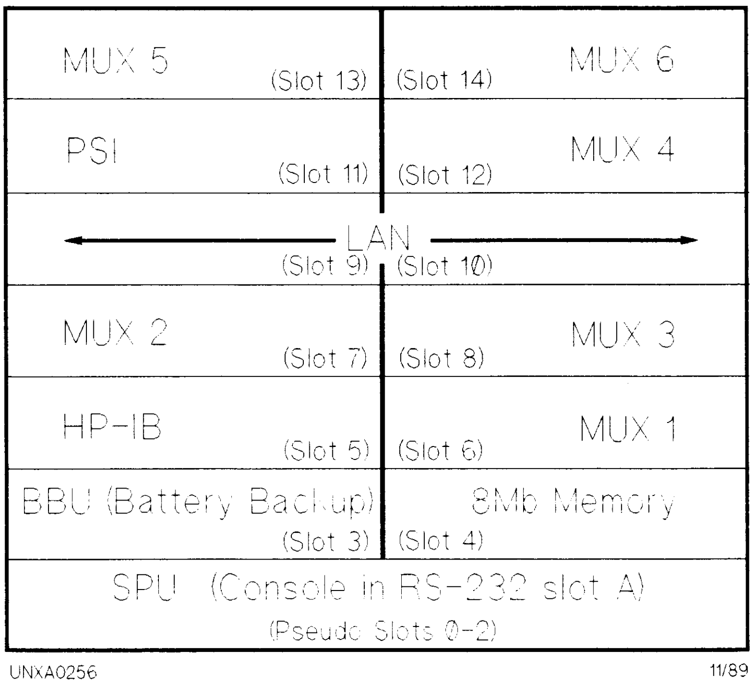


Figure 2-3. 808/815 Standard Hardware Configuration

Note

Slots 9 through 14 in the Model 808 are used to enclose the embedded hard disk.

HP 9000 Model 822/832/842/852/642/652

822/832/842/852/642/652 Standard Hardware Configuration

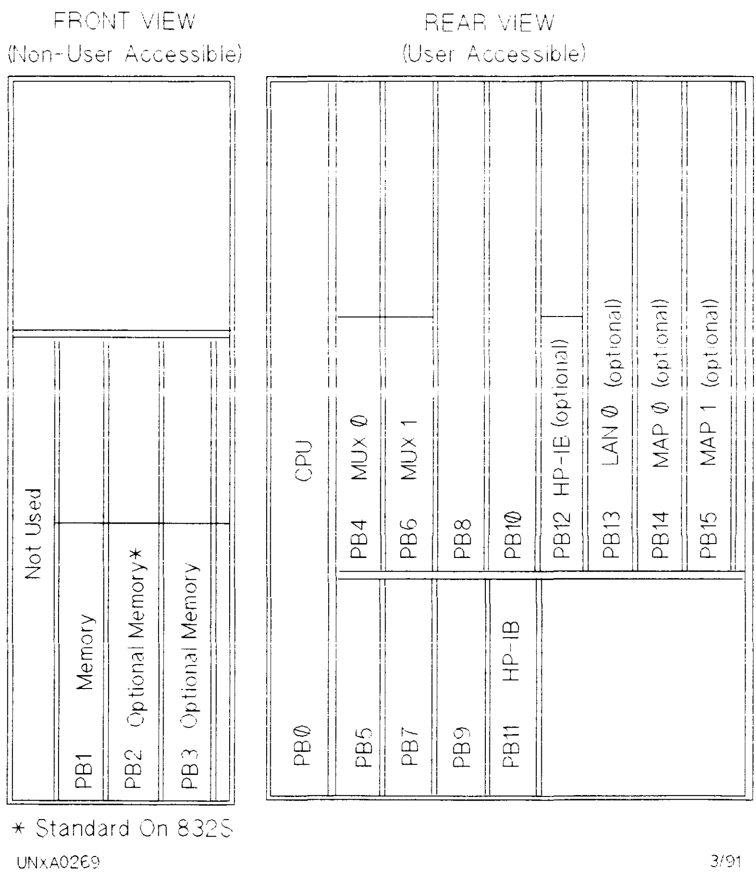


Figure 2-4. 822/832/842/852/642/652 Standard Hardware Configuration

Model 825/835 Without Remote Support

	CIO Slot	825/835 (No Access Port)		MB Slot		
high priority	Ø	HP-IB/HP-FL (root disk)		8		low priority
	1	MUX (console)		7	F	
	2	HP-IB (mt/lp/etc)		6	R	
	3	MUX		5	O	
	4	LAN		4	N	
	5	MUX, HP-IB (disk)[1]		3	T	
	6	HP-IB (disk)	Graphics or CA[2]	2		
	n/a	System Card (825), PDH Card (835)		1		
low priority	n/a	Main CPU Card (825), Processor Card (835)		Ø		high priority

Model 825/835 With Remote Support

	CIO Slot	825/835 (No Access Port)		MB Slot		
high priority	Ø	Access Port (AP)		8		low priority
	1	HP-IB/HP-FL (root disk)		7	F	
	2	HP-IB (mt/lp/etc)		6	R	
	3	MUX (console)		5	O	
	4	LAN		4	N	
	5	MUX, HP-IB (disk)[1]		3	T	
	6	HP-IB (disk)	Graphics or CA[2]	2		
	n/a	System Card (825), PDH Card (835)		1		
low priority	n/a	Main CPU Card (825), Processor Card (835)		Ø		high priority

[1] MUX on multi-user systems, HP-IB (disk) on graphics systems.

[2] Full-size Graphics interface card can go here (recommended)
Full-size Channel Adapter card can go here (825 only).

UNXA0271 3/91

Figure 2-5. 825/835 Standard Hardware Configuration

834 Standard Hardware Configuration

CIO Slot	HP 9000/834		Module	Mid-Bus Slot
0	HP-IB	Optional Memory	32	8
1	Open I/O	Optional Memory	28	7
2	LAN	8 Mb Memory	24	6
3	Optional 2D Accelerator		20	5
4	2D Graphics		16	4
5	Graphics Interface		12	3
6	Future Systems Slot		8	2
N/A	PDH		4	1
N/A	Processor		0	0

UNXA0276 3/91

Figure 2-6. 834 Standard Hardware Configuration

635 Standard Hardware Configuration (no Access Port)

CIO Slot			Mid-Bus Slot
0	HP-IB/HP-FL (root disk)	16 MB Memory Array	8
1	MUX (console)	8MB Memory Array	7
2	HP-IB	Open for Memory Expansion	6
3		Open for Memory Expansion	5
4	LAN	Open for Memory Expansion	4
5		Open for Memory Expansion	3
6		Open for Memory Expansion	2
n/a	PDH Card (835)		1
n/a	Processor Card (835)		0

UNXA0278 3/91

Figure 2-7. 635 Standard Hardware Configuration (no Access Port)

635 Standard Hardware Configuration (Access Port)

CIO Slot			Mid-Bus Slot
0	Access Port (AP)	16 MB Memory Array	8
1	HP-IB/HP-FL (root disk)	8 MB Memory Array	7
2	HP-IB	Open for Memory Expansion	6
3	MUX (Console)	Open for Memory Expansion	5
4	LAN	Open for Memory Expansion	4
5		Open for Memory Expansion	3
6		Open for Memory Expansion	2
n/a	PDH Card (835)		1
n/a	Processor Card (835)		0

UNXA0279 3/91

Figure 2-8. 635 Standard Hardware Configuration (Access Port)

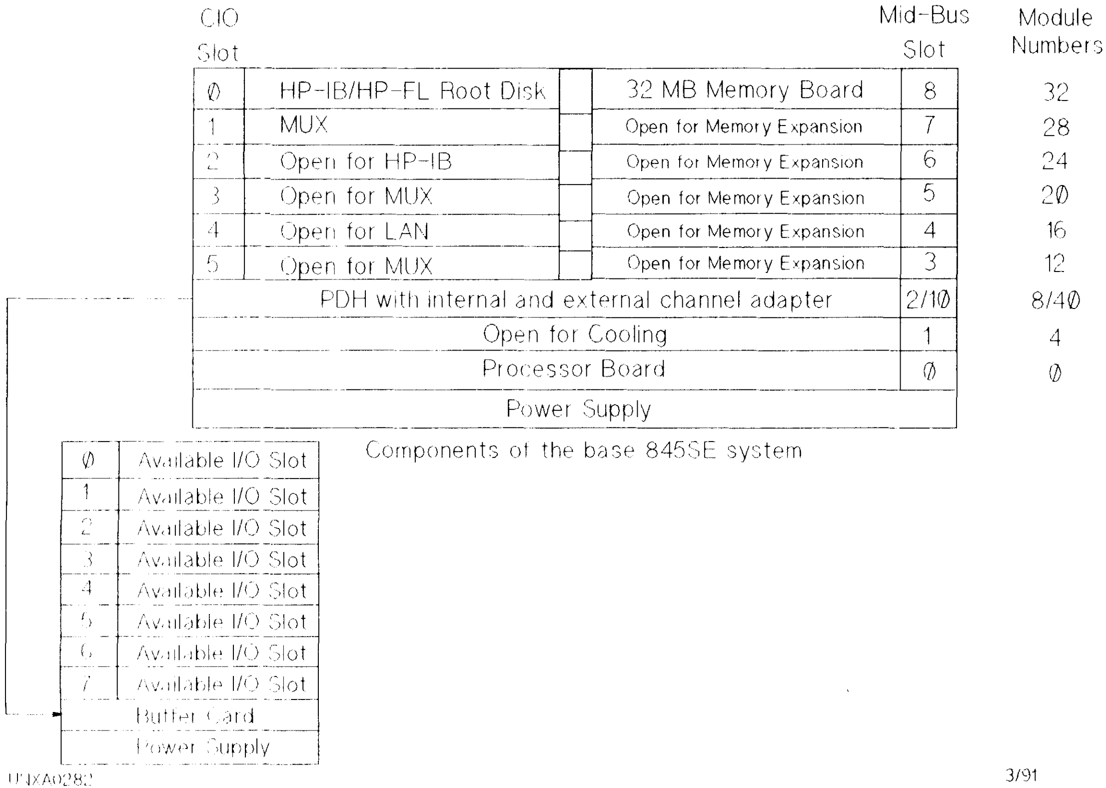


Figure 2-9. 845 Standard Hardware Configuration (no Access Port)

845 Standard Hardware Configuration (Access Port)

CIO Slot			Mid-Bus Slot	Module Numbers
Ø	Access Port	32 MB Memory Board	8	32
1	HP-IB/HP-FL Root Disk	Open for Memory Expansion	7	28
2	Open for HP-IB	Open for Memory Expansion	6	24
3	MUX	Open for Memory Expansion	5	2Ø
4	Open for LAN	Open for Memory Expansion	4	16
5	Open for MUX	Open for Memory Expansion	3	12
	PDH with internal and external channel adapter		2/1Ø	8/4Ø
	Open for Cooling		1	4
	Processor Board		Ø	Ø
	Power Supply			

Components of the base 845SE system with Access Port (ordered separately)

Ø	Available I/O Slot
1	Available I/O Slot
2	Available I/O Slot
3	Available I/O Slot
4	Available I/O Slot
5	Available I/O Slot
6	Available I/O Slot
7	Available I/O Slot
	Buffer Card
	Power Supply

UNXA0283 3/91

Figure 2-10. 845 Standard Hardware Configuration (Access Port)

645 Standard Hardware Configuration

CIO Slot			Mid-Bus Slot	Module Numbers
0	HP-IB/HP-FL Root Disk	32 MB Memory Board	8	32
1	MUX	Open for Memory Expansion	7	28
2	Open for HP-IB	Open for Memory Expansion	6	24
3		Open for Memory Expansion	5	20
4	LAN	Open for Memory Expansion	4	16
5		Open for Memory Expansion	3	12
	PDH with internal channel adapter		2	8
	Open for Cooling		1	4
	Processor Board		0	0
	Power Supply			

UNXA0284 3/91

Figure 2-11. 645 Standard Hardware Configuration

HP 9000 Model 850/855/860/865/870

850/855/860/865/870 Standard Hardware Configuration

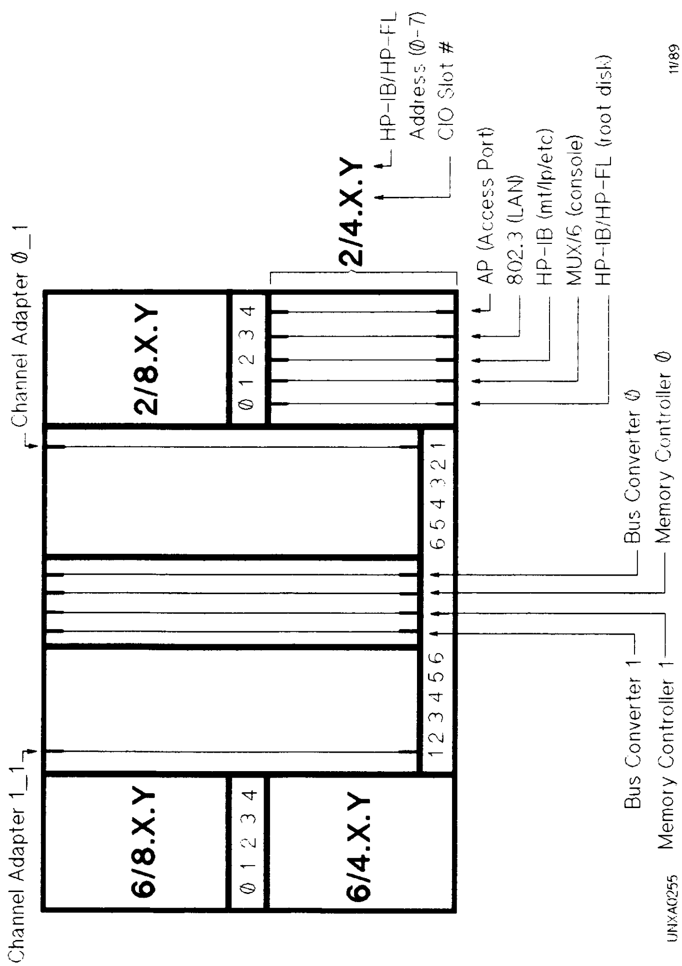


Figure 2-12. 850/855/860/865/870 Standard Hardware Configuration

HP 9000 Model 890

890 Standard Hardware Configuration

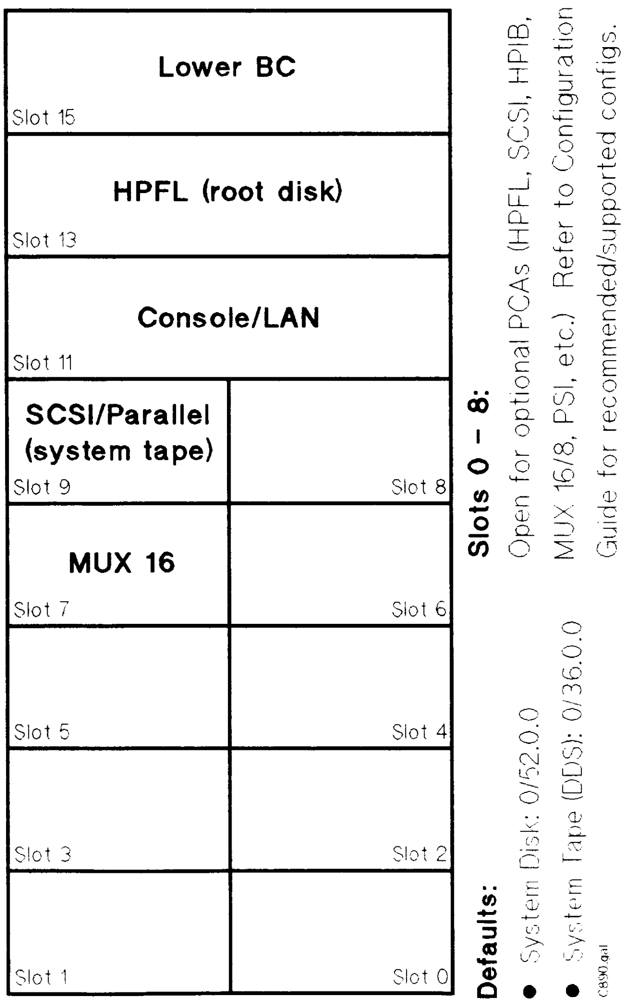


Figure 2-13. 890 Standard Hardware Configuration

Updating HP-UX

Note

Be aware that to migrate HP-UX from a hard partition on the HP-UX 8.0 release to a root logical volume on the 9.0 release, you *must* perform an installation, NOT an update. If this is your intended task, refer to *Installing and Updating HP-UX*, Chapters 1, 2, and 3 for specifics. Do NOT use this section.

- In contrast to an installation, which overwrites files on the root disk, an update provides new functionality without destroying the existing system. You must have a functioning HP-UX system to perform an update. After performing the update you will probably have to edit your customized files or remove certain filesets.
- You update HP-UX (as opposed to install) when you have any of the following situations:
  - You are running an 8.x release and want to move to the 9.0 release, keeping your root file system on a hard partition as it is now.
  - You are running the 9.0 release and you want to add software to that release.
  - You want to convert an existing system to a network distribution server (netdist server).

Loading the TOOL Fileset

If You Have DDS, HP 9114(5), or 9-track Tapes ...

1. Write protect the tape according to its documentation.
2. Insert the tape in its drive. Wait until the drive is ready according to the drive's documentation.
3. Be in the root directory. Execute cd / if necessary.
4. Execute the following command, which assumes a device file named `0m`. You need to specify the appropriate device file name. Wait for the extraction to complete before you type anything. After no files are extracted for a few minutes, stop the command by typing the interrupt key (probably Break or Ctrl-C). Otherwise, it can take a long time for the command to read the entire tape.

```
tar -xvf /dev/rmt/0m TOOL
```

If You Have CD-ROM ...

1. Be in the root directory. Execute cd / if necessary.
2. Execute the following command, which assumes the CD-ROM drive is mounted under the directory named `/UPDATE_CDROM`. If your system differs from this, specify the directory where the CD-ROM drive is mounted. You get an error message if you do not insert the CD-ROM disc or mount the CD-ROM drive (`invalid update source media`). Wait for the extraction to complete.

```
tar -xvf /UPDATE_CDROM/TOOL
```

Options for loading continue on the next page.

If You Use a Netdist Server ...

1. Have the netdist server configured and have the update program available for clients.
2. Copy the 9.0 TOOL fileset from the netdist server to the local client system. HP recommends using FTP in the anonymous mode. The *Installing and Administering ARPA Services* document has information.

   a. Type ftp *netdist_server_name* (Use the name of the server from which you copy `/etc/update`.)

   b. At the login prompt, type: anonymous

   c. At the password prompt, type: ftp

   d. You should see the `ftp>:` prompt. (If you do not, the previous steps did not succeed, and you must try again.)

      To extract the fileset, type:

      get dist/TOOL.800 /tmp/TOOL

   e. You should see messages like this:

      ```
      Opening data connection for dist/ ...
      nn bytes received ...
      ```

   f. When the `ftp>:` prompt reappears, type: bye, which returns you to the client system.

3. From the client system, type:

   cd /

4. To complete the extraction, type the following command. Be aware several minutes can pass before the first `TOOL` file is extracted. Then, the extraction occurs rapidly.

   tar -xvf /tmp/TOOL

Starting *update*

Having booted HP-UX from the primary boot path and having become the root user, start the loading by executing the following command:

/etc/update

You get a screen like the following one.

```
--------------------------------------------------------------------

INSTALL                    Main Menu

Highlight an item and then press "Return" or
"Select Item".  To refresh the screen, press CTRL-L.

Source:  Tape Device          Destination: Local System
         /dev/rmt/0m                       /


       Select ALL Filesets on the Source Media ->
       Select Filesets for a Minimum System ->
       Select/View Partitions and Filesets ->
       Enter Codeword ->

       How to Use Install

 Help        Shell   Select                        Exit
                      Item                        Update
--------------------------------------------------------------------
```

Option 1: Select All Filesets on the Source Media ->

Selecting `Select All Filesets on the Source Media ->` automatically loads every fileset on the source media. If you activate it, the `Select All...` screen appears. At this point, if you wish to examine the filesets selected for loading, activate `Modify/View Partitions and Filesets`, which gives you another opportunity to load or deselect filesets. When you are ready, activate `Start Loading`.

Option 2: Select Filesets for a Minimum System->

Activating this option causes the following minimum set of filesets (plus the appropriate user license) to be selected:

```
UX-CORE          CORE-DIAG
KERN-BLD         C-MIN
TOOL             EDITORS
CORE-SHLIBS      CMDS-MIN
```

Checklist Item	Information and Tasks
Related Information	The `Select Filesets for a Minimum System` screen appears. You can update to a minimal system now and then, after the update, use *update*(1M) to load additional filesets.
Load the filesets	Activate `Start Loading` from this screen.

Option 3: View/Select Partitions and Filesets->

You can use the following screen to tailor the functionality you load onto your system.

```
------------------------------------------------------------------
          View or Select Individual Partitions


Mark "y" or "n" to make a selection.
Press "Main" to return to the partition selection
screen.

Select   Partition       Partition Description     Size Kb

n      DIAGNOSTICS       Hardware Diagnostics       37663
p      NETWORKING        Networking Products         8919
y      NLS               Native Language Support      472
y      OS-ADMIN          Rec. Administration Cmds    2292
y      OS-CORE           Recommended System Core     5517
y      OS-FEATURES       Selectable OS Features      8176
y      PROG-LANGUAGES    Programming Languages       8542
y      REFERENCE-DOC     Reference Manual Pages       348
y      SHARED-LIBS       Runtime Shared Libraries    2757
y      WINDOWS           Windowing Products           102

Help      Shell     Start     Disk   View     Global Main
                    Loading   Space  Filesets Select
------------------------------------------------------------------
```

Checklist Item	Information and Tasks
Using Y, N, and P	Y selects and N deselects a partition. If you select or deselect filesets in a partition, a P appears to indicate a partial selection.
Update checks dependencies	The install program does not let you load (or not load) partitions without accommodating required dependencies.
You eventually start loading	After you select or deselect the partitions or filesets, activate `Start Loading`.

Reconfirm Having Enough Disk Space

Your current system might need to grow to accommodate your 9.0 products. If you do not have enough space, you have two alternatives.

1. Free up disk space on your existing system, or
2. Shut down your system, add another disk (or disks), restart the system, and perform the update.

How to Free Disk Space

1. Deselect filesets to load during an update.
2. Remove some existing files on the system.
3. Mount another file system.
4. Create symbolic links.
5. Use a combination of the above methods.

Fixing Overflow with Symbolic Links

- A symbolic link transparently links a file or directory to another file or directory. This is often used when a file system overflows. On overflow, all the files in a selected directory are moved to a new directory on a new file system. The original directory is then symbolically linked to the new directory where the data then resides. The user can access the data using the original path name (using the symbolic link) or directly using the new path name.
- The */bin/ln* command and parameters required to link one directory to another are shown below:

  # ln -s new_directory old_directory

Key	Operation
-s	Causes *ln* to create a symbolic link.
new_directory	Full path name of new directory where the moved files reside.
old_directory	Full path name of original directory where files resided previously.

- Example:

  # mkdir /mnt/users
  # cp -rp /users/ /mnt/users && rm -rf /users/**
  # ln -s /mnt/users /users

Remove Unwanted Software Using rmfn(1M)

After you finish installing HP-UX, you might have software you do not need. The *rmfn*(1M) (remove functionality) utility lets you remove unnecessary system software. The *HP-UX Reference* manual has complete information for this command.

Use *rmfn* with Caution

Although *rmfn*(1M) checks dependencies to prevent you from inadvertently removing functionality, you still need to be cautious. The command is designed to help you quickly remove major pieces of software. It is important to avoid making mistakes.

Important Points About Using rmfn(1M)

Checklist Item	Information and Tasks
Filesets and partitions	The filesets and partitions that the `rmfn` command displays depend on the contents of the directories named `/etc/filesets` and `/system`. Do not change the contents of these directories or the command will display an inaccurate list of filesets.
Dependencies	The *rmfn*(1M) command only lets you remove filesets that will not subsequently harm the integrity of your system. The *rmfn*(1M) command does not allow you to remove a minimum set of filesets needed by the system. For example, you cannot remove `UX-CORE`.
Remote systems	The *rmfn*(1M) command does not remove files on a remote, mounted system (NFS).
Symbolic links	As the *rmfn*(1M) command removes a symbolic link contained in a fileset, it does not remove a symbolic link's target file. A target file remains intact until *rmfn*(1M) removes the fileset containing the target file.

How to Use *rmfn*

To use the remove fileset command, log in a system administrator and execute:

#/etc/rmfn

The example below shows a typical main screen.

```
------------------------------------------------------------

rmfn                Partitions

Press "y" to select an entire partition for deletion.
Press "n" to undo a selection.  Press the "Select
Filesets" key to view the filesets within a
partition.  Press the "Start Removing" key when
selection is complete.

Mark                                                    Size
Part.                 Arch. Part. Desc.                  Kb.

p     DIAGNOSTICS     800   Hardware Diagnostics       37663
n     NETWORKING      800   Networking Products         8919
n     NLS             800   Native Lang. Support         472
n     OS-ADMIN        800   Recommended Admin.Cmds      2292
n     OS-CORE         800   Recommended System Core     5517
n     OS-FEATURES     800   Selectable OS Features      8176
y     PROG-LANG.      800   Programming Languages       8542
n     REFERENCE-DOC   800   Reference Manual Pages       348
n     SHARED-LIBS     800   Runtime Shared Libs         2757
n     WINDOWS         800   Windowing Products           102

[Help]                                [Shell] [Start Rem]

------------------------------------------------------------
```

The following page explains the options and use.

Checklist Item	Information and Tasks
Arrow keys	Move the highlight among the items in a screen as implied by the key.
Y	This key selects a highlighted item for deletion. The letter `y` appears on the screen under `Mark`.
N	This key undoes the effects of a selection. For example, if you select a partition with Y, pressing N deselects the partition. The letter `n` appears on the screen under `Mark`.
`Help`	Pressing this softkey explains how to use the `rmfn` command.
`Shell`	Pressing this softkey lets you escape to a shell to execute HP-UX commands. Type exit on a shell command line to return to the `rmfn` screen.
`Exit rmfn`	Pressing this softkey terminates the removal process and exits the `rmfn` command.
`Se-lect Filesets`	For a highlighted partition, pressing this softkey lets you select individual filesets in that partition for deletion. If you do this, a `p` appears on the screen under `Mark` to indicate a partial selection of filesets.
`View Selected`	Pressing this softkey lists the names and sizes of the partitions and filesets selected for removal.
`Start Removing`	Pressing this softkey removes the selected partitions and filesets from your system.

HP-UX version 8.0 Boot Paths and Installation Commands

- This section is included to assist individuals in installing HP-UX release 8.0.

Caution Do NOT use information in this section when installing HP-UX version 9.0. The information presented here is NOT compatible with HP-UX version 9.0. Refer to *Installing HP-UX* earlier in this section for 9.0 installation information .

- For specifics on installing HP-UX version 8.0 on models or configurations not included in this section, see the manual *Installing and Updating HP-UX*, HP part number 92453-90035.

808/815 Boot Paths and Installation Commands

- Installation of the 808/815 (with Mag Tape Drive)

HP-IB Card (Slot 5, Module No. 20)	**Installation Commands and Boot Paths**
System Disk: Address 0 Mag Tape Drive: Address 4	Command to Boot from Tape: *hpux -a (20.0) (20.4;0xa0000,1)* Primary Boot Path: 20.0[1] Alternate Boot Path: 20.4[1]

1 Default boot path.

- Installation of the 808/815 (with Cartridge Tape Drive)

HP-IB Card (Slot 5, Module No. 20)	**Installation Commands and Boot Paths**
System Disk: Address 0 Cartridge Tape Drive: Address 3	Command to Boot from Tape: *hpux -a (20.0) (20.3;0x400000)* or *hpux (20.3;0x400000)* or *hpux install* Primary Boot Path: 20.0[1] Alternate Boot Path: 20.3[1]

1 Default boot path.

822/832/842/852/642/652 Boot Paths and Installation Commands

- Installation of the 822/832/842/852/642/652 (with DDS)

HP-IB Card (Slot 11, Module No. 44)	**Installation Commands and Boot Paths**
System Disk: Address 0 DDS Drive: Address 7	Command to Boot from Tape: *hpux -a (44.0) (44.7;0xa0000,1)* or *hpux (44.7;0xa0000,1)* or *hpux install* Primary Boot Path: 44.0[1] Alternate Boot Path: 44.7[1]

1 Default boot path.

- Installation of the 822/832/842/852/642/652 (with Cartridge Tape Drive)

HP-IB Card (Slot 11, Module No. 44)	**Installation Commands and Boot Paths**
System Disk: Address 0 Cartridge Tape Drive: Address 3	Command to Boot from Tape: *hpux -a (44.0) (44.4:0x400000)* or *hpux (44.4;0x400000)* or *hpux install* Primary Boot Path: 44.0[1] Alternate Boot Path: 44.3[1]

1 Default boot path.

825/835 Boot Paths and Installation Commands

- Installation of the 825/835 (with Mag, or DDS Tape Drive, no Access Port, all Consoles)

CIO Configuration	Installation Commands and Boot Paths
CIO Slot 0: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 2: HP-IB device adapter with mag, or DDS tape drive at Address 3	Command to Boot from Tape: *hpux install* or *hpux -a (4.0.0) (4.2.3;0xa0000,1)* or *hpux (4.2.3;0xa0000,1)* Primary Boot Path: 4.0.0[1] Alternate Boot Path: 4.2.3[1]

1 Default boot path.

- Installation of the 825/835 (with Mag, or DDS Tape Drive, Access Port, all Consoles)

CIO Configuration	Installation Commands and Boot Paths
CIO Slot 0: Access Port (AP) CIO Slot 1: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 2: HP-IB device adapter with mag, or DDS tape drive at Address 3	Command to Boot from Tape: *hpux install* or *hpux -a (4.1.0) (4.2.3;0xa0000,1)* or *hpux (4.2.3;0xa0000,1)* Primary Boot Path: 4.1.0[1] Alternate Boot Path: 4.2.3[1]

1 Default boot path.

- Installation of the 825/835 (with CTD, no Access Port, Non-Graphics Console)

CIO Configuration	Installation Commands and Boot Paths
CIO Slot 0: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 6: HP-IB device adapter with CTD at Address 3	Command to Boot from Tape: *hpux -a (4.0.0) (4.6.3;0x400000)* or *hpux (4.6.3;0x400000)* Primary Boot Path: 4.0.0[1] Alternate Boot Path: 4.6.3[1]

1 Default boot path.

- Installation of the 825/835 (with CTD, Access Port, Non-Graphics Console)

CIO Configuration	Installation Commands and Boot Paths
CIO Slot 0: Access Port (AP) CIO Slot 1: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 6: HP-IB device adapter with CTD at Address 3	Command to Boot from Tape: *hpux -a (4.1.0) (4.6.3;0x400000)* or *hpux (4.6.3;0x400000)* Primary Boot Path: 4.1.0[1] Alternate Boot Path: 4.6.3[1]

1 Default boot path.

- Installation of the 825/835 (with CTD, no Access Port, Graphics Console)

CIO Configuration	**Installation Commands and Boot Paths**
CIO Slot 0: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 5: HP-IB device adapter with CTD at Address 3	Command to Boot from Tape: *hpux -a (4.0.0) (4.5.3;0x400000)* or *hpux (4.5.3;0x400000)* Primary Boot Path: 4.0.0[1] Alternate Boot Path: 4.5.3[1]

1 Default boot path.

- Installation of the 825/834/835/845/635/645 (with CTD, Access Port, Graphics Console)

CIO Configuration	**Installation Commands and Boot Paths**
CIO Slot 0: Access Port (AP) CIO Slot 1: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 5: HP-IB device adapter with CTD at Address 3	Command to Boot from Tape: *hpux -a (4.1.0) (4.5.3;0x400000)* or *hpux (4.5.3;0x400000)* Primary Boot Path: 4.1.0[1] Alternate Boot Path: 4.5.3[1]

1 Default boot path.

- Installation of the 825/835 (with CTD, no Access Port, A1074A GAI Card)

CIO Configuration	Installation Commands and Boot Paths
CIO Slot 0: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 2: HP-IB device adapter with CTD at Address 3	Command to Boot from Tape: *hpux -a (4.0.0) (4.2.3;0x400000)* or *hpux (4.2.3;0x400000)* Primary Boot Path: 4.0.0[1] Alternate Boot Path: 4.2.3[1]

1 Default boot path.

- Installation of the 825/835 (with CTD, Access Port, A1074A GAI Card)

CIO Configuration	Installation Commands and Boot Paths
CIO Slot 0: Access Port (AP) CIO Slot 1: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 5: HP-IB device adapter with CTD at Address 3	Command to Boot from Tape: *hpux -a (4.1.0) (4.5.3;0x400000)* or *hpux (4.5.3;0x400000)* Primary Boot Path: 4.1.0[1] Alternate Boot Path: 4.5.3[1]

1 Default boot path.

834 Boot Paths and Installation Commands

- Installation of the 834 (with CTD, no Access Port, all Consoles)

CIO Configuration	Installation Commands and Boot Paths
CIO Slot 0: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 2: HP-IB device adapter with CTD at Address 3	Command to Boot from Tape: *hpux install* or *hpux -a (4.0.0) (4.2.3;0x400000)* or *hpux (4.2.3;0x400000)* Primary Boot Path: 4.0.0[1] Alternate Boot Path: 4.2.3[1]

1 Default boot path.

635 Boot Paths and Installation Commands

- Installation of the 635 (with Mag, or DDS Tape Drive, no Access Port, all Consoles)

CIO Configuration	Installation Commands and Boot Paths
CIO Slot 0: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 2: HP-IB device adapter with mag, or DDS tape drive at Address 3	Command to Boot from Tape: *hpux install* or *hpux -a (4.0.0) (8.2.3;0xa0000,1)* or *hpux (4.2.3;0xa0000,1)* Primary Boot Path: 4.0.0[1] Alternate Boot Path: 4.2.3[1]

1 Default boot path.

- Installation of the 635 (with CTD, no Access Port, all Consoles)

CIO Configuration	Installation Commands and Boot Paths
CIO Slot 0: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 2: HP-IB device adapter with mag tape drive at Address 3	Command to Boot from Tape: *hpux install* or *hpux -a (4.0.0) (8.2.3;0x400000)* or *hpux (4.2.3;0x400000)* Primary Boot Path: 4.0.0[1] Alternate Boot Path: 4.2.3[1]

1 Default boot path.

- Installation of the 635 (with Mag Tape Drive, no Access Port, all Consoles)

CIO Configuration	Installation Commands and Boot Paths
CIO Slot 1: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 2: HP-IB device adapter with mag tape drive at Address 3	Command to Boot from Tape: *hpux install* or *hpux -a (4.1.0) (4.2.3;0xa0000,1)* or *hpux (4.2.3;0xa0000,1)* Primary Boot Path: 4.1.0[1] Alternate Boot Path: 4.2.3[1]

1 Default boot path.

- Installation of the 635 (with CTD, Access Port, all Consoles)

CIO Configuration	Installation Commands and Boot Paths
CIO Slot 1: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 2: HP-IB device adapter with mag tape drive at Address 3	Command to Boot from Tape: *hpux install* or *hpux -a (4.1.0) (4.2.3;0x400000)* or *hpux (4.6.3;0x400000)* Primary Boot Path: 4.1.0[1] Alternate Boot Path: 4.6.3[1]

1 Default boot path.

845/645 Boot Paths and Installation Commands

- Installation of the 845/645 (with Mag, or DDS Tape Drive, no Access Port, all Consoles)

CIO Configuration	Installation Commands and Boot Paths
CIO Slot 0: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 2: HP-IB device adapter with mag, or DDS tape drive at Address 3	Command to Boot from Tape: *hpux install* or *hpux -a (8.0.0) (8.2.3;0xa0000,1)* or *hpux (8.2.3;0xa0000,1)* Primary Boot Path: 8.0.0[1] Alternate Boot Path: 8.2.3[1]

1 Default boot path.

- Installation of the 845/645 (with CTD, no Access Port, all Consoles)

CIO Configuration	Installation Commands and Boot Paths
CIO Slot 0: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 2: HP-IB device adapter with CTD at Address 3	Command to Boot from Tape: *hpux install* or *hpux -a (8.0.0) (8.2.3:0x400000)* or *hpux (8.2.3;0x400000)* Primary Boot Path: 8.0.0[1] Alternate Boot Path: 8.2.3[1]

1 Default boot path.

- Installation of the 845/645 (with Mag, or DDS Tape Drive, Access Port, all Consoles)

CIO Configuration	**Installation Commands and Boot Paths**
CIO Slot 0: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 2: HP-IB device adapter with mag, or DDS tape drive at Address 3	Command to Boot from Tape: *hpux install* or *hpux -a (8.1.0) (8.2.3;0xa0000,1)* or *hpux (8.2.3;0xa0000,1)* Primary Boot Path: 8.1.0[1] Alternate Boot Path: 8.2.3[1]

1 Default boot path.

- Installation of the 845/645 (with CTD, Access Port, all Consoles)

CIO Configuration	**Installation Commands and Boot Paths**
CIO Slot 0: HP-IB/HP-FL device adapter with system disk at Address 0 CIO Slot 2: HP-IB device adapter with mag tape drive at Address 3	Command to Boot from Tape: *hpux install* or *hpux -a (8.1.0) (8.2.3;0x400000)* or *hpux (8.2.3;0x400000)* Primary Boot Path: 8.1.0[1] Alternate Boot Path: 8.2.3[1]

1 Default boot path.

850/855/860/865/870 Boot Paths and Installation Commands

- Installation of the 850/855/860/865/870 (with Mag Tape Drive)

CIO Configuration	Installation Commands and Boot Paths
CIO cardcage 0_1, slot 0: HP-IB/HP-FL device adapter with system disk at Address 0 CIO cardcage 0_1, slot 2: HP-IB/HP-FL device adapter with system disk at Address 3	Command to Boot Tape: *hpux install* or *hpux -a (2/4.0.0) * *(2/4.2.3;0x0a0000,1)* or *hpux (2/4.2.3;0x0a0000,1)* Primary Boot Path: 2/4.0.0[1] Alternate Boot Path: 2/4.2.3[1]

1 Default boot path.

- Installation of the 850/855/860/865/870 (with CTD)

CIO Configuration	Installation Commands and Boot Paths
CIO cardcage 0_1, slot 0: HP-IB/HP-FL device adapter with system disk at Address 0 CIO cardcage 0_1, slot 2: HP-IB/HP-FL device adapter with CTD at Address 3	Command to Boot from Tape: *hpux install* or *hpux -a (2/4.0.0) * *(2/4.2.3;0x400000)* or *hpux (2/4.2.3;0x400000)* Primary Boot Path: 2/4.0.0[1] Alternate Boot Path: 2/4.2.3[1]

1 Default boot path.

3

Diagnostics and Support Tape

HP-UX 9.0 Diagnostic Passwords

Beginning with HP-UX release 9.0, most diagnostics and support tools that run on HP 9000 Series 600/800 systems require a password.

Before invoking the Diagnostic User Interface (DUI) or the SupportWave Support Tools Manager, enter a password using the *suplicen* command.

Individual HP 9000 Series 600/800 off-line diagnostics and utilities distributed on the Support Tape are also passworded beginning with HP-UX release 9.0. When these diagnostics and utilities are run from the Support Tape's *ISL* prompt, the user will be prompted for a password. Once a correct password has been entered, the diagnostics and utilities run in the conventional manner.

Diagnostic Categories

- HP 9000 Series 600/800 on-line diagnostics are now divided by licensing requirements into three categories.

NO CHARGE	Diagnostics available with no password
LICENSED	Diagnostics available to customers with a password
HP USE ONLY	Diagnostics and utilities available for Hewlett-Packard support personnel ONLY. Password required.

No Charge Diagnostics

- The following on-line diagnostics are available at no charge to customers.

LOGTOOL	System logging tool
SYSMAP	System mapping tool
VERIFY	System verification tool

- The following off-line utility is available at no charge to customers.

IOMAP	Maps system hardware from the *ISL* prompt

Diagnostic Password Types

- There are four types of diagnostic passwords.

HP INTERNAL	Unlocks all diagnostics.
CLASS	Unlocks diagnostics for classes of systems (such as lower end system, mid range, and high end system). Valid for six months.
SYSTEM SPECIFIC	Unlocks licensed diagnostics in certain high security situations. A customer using these passwords will require a password for each system (introduction in mid 1993).
TEMPORARY	Unlocks licensed diagnostics for 1 to 20 days.

Entering the Password

- After root logon, enter the diagnostic password using the *suplicen* command.

  \# *suplicen*

  Enter the diagnostic password when prompted. Once this has been accomplished, run *sysdiag*.

Diagnostic User Interface (DUI)

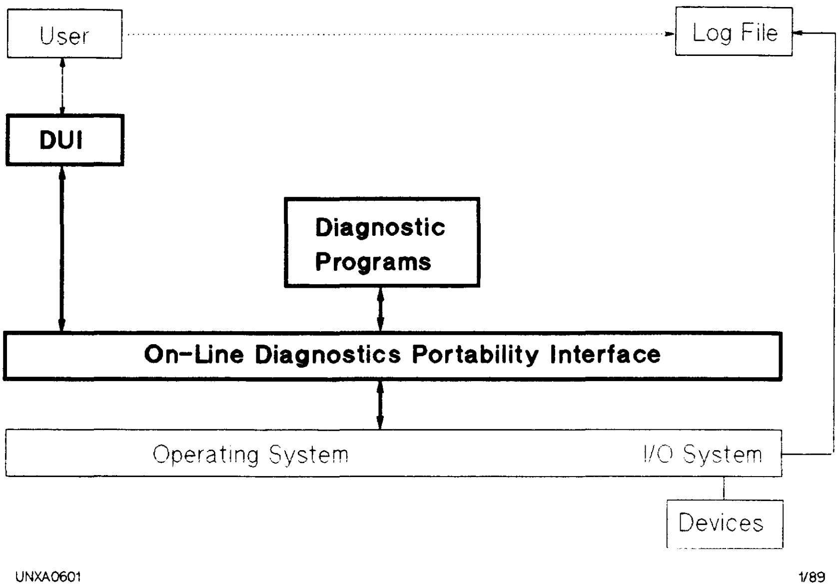


Figure 3-1. HP-UX On-line Diagnostic Subsystem

Running DUI

- DUI is invoked by running *sysdiag*.

  \# *sysdiag*

```
******************************************************
******                                          ******
******        ONLINE DIAGNOSTIC SYSTEM          ******
******                                          ******
******   (C) Copyright Hewlett Packard Co.      ******
******        1987, 1989, 1990, 1992            ******
******          All Rights Reserved             ******
******                                          ******
******          DUI Version A.02.21             ******
******   Diagnostic Monitor Version  A.02.19    ******
******                                          ******
******************************************************

    Type "HELP" for assistance.

DUI >
```

Key *DUI* Commands

HELP	Gives help on commands, syntax, and diagnostic descriptions.
LIST	Lists the diagnostics available on the system, diagnosable HP product numbers, etc.
RUN	Runs the specified diagnostic.

HELP

- DUI> *HELP*

The following is a partial list of commands available in *DUI*. Information about a particular command can be obtained by typing *HELP* followed by the command name.

Command	**Description**
ABORT	Abort a diagnostic system program.
CI	Invoke the system command interpreter.
DO	Re-execute a command.
EXIT	Exit the DUI.
HARDCOPY	Echo input/output data to a hardcopy device.
HELP	Help facility.
INSTALL	Add a program to the diagnostic system.
LIST	Display information on installed programs.
LISTREDO	Display the command history stack.
OUTFILE	Echo input/output to a file.
REDO	Edit and re-execute a command.
RUN	Run a diagnostic system program.
UNLOCK	Unlock a malfunctioning locked device.

Diagnostic Descriptions.

- To obtain a diagnostic's functional description enter *HELP diagnostic_name*. Example:

  DUI > HELP flexdiag

LIST

- To list all diagnostics

  DUI > *LIST*

```
Enter the full directory path in which the programs are
installed.
[<cr> /usr/diag/bin] :

AFIDAD      CARTDIAG    CENTPBA     CIPERLPD
CONSOLAN    CS80DIAG    DASSDIAG    DIAG7478
DTDUTIL     FLEXDIAG    GP3DDIAG    GPIODAD
GS2DDIAG    GS3DDIAG    HPFLDIAG    HPIBDAD
HPIBDDS     HPIBDIAG    LAN3PBB     LANDAD
LOGTOOL     MEMDIAG     MUXDIAG     OPDIAG
OSIDAD      PMUXDIAG    PRNT5000    PSIDAD
REELDIAG    SCSICD      SCSICIO     SCSIDDS
SCSIDISK    SCSIPBA     SCSIQIC     SCSIREEL
SS80DIAG    SYSMAP
```

- To view the products diagnosed by a specific diagnostic, enter *list long diagnostic_name*. Example:

  DUI > *LIST LONG flexdiag*

```
Enter the full directory path in which the programs are installed.
[<cr> /usr/diag/bin] :

           Program  Program         Catalog
Name       Version  Type            Languages      Devices
=======    =====    =========       =======        ========
FLEXDIAG   A.02.15  DIAGNOSTIC      n-computer     HP7936FL
                                                   HP7937FL
                                                   HPC2201A
                                                   HPC2204A
                                                   HPC2251S
                                                   HPC2252B
                                                   HPC2252HA
                                                   HPC2254
                                                   HPC2254HA
```

- To find a diagnostic for a product.

  DUI > *list product_number*

  product_number = HP product number of the device to be diagnosed

RUN

- The run command and common command modifiers are shown below.

  DUI > *[RUN] diagnostic [command_modifier [command_modifier] . . .]*

  Where:

diagnostic	Name of diagnostic to execute
command_modifier:	pdev = *physical path to device*
	sections = *section range list*
	steps = *step range list*
	errcount = *number*
	loopcount = *number*
	background
	erronly
	errpause

DUI RUN Example.

•DUI > *RUN SCSIDISK PDEV=52.6.0* \
SECTIONS=10 LOOPCOUNT=2 ERRCOUNT=10

This command runs *scsidisk* on the disk located at physical path *52.6.0*. The diagnostic will run section 10 twice unless the error count exceeds 10.

Frequently Used *DUI* Commands

- REDO

  Allows editing and execution of a previous command.

- HARDCOPY [ON | OFF]

  Prints screen information on a printer. Hardcopy *ON* begins storing information into a print file. Hardcopy *OFF* places an end of file mark on the data and sends the completed print file to the LP spooler to be printed.

- OUTFILE [filename | OFF]

  Copies all information displayed on the screen to a file on disk. Outfile *filename* begins storing information into *filename*. Outfile *OFF* writes an end of file mark and closes *filename*.

- ABORT

  Aborts diagnostic (PID required).

- CI

  Suspends DUI; returns to HP-UX shell. *exit* returns to DUI.

Obtaining Diagnostic Sectioning Information

- DUI 10> HELP CS80DIAG SECTIONS

The following sections are available:

DEFAULT

Section 10	Diagnostic Trouble Tree—This section will execute the fault isolating diagnostic trouble tree.
Section 17	External Exerciser—This section provides the user with interactive access to the disk drive's internal diagnostics and utilities.

DUI > *LOGTOOL*

•DUI > *LOGTOOL*

LOGTOOL (System and Memory Log Analysis Tool) is useful in troubleshooting intermittent problems. It provides the following:

- System log files for both hardware and software.
- Memory log files for logging priority errors.
- Log file maintenance commands—Clear, Delete, and Switch.
- Log file decoding and display.

Miscellaneous *LOGTOOL* **Commands**

- DISPLAYLOG

  Display I/O entries as information is logged. (This command may appear to be hung since it is waiting for realtime I/O entries)

- EXIT

  Exit *LOGTOOL*, and return to *DUI*.

- HELP

  Help about running *LOGTOOL*.

- REDO

  Edit the last line of text entered.

LOGTOOL System Logfile Commands

- LIST

  Lists the contents of closed system log file(s). (see SWITCHLOG below)

- PURGESYSLOG

  Delete the specified system log file(s).

- PURGEWORK

  Delete specified *work* files from the disk.

- SELECT

  Select specified records from system log files.

- STATUS [detail]

  Report on the status of all system log files. An asterisk (*) indicates the current or active log file being logged to. Status *DETAIL* displays log file starting date/time, number of records in log, etc.

- SWITCHLOG

  Cause the system to close the active log file and start a new one. This command must be executed to see the latest information logged using the *LIST* command.

- TYPES

  Description of system log file *types*.

Examining the Current *LOGTOOL* System Log

- To read the latest system log file:

  LOGTOOL> *STATUS DETAIL* (note the bottom log number)

  LOGTOOL> *SWITCHLOG*

  LOGTOOL> *LIST LOG=lognumber* (*lognumber* is the bottom log number found with *status detail* above)

DUI > *SYSMAP*

•DUI > *SYSMAP*

SYSMAP provides on-line system configuration information in the following output maps:

- *CPUMAP*

  Data includes information about CPUs, coprocessors, and caches.

- *MEMMAP*

  Data includes information about controller and array sizes, interleaved status, and enabled status.

- *MODULEMAP*

  Data includes information about CPUs, memories, graphic cards, and other modules that reside on the system bus.

- *IOMAP*

  A map of all I/O devices configured into the system having CONFIRM mode On; shows devices actually connected.

SupportWave

SupportWave is an integrated support tools platform that provides a single point access for:

- Diagnostics
  - Diagnostics provide the capability to interact with the device firmware, and to perform other tests. Diagnostics are expected to provide problem isolation to the field replaceable unit (FRU) level. When invoked from the Support Tool Manager, diagnostics execute their built-in trouble tree. Diagnostics can also be invoked in an expert mode and individual tests run.
- Verifiers
  - Verifiers provide a simple test of the component to give a go/no go answer, indicating whether it is functioning correctly.
- Exercisers
  - Exercisers provide the ability to provide continuous stress on the system or subsystem. This is useful to provide robust verification, and to help isolate intermittent errors.

Running SupportWave

Support Wave has three interfaces, each is called Support Tool Manager.

- To run the graphical user interface called *XSTM* on an X11 windows terminal:

  # */bin/xstm*

- To run the command line interface called *CSTM*:

  # */bin/cstm*

- To run the menu interface called *MSTM*:

  # */bin/mstm*

Running SupportWave's Command Line Interface (CSTM)

- CSTM is invoked by running *cstm*.

  \# *cstm*

```
************************************************************
******                                                ******
******            SUPPORT TOOLS MANAGER               ******
******            Command Line Interface              ******
******                                                ******
******               Version A.00.13                  ******
******                                                ******
******          Part Number B24780-10002              ******
******                                                ******
******    (C) Copyright Hewlett Packard Co.           ******
******             All Rights Reserved                ******
******                                                ******
******             DUI Version A.02.21                ******
******    This program is intended for use by         ******
******    trained HP support personnel only.          ******
******                                                ******
************************************************************

Please type HELP or ? to list available commands.

CSTM>
```

Key *CSTM* Commands

HELP	Displays the help menu or specific information on a specified command.
MAP	Display a map of system devices.
DIAGNOSE	Use to diagnose a specified device.
EXERCISE	Use to exercise a specified device.
VERIFY	Use to verify a specified device or ALL devices

HELP

- CSTM> *HELP*

The following is a partial list of commands available in *CSTM*. Information about a particular command can be obtained by typing *HELP* followed by the command name.

Command	Description
ABORT	Abort a all currently executing processes and exit.
CI	Escape to the shell. *exit* returns to CSTM.
DIAGNOSE	Diagnose a specified device.
EXIT	Wait for all currently executing processes to complete and then exit.
EXERCISE	Exercise a specified device.
HELP	Display this help menu or more specific information on a specified command.
MAP	Display a map of system devices.
TIME	Set a default time value (how many minutes the test should run).
VERIFY	Verify a specified device or ALL devices.
VIEWLOG	View the session log or test results log.

CSTM Run Examples.

•CSTM> *VERIFY CPU*

This command runs verifies the cpu.

•CSTM> *DIAGNOSE 52.6.0*

This runs the appropriate diagnostic for the device at physical address 52.6.0.

•CSTM> *EXERCISE 52.6.0 TIME 5*

This command exercises the device at physical device 52.6.0 for a period of 5 minutes.

Support Tape

The Support Tape allows diagnosis and repair of operating systems and hardware that cannot be booted from the system disk. The Support Tape runs on minimal hardware, with or without the system disk.

The Support Tape is intended to:

- Provide a vehicle for distributing ISL-based support tools
- Host the on-line diagnostics subsystem environment (DUI)
- Provide a means to recover an unbootable HP-UX operating system on HP 9000 Series 800 computer systems.

HP-UX Off-line Diagnostics and Utilities

Running Off-line Diagnostics and Utilities from Support Tape

- To run *IOMAP* from reset or transfer of control

```
Boot from primary boot path (Y or N)?> n
Boot from alternate path (Y or N)?> n
Enter boot path or ?> 4.2.3 (physical address of
Support Services Media device)
.
Booting
.
Console IO Dependent Code (IODC) revision 4
Boot IO Dependent Code (IODC) revision 4
.
Interact with IPL (Y or N)?> y
.
Hard Booted
.
ISL Revision A.00.02  June 22, 1989
ISL> IOMAP
```

Note A complete listing of off-line diagnostics and utilities follows.

Available Off-line Diagnostics and Utilities

- Diagnostics and utilities available from Support Tape's *ISL* prompt

Name	HP Mode Number	Description
SS_CONFIG	All HPPA SPUs	Stable Store Config. Utility
HPUX	All HPPA systems	HP-UX Loader Utility
IOMAP	All HPPA systems	Input/Output Map Utility
RECOVERY	All HPPA systems	Support Kernel Loader
CLKUTIL	All HPPA SPUs	Clock Utility
BCDIAG	All CIO SPUs	A1126A Bus Converter Diag.
A1002AI	825/832/834/835/842/845/852/635/645	A1002A SPU I/O Diag.
A1002AM	825/834/835/845/635/645	A1002A SPU Memory Diag.
A1002AP	825/832/834/835/845/635/645	A1002A SPU Proc. Diag.
A1100AI	850/855/860/870	A1100A SPU I/O Diag.
A1100AM	850/855/860/870	A1100A SPU Memory Diag.
A1100AP	850/855/860	A1100A SPU Proc. Diag.
EDBC	890	Bus Converter Diag.
EDPROC	890	SPU Diag.
MPROC	870	Multiprocessor Diag.
MULTIDIAG	817/827/837/847/857/867/877	SPU Proc/Mem/IO Diag.
TDIAG	887/897	PCX-T Proc. Diag.
UNIPROC	842/852/865/870	Single Processor Diag.

Booting the HP-UX Support Tape

- To run the Support Tape's operating system from reset or transfer of control

```
Boot from primary boot path (Y or N)?> n
Boot from alternate path (Y or N)?> n
Enter boot path or ?> 4.2.3 (physical address of
Support Services Media device)
.
Booting
.
Console IO Dependent Code (IODC) revision 4
Boot IO Dependent Code (IODC) revision 4
.
Interact with IPL (Y or N)?> y
.
Hard Booted
.
ISL Revision A.00.02  June 22, 1989
ISL> hpux ():RECOVERY
```

Note Booting the Support Tape operating system may take SEVERAL minutes.

Support Tape Main Menu

- To get the Support Tape Main Menu from the HP-UX shell type:

  # menu

- Main Menu Screen:

```
                 SUPPORT TAPE MAIN MENU

     s.   Search for a file
     b.   Reboot
     l.   Load a file
     d.   On-line Diagnostics
     m.   Display Manual page for a specific command
     r.   Recover an unbootable HP-UX system
     u.   Utilities
     x.   Exit to shell

This menu is for listing and loading the tools
contained on the support tape.  Once a tool is
loaded, it may be run from the shell.

Select one of the above: _
```

Loading a File from Support Tape

- To load a file from the Support Tape Main Menu use the *l* option, as shown below:

```
                 SUPPORT TAPE MAIN MENU

    s.   Search for a file
    b.   Reboot
    l.   Load a file
    d.   On-line Diagnostics
    m.   Display Manual page for a specific command
    r.   Recover an unbootable HP-UX system
    u.   Utilities
    x.   Exit to shell

This menu is for listing and loading the tools
contained on the support tape.  Once a tool is
loaded, it may be run from the shell.

Select one of the above:   1

Filesystem   kbytes  used  avail capacity  Mounted on
/dev/fs        2464  1937    527   79%

Enter the name of the files that you want to load
separated by spaces:   newfs  disktab
```

- Once the file(s) has been loaded, exit to the shell for use.

Support Tape Utilities Menu

- If you select *u* from the SUPPORT TAPE MAIN MENU, the following menu will be displayed:

```
                SUPPORT TAPE UTILITIES MENU

        p.   Try to resynchronize position on tape
        t.   Table of contents of a tape section
        r.   Return to previous Menu
        x.   Exit to the shell

  Select one of the above: _
```

Support Tape On-line Diagnostics Menu

- If you select *d* from the SUPPORT TAPE MAIN MENU, the following menu will be displayed:

```
            ON LINE DIAGNOSTICS MENU

        0.   CS80DIAG : CS80 disks
        1.   FLEXDIAG : FLEX disks
        2.   SCSIDISK : SCSI disks
        3.   DTDUTIL  : Disk-Tape-Disk UTILity

        r.   Return previous menu
        x.   Break to the shell

  Select one of the above: _
```

HPUX Recovery - Main Menu

Support Tape HP-UX Recovery Main Menu

- If you select *r* from the SUPPORT TAPE MAIN MENU, the following menu will be displayed:

```
HP-UX     RECOVERY MAIN MENU

   Select one of the following:
   a.  Rebuild the bootlif (e.g., ISL) and
       install all 'critical' files required
       to boot HP-UX on a customer's root file
       system.
   b.  Do not rebuild the bootlif (e.g., ISL) but
       install 'critical' files required to boot
       HP-UX on a customer's root file system.
   c.  Rebuild only the bootlif (e.g., ISL).

   m.  Return to the 'Support Tape Main Menu'.
   x.  Exit to shell

Use this menu to select the level of recovery
desired.

Selection: _
```

PA-RISC System Exerciser (SX)

System Exerciser (SX) is a suite of system exercisers designed to stress test PA RISC systems to isolate intermittent failures. SX can stress any or all of the following subsystems for a specified number of passes or length of time:

- CPU
- Memory
- Disks
- Network

SX is a separate diagnostic product which is available to HP support personnel only.

Running SX

- SX in invoked by running *SX*.

  \# *SX*

```
**********************************************************
******        PA-RISC System Exerciser             ******
******                                             ******
******   (C) Copyright Hewlett-Packard Co.         ******
******          All Rights Reserved                ******
******                                             ******
****** This program may only be used by HP         ******
****** support personnel and those customers       ******
****** with the appropriate Class license or       ******
****** Node license for systems specified by       ******
****** the license.  HP shall not be liable        ******
****** for any damages resulting from misuse       ******
****** or unauthorized use of this program.        ******
****** This program remains the property of        ******
****** HP.                                         ******
******                                             ******
******          Version: B.01.07                   ******
******                                             ******
**********************************************************

Processing sxparm ..............................
System problems sometimes cause data corruption;
system stressing tools such as the System Exerciser
may accentuate these problems.  Therefore, it is
recommended that precautions be taken to protect the
integrity of customer data before using this or any
diagnostic tool.

Lock is:   739503
Enter key:
```

Help

To get help if you are in the System Exerciser, press the Help Key (F7). The help section is context-sensitive, so you will be shown information relating to the current System Exerciser screen.

To explore the on-line help section, use the cursor movement keys (Arrow keys, Home key, Page Up, and Page Down keys).

4

HPUX 8.0 Diskless Clusters

Creating a Diskless Cluster

To create a cluster server and add clients, perform the following:

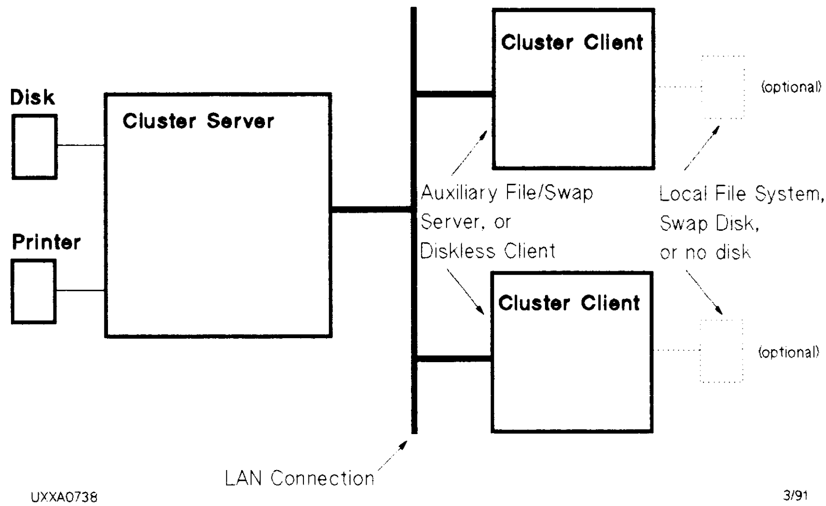
Note Diskless clusters are NOT supported on HP-UX version 9.0.

1. Use the System Administration Manager program "Create an HP-UX Cluster." This will "clusterize" a standalone system making it a cluster server.

Note During this step, you may add clients that are of the same type (S300 or S800) as your server.

2. Where a Series 800 machine is to act as a server for Series 3XX clients, run */etc/update* to update the server, and load Series 3XX software.
3. Use the System Administration Manager program "Add Cluster Clients."

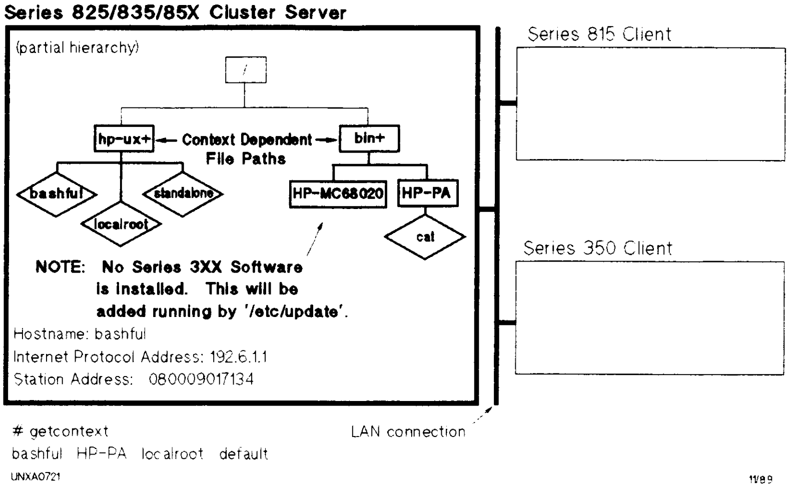
Diskless Basics



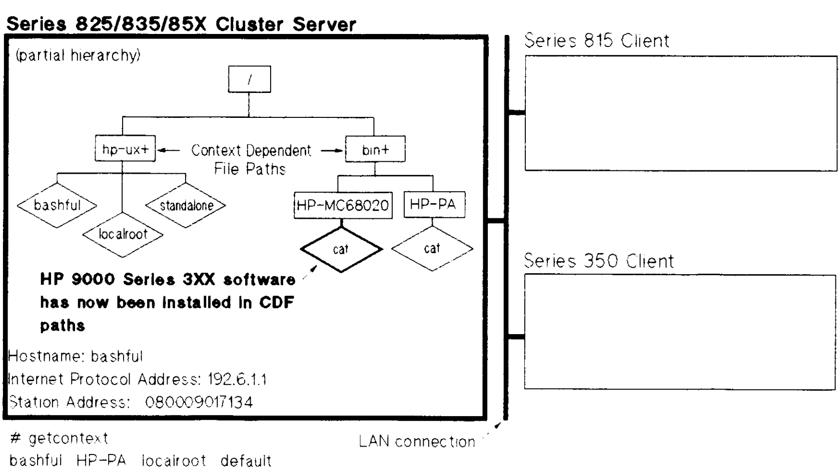
Servers: Series 600 and 800 (except Models 808 and 815) can serve all types of clients. Series 3XX and 4XX can only serve 3/4XX machines.

Clients: Series 800 Models 815, 822, 832, 842, 852, Series 3XX, and 4XX. No Series 800 machine may be served by a Series 3XX or 4XX machine.

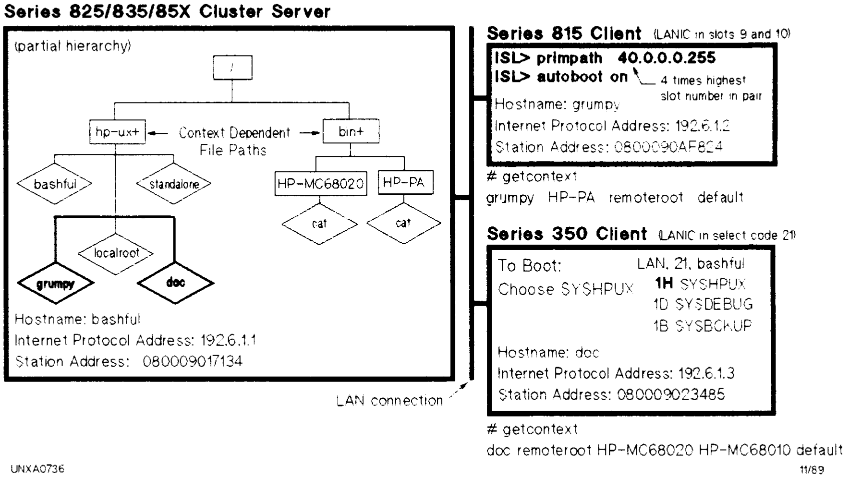
Newly Clusterized Server



After Updating HP-UX Series 3XX Software



New Cluster Clients Added



SAM—Create a Cluster

Cluster Configuration

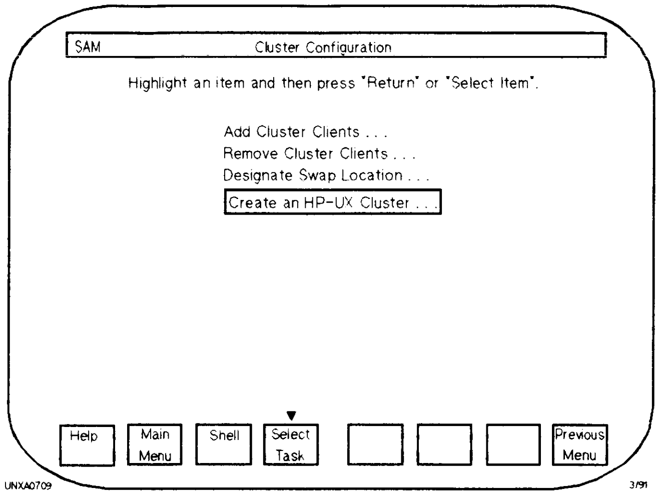
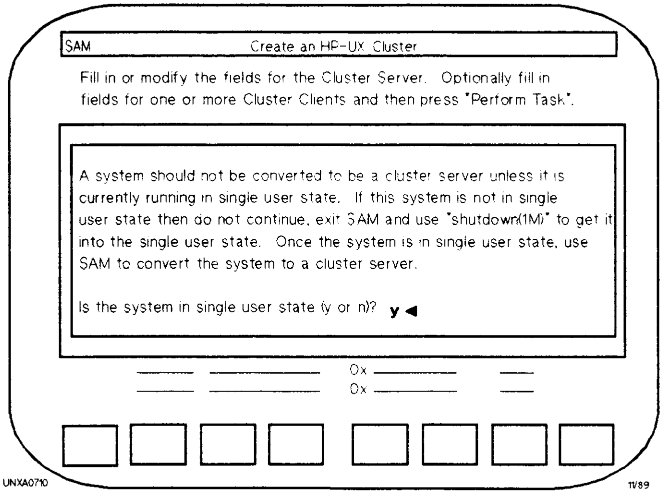
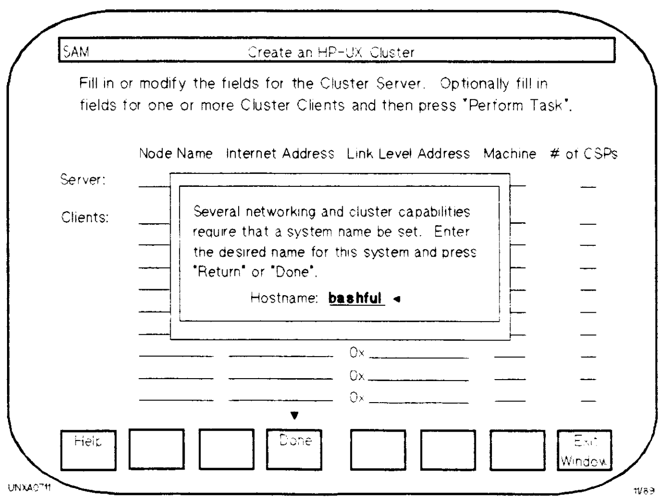


Figure 4-1. SAM: *Cluster Configuration*

Create an HP-UX Cluster

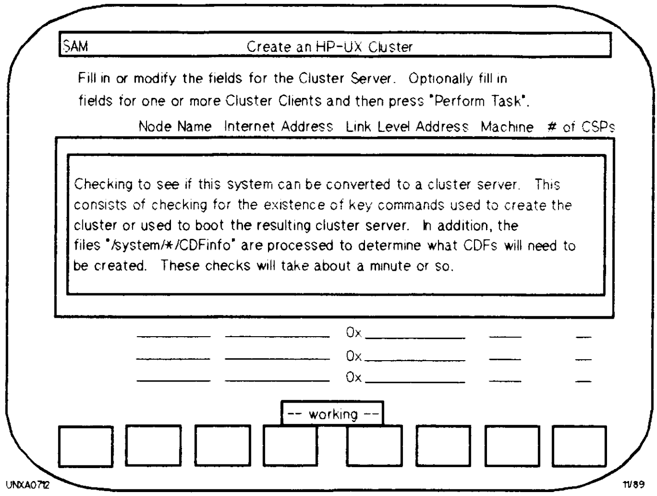


SAM: *Create an HP-UX Cluster*, **Slide 1 of 11**

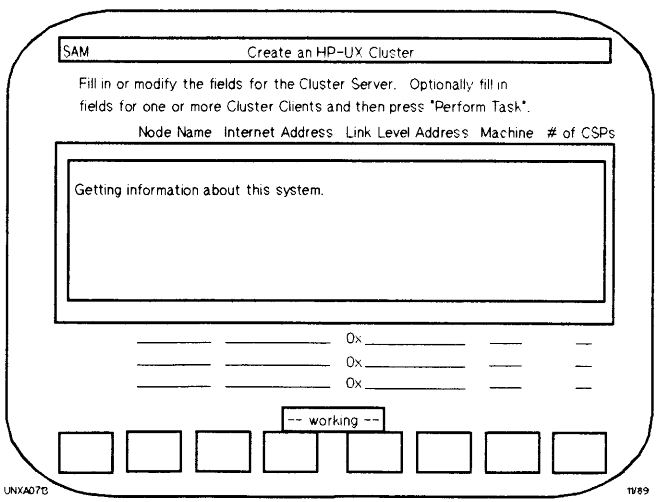


SAM: *Create an HP-UX Cluster*, **Slide 2 of 11**

Create an HP-UX Cluster, Cont.

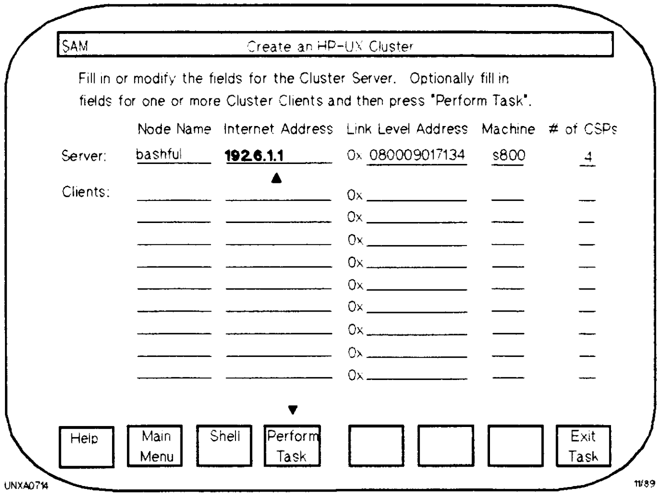


SAM: *Create an HP-UX Cluster*, **Slide 3 of 11**

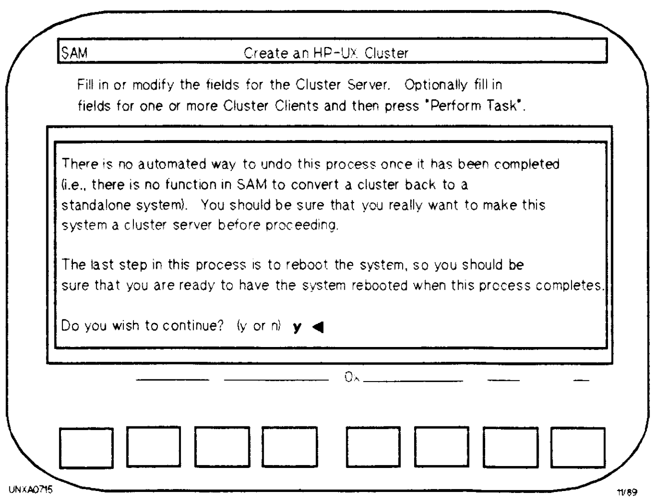


SAM: *Create an HP-UX Cluster*, **Slide 4 of 11**

Create an HP-UX Cluster, Cont.

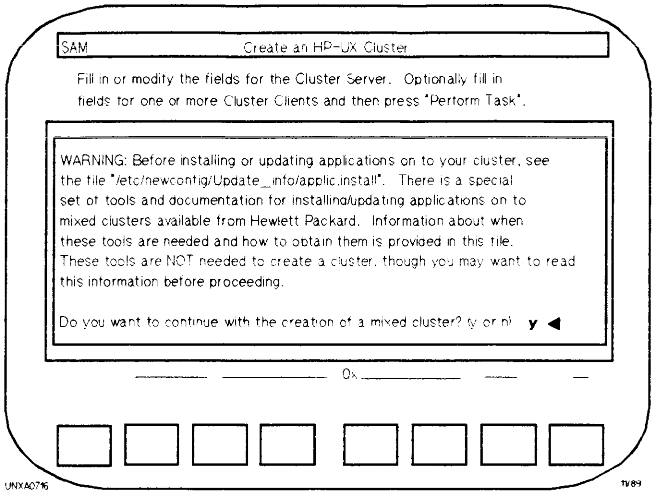


SAM: *Create an HP-UX Cluster*, **Slide 5 of 11**

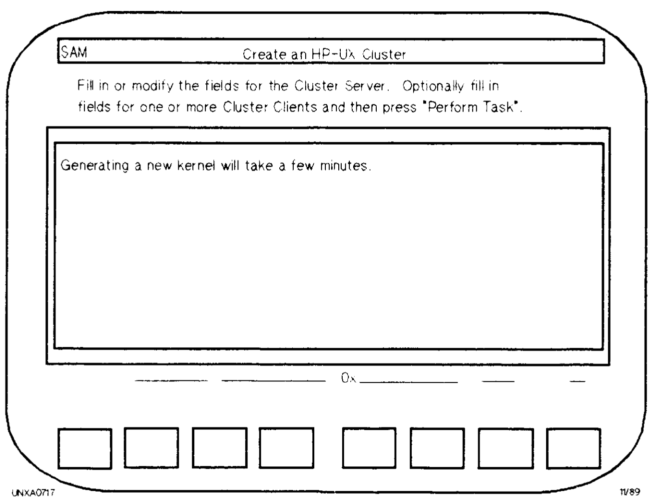


SAM: *Create an HP-UX Cluster*, **Slide 6 of 11**

Create an HP-UX Cluster, Cont.

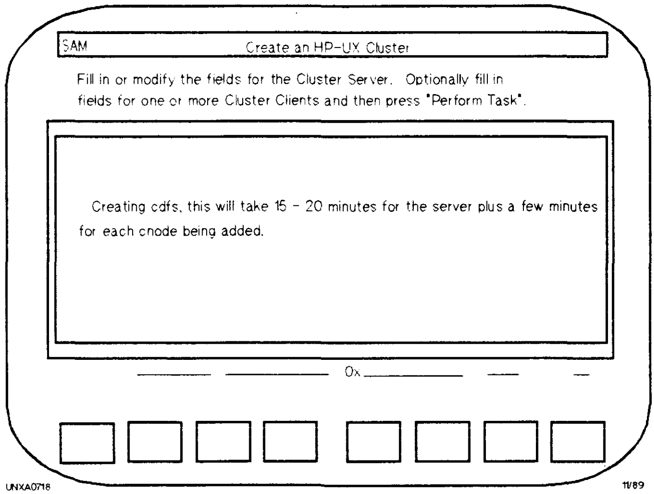


SAM: *Create an HP-UX Cluster*, **Slide 7 of 11**

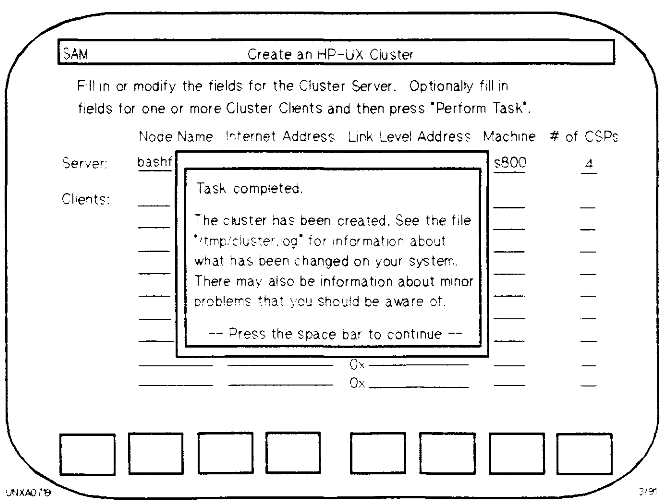


SAM: *Create an HP-UX Cluster*, **Slide 8 of 11**

Create an HP-UX Cluster, Cont.

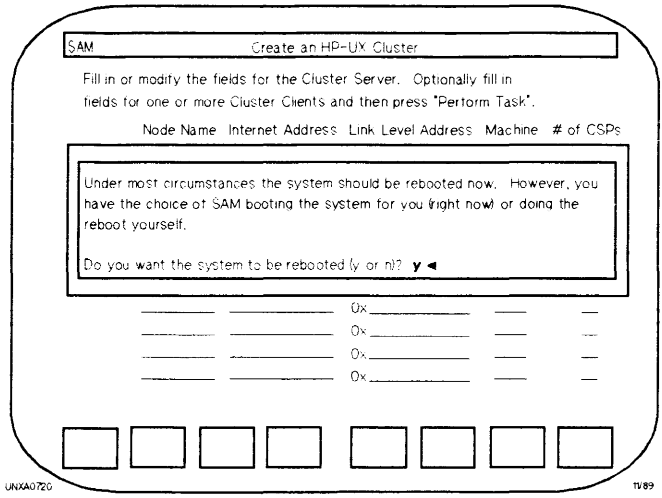


SAM: *Create an HP-UX Cluster*, **Slide 9 of 11**



SAM: *Create an HP-UX Cluster*, **Slide 10 of 11**

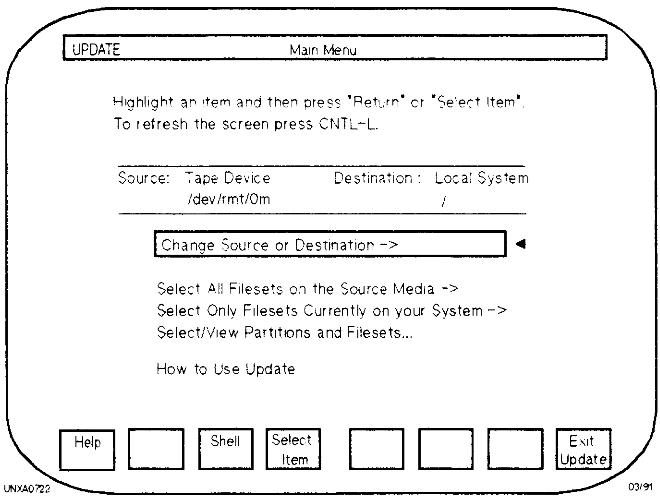
Create an HP-UX Cluster, Cont.



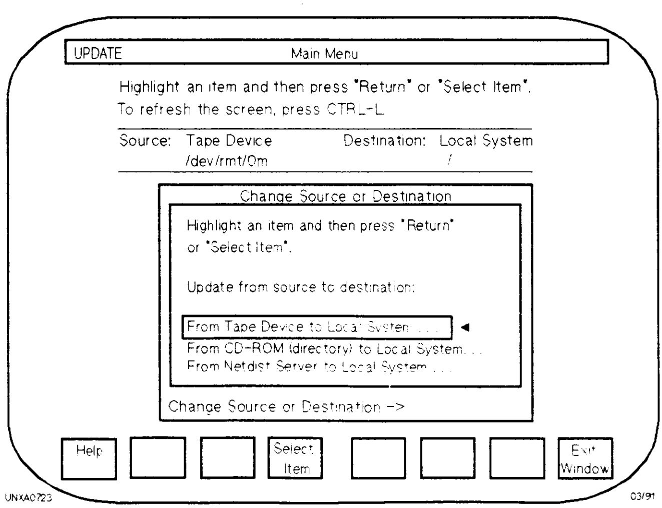
SAM: *Create an HP-UX Cluster*, **Slide 11 of 11**

Update

Main Menu

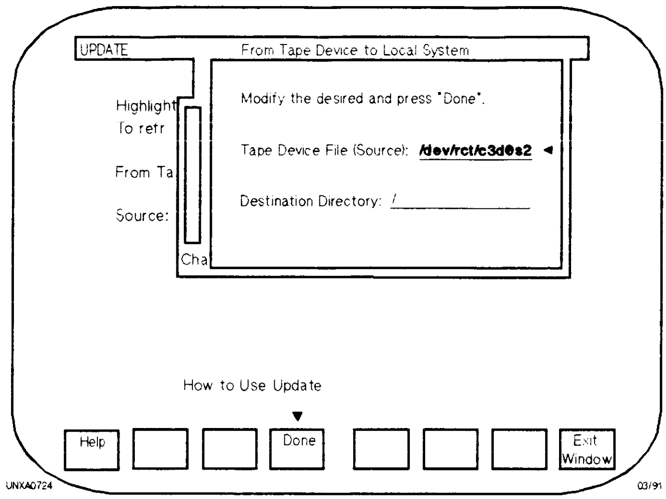


SAM: *Update*, **Slide 1 of 5**



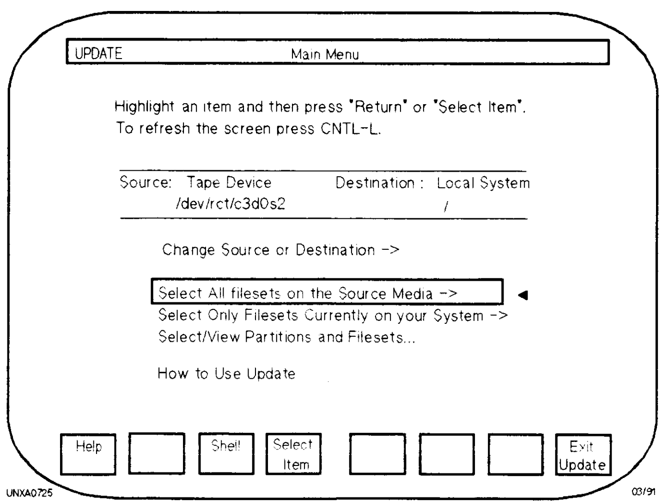
SAM: *Update*, **Slide 2 of 5**

From Tape Device to Local System



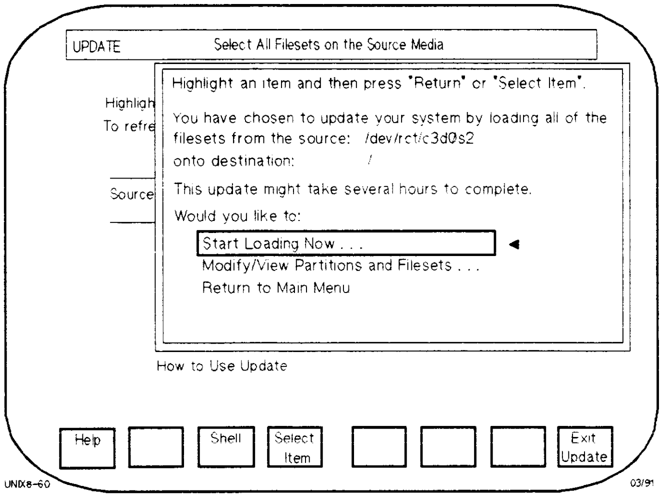
SAM: *Update*, **Slide 3 of 5**

Main Menu



SAM: *Update*, **Slide 4 of 5**

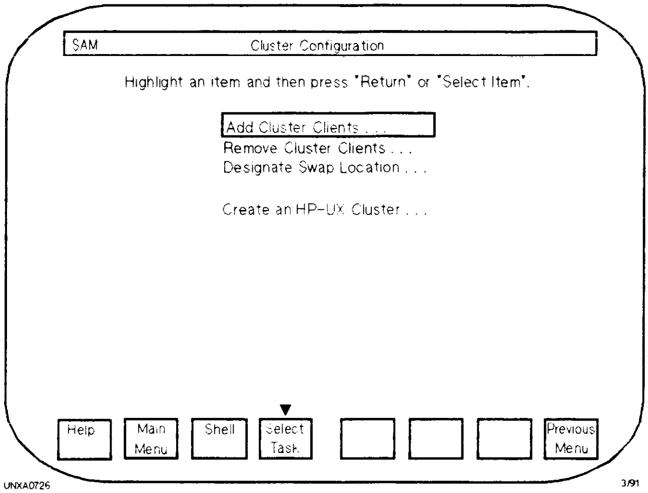
Select all Filesets on the Source Media



SAM: *Update*, **Slide 5 of 5**

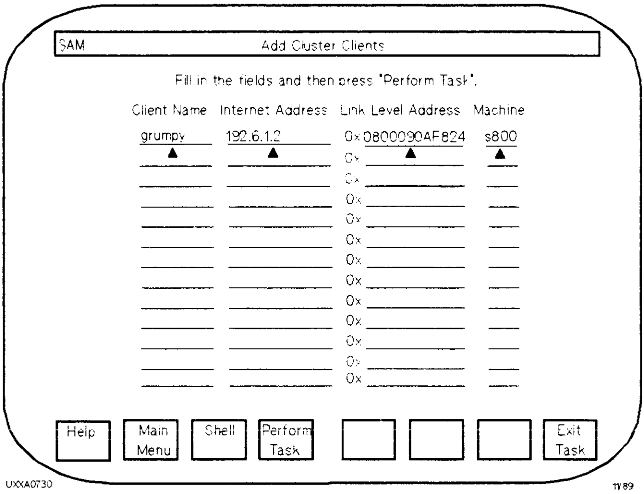
SAM—Cluster Clients

Cluster Configuration

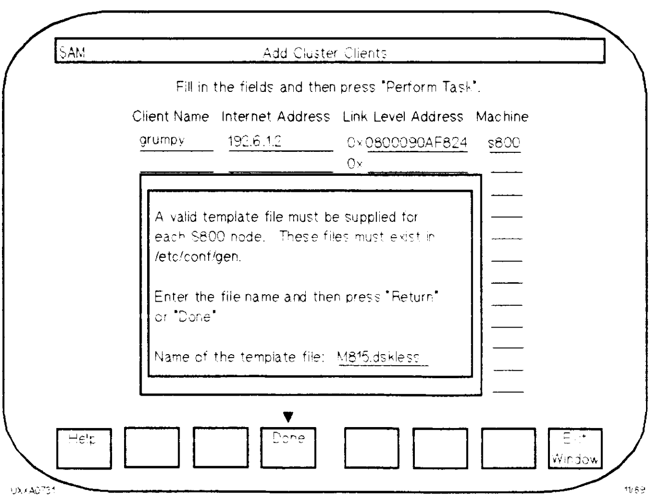


SAM: *Cluster Configuration*

Add Cluster Clients

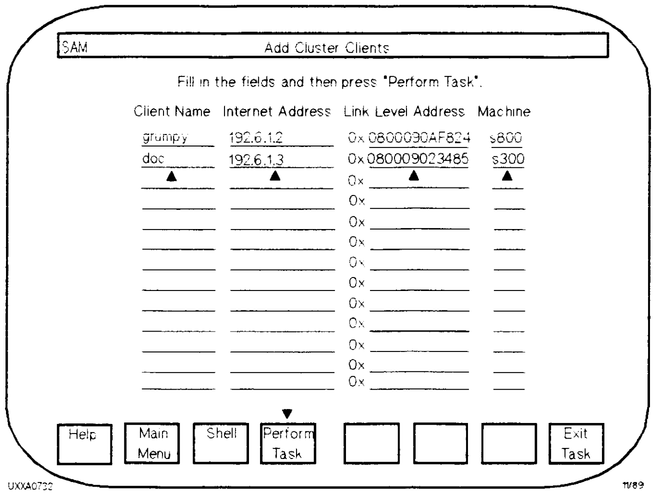


SAM: *Add Cluster Clients*, **Slide 1 of 7**

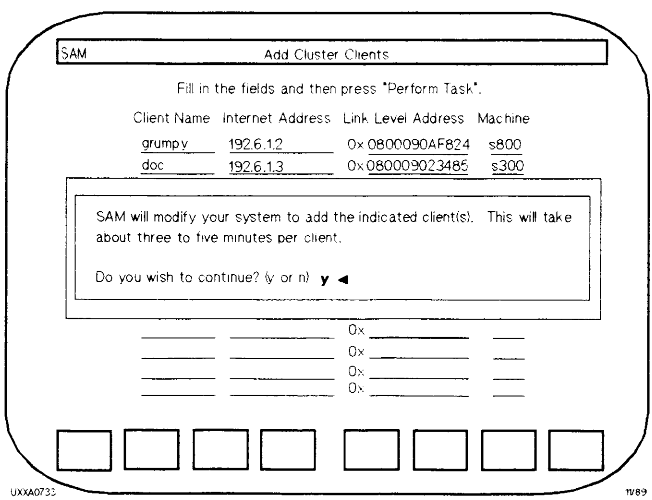


SAM: *Add Cluster Clients*, **Slide 2 of 7**

Add Cluster Clients, Cont.

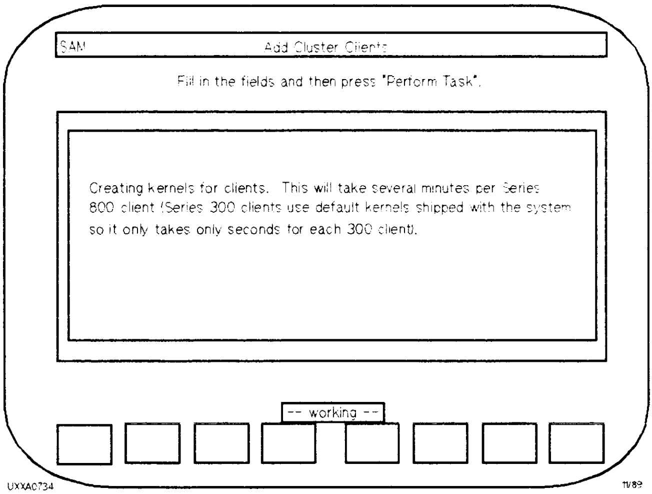


SAM: *Add Cluster Clients*, **Slide 3 of 7**

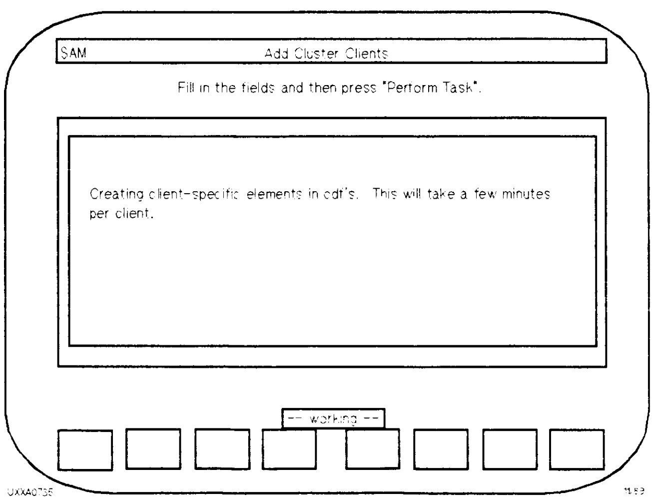


SAM: *Add Cluster Clients*, **Slide 4 of 7**

Add Cluster Clients, Cont.

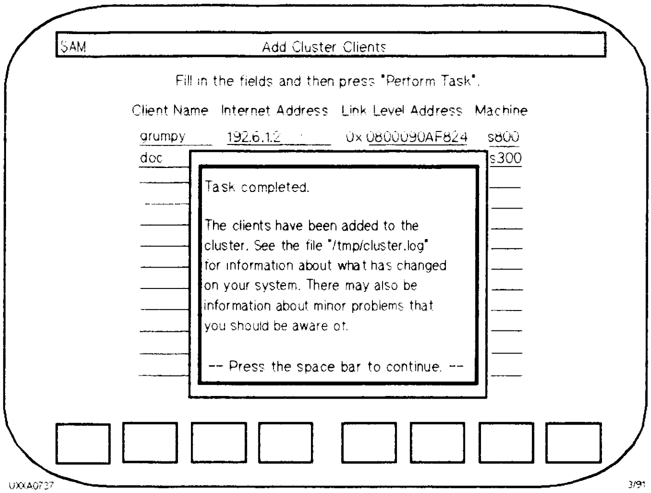


SAM: *Add Cluster Clients*, **Slide 5 of 7**



SAM: *Add Cluster Clients*, **Slide 6 of 7**

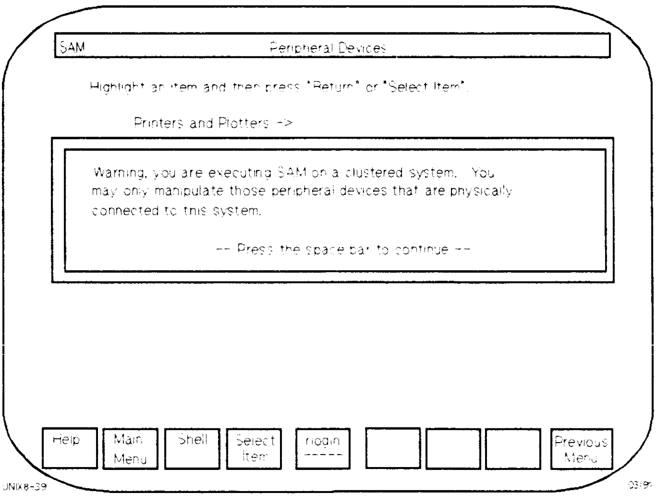
Add Cluster Clients, Cont.



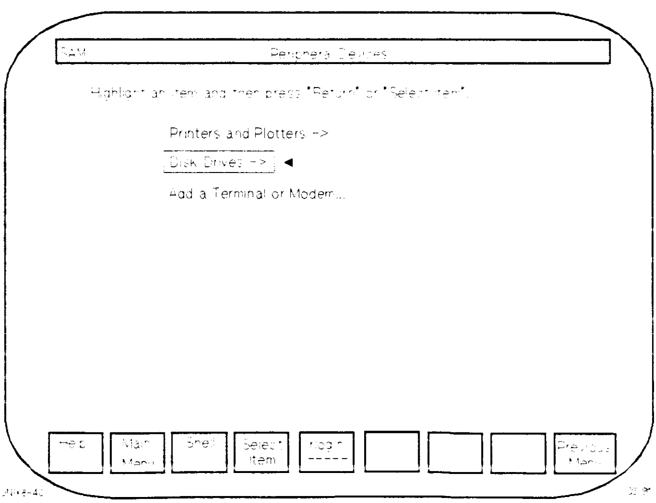
SAM: *Add Cluster Clients*, **Slide 7 of 7**

SAM—Auxiliary File and Swap Server Configuration

Peripherals Devices

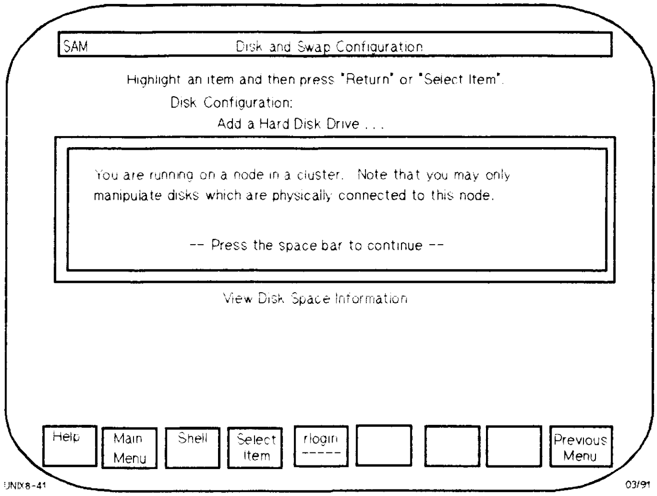


SAM: *Peripheral Devices*, **Slide 1 of 12**

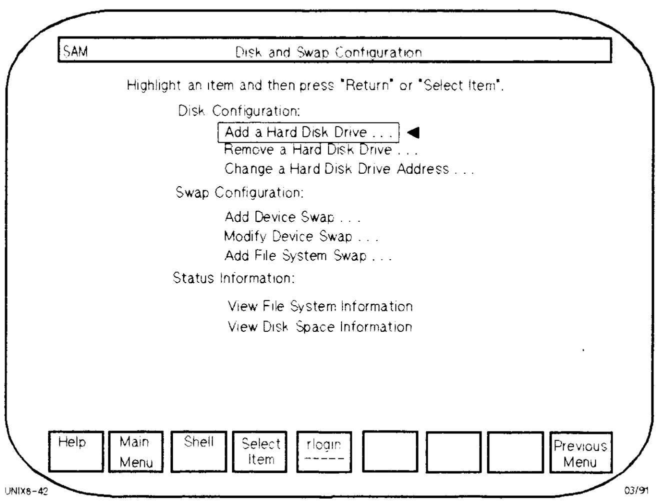


SAM: *Peripheral Devices*, **Slide 2 of 12**

Disk and Swap Configuration

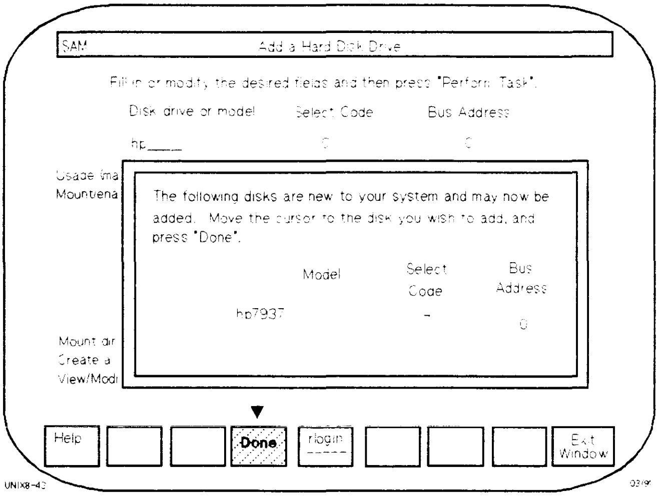


SAM: *Peripheral Devices*, **Slide 3 of 12**

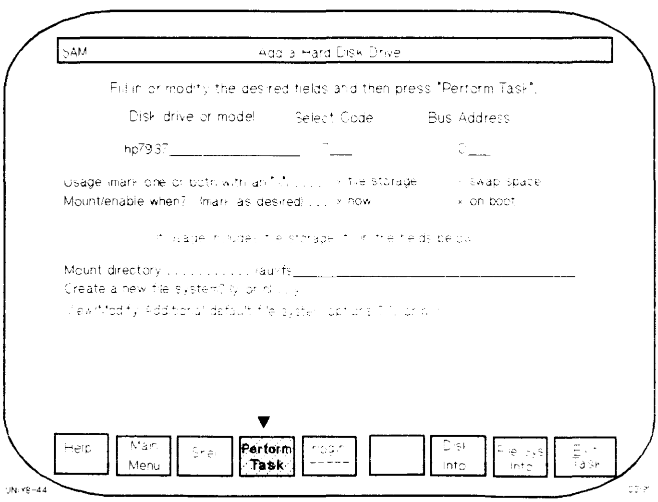


SAM: *Peripheral Devices*, **Slide 4 of 12**

Add a Hard Disk Drive

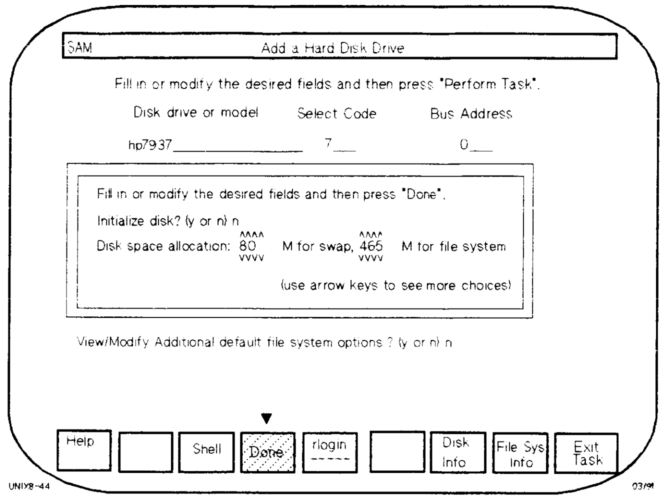


SAM: *Peripheral Devices*, **Slide 5 of 12**

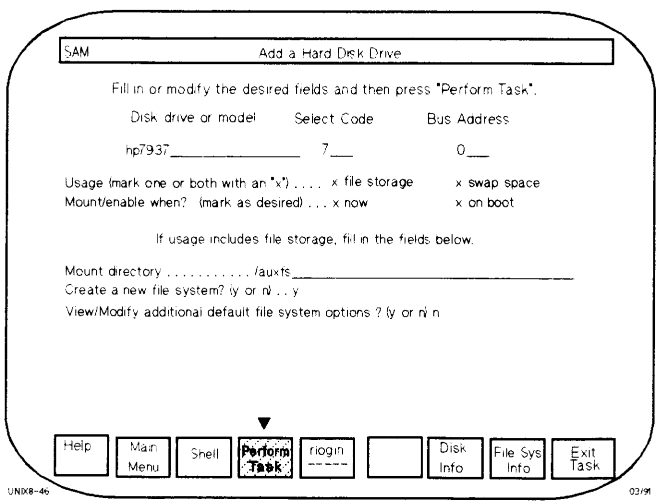


SAM: *Peripheral Devices*, **Slide 6 of 12**

Add a Hard Disk Drive, Cont.

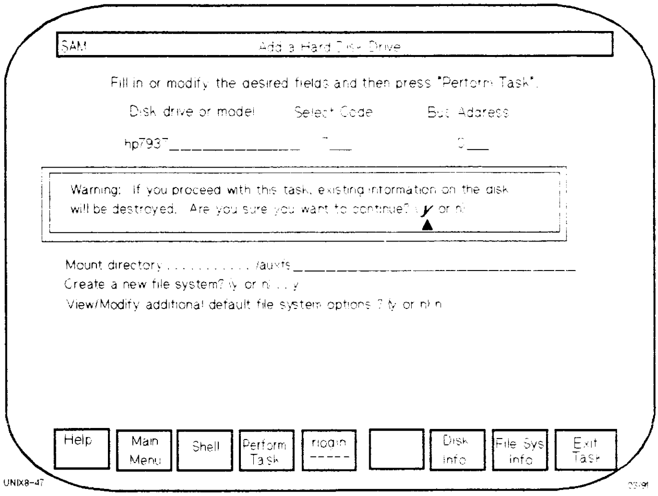


SAM: *Peripheral Devices*, **Slide 7 of 12**

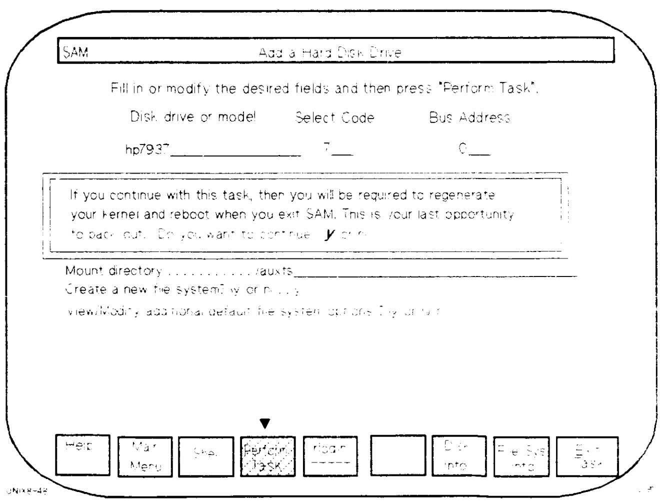


SAM: *Peripheral Devices*, **Slide 8 of 12**

Add a Hard Disk Drive, Cont.

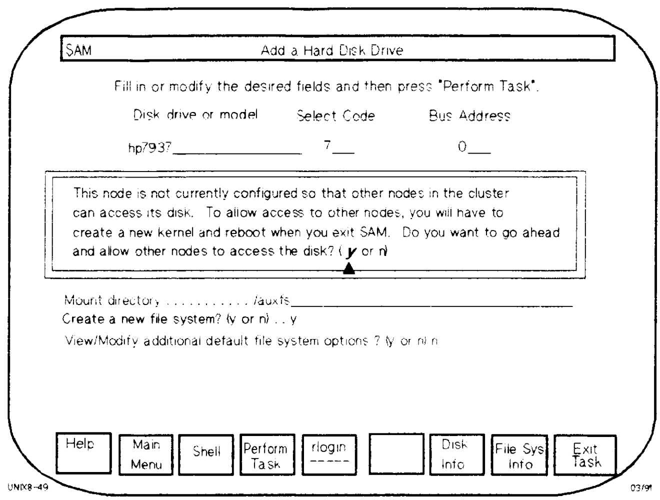


SAM: *Peripheral Devices*, **Slide 9 of 12**

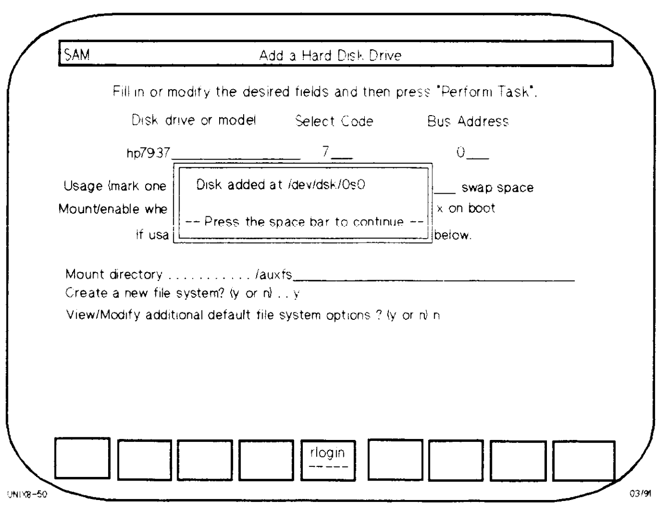


SAM: *Peripheral Devices*, **Slide 10 of 12**

Add a Hard Disk Drive, Cont.

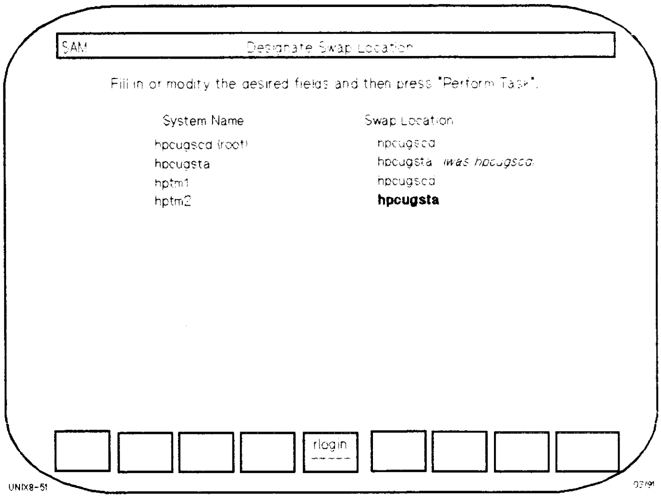


SAM: *Peripheral Devices*, **Slide 11 of 12**

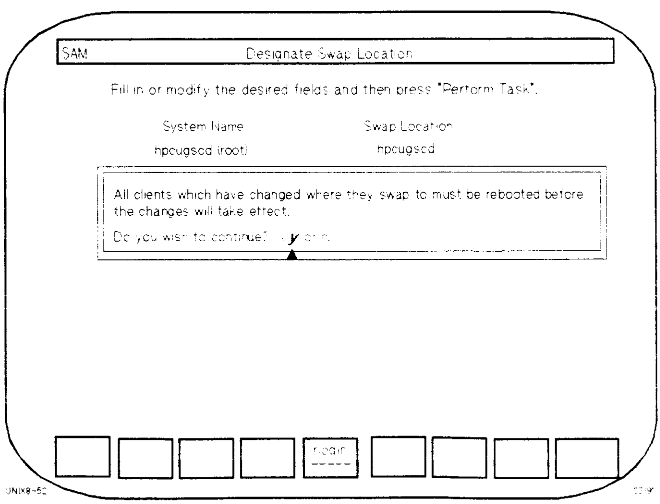


SAM: *Peripheral Devices*, **Slide 12 of 12**

Designate Swap Location

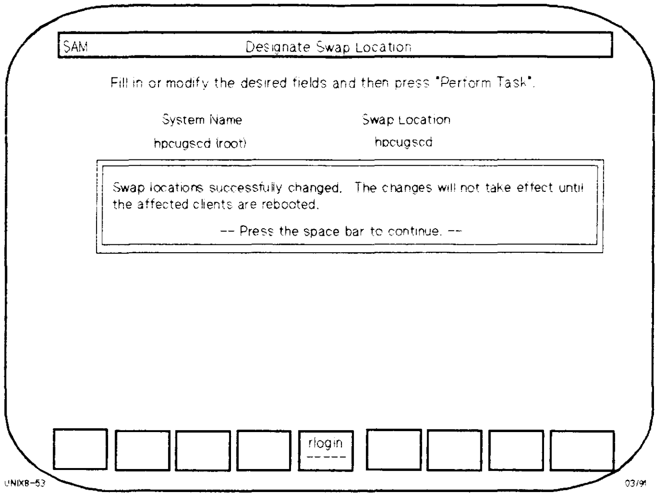


SAM: *Designate Swap Location*, **Slide 1 of 4**



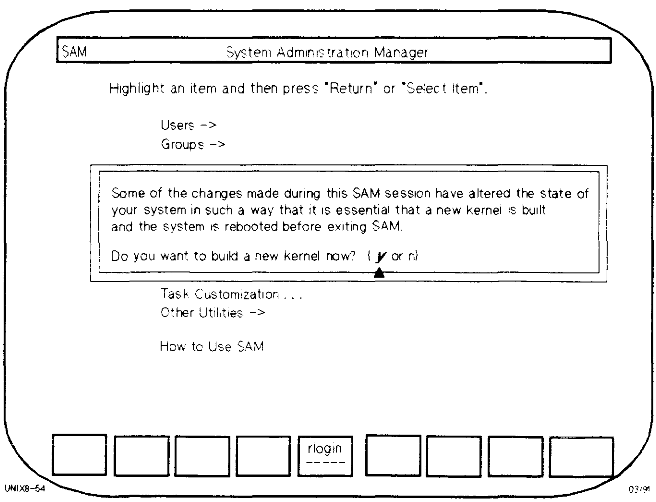
SAM: *Designate Swap Location*, **Slide 2 of 4**

Designate Swap Location, Cont.



SAM: *Designate Swap Location*, **Slide 3 of 4**

System Administration Manager



SAM: *Designate Swap Location*, **Slide 4 of 4**

5

DataPair/800

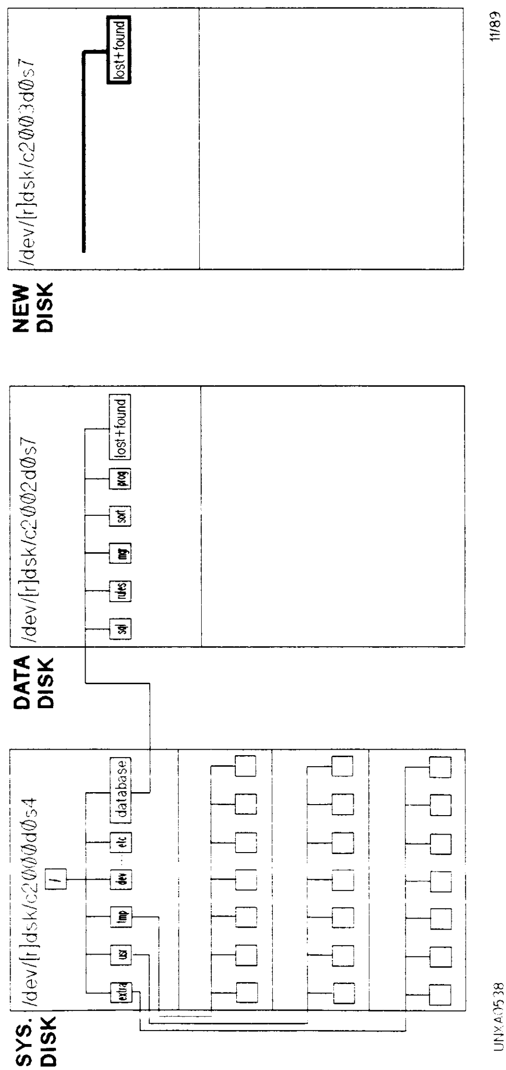
Mirror Disk Basics

- A mirrored disk is a pair of disk sections that are copies of each other. To the user they look and act like a single section.
- The pair is managed by kernel code called the *mirror driver*. A physical write is split into two write calls, one for each section of the pair. A physical read is routed to the least busy drive.
- Mirror disks are supported only on HP-FL drives on HP 9000 Series 8XX machines (except 808 and 815).
- Section pairs must be on identical disk model numbers, and must have identical section numbers. Pair section internal structures (block size, fragment size, etc.) must be identical.
- The mirror driver allows a single HP 9000 Series 800 CPU to create and share up to 16 mirror disk pairs (32 drives).
- It is recommended (but not required) that each member of a pair be installed on separate HP-FL interfaces to prevent a single point interface failure from affecting both members of a pair.

Creating a Mirror Disk

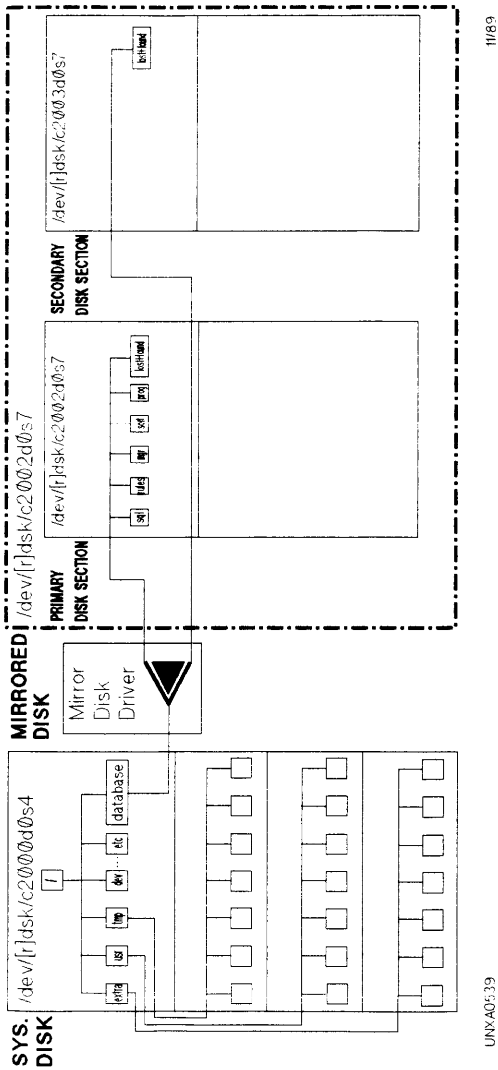
1. Create a new file system using the *newfs* command. For example:

   # newfs /dev/rdsk/c2003d0s7 hp7937



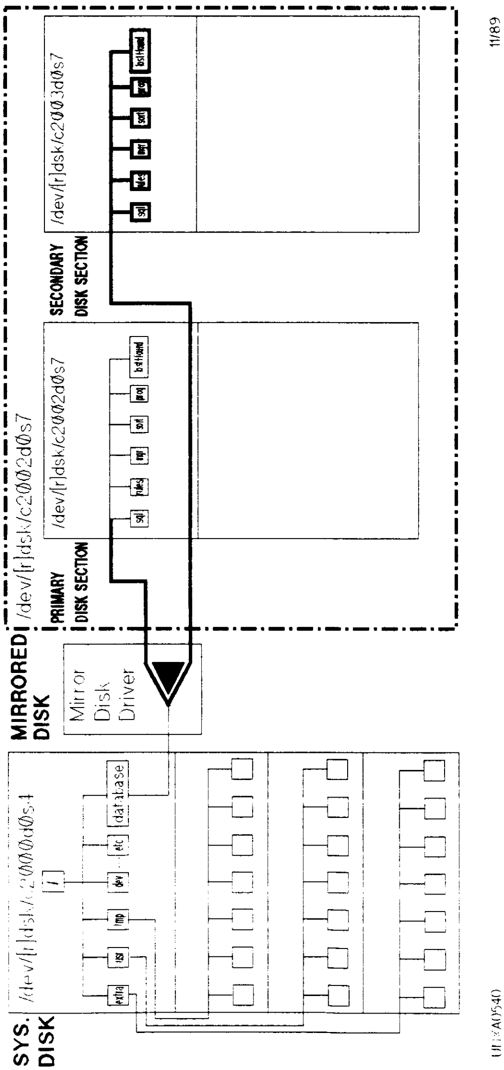
2. Mirror the primary and secondary disk sections. For example:

   # mirror -c /dev/rdsk/c2002d0s7 online /dev/rdsk/c2003d0s7 offline



3. Update the secondary disk section with an exact image of the primary disk section using the *mirror* command. For example:

   # mirror -r /dev/dsk/c2002d0s7



/etc/mirror

- The *mirror* command is used to configure, unconfigure, and control mirrored disks. The *mirror* command and options are shown below:

  # *mirror -c [-f] primarydev pstate secondarydev sstate*

  # *mirror -u mirrordev ...*

  # *mirror -o [-f] -p|-s mirrordev [[-p|-s] mirrordev ...]*

  # *mirror -r [-t] mirrordev*

  # *mirror -l [device]*

Where:

Key	**Operation**
-c	Configure a mirror disk (does not work for *root* and *swap*; uxgen required).
-f	Manually set fail flag for the OFFLINE section (when set, indicates hardware failure).
primarydev	Block or character special file path of the primary disk section of the mirror.
pstate	State of primary disk section. Set *pstate* to *online* or *offline* (typically set to *online*).
secondarydev	Block or character special file path of the secondary disk section of the mirror.
sstate	State of secondary disk section. Set *sstate* to *online* or *offline* (typically set to *online*).
-u	Unconfigure the named mirror(s); revert mirrored section(s) to unmirrored.
mirrordev	Block or character special file path of primary disk section in mirror.
-p|-s	*-p* requests primary disk section go offline; *-s* requests secondary disk go offline.
-r	Reimage the named mirror (assures identical data on both sections of mirror).
-t	Requests that a table-driven reimage (update only changed files since going offline).
-l	List mirrors (uses */etc/mirrorlog* daemon and */etc/mirrortab*).
device	Special file path of desired mirror disk.

Listing Mirror Disk Status

- To list mirror status, type:

  # /etc/mirror -l

- *mirror -l* displays one line per mirror in the following format:

  primarydev pstate secondarydev sstate fail

  Where:

Key	Operation
primarydev	Block special file path of primary disk section.
pstate	*ONLINE*—Disk section is available for reads and writes. *OFFLINE*—Disk section is not available for reads and writes. *REIMAGE*—Disk section is being re-imaged.
secondarydev	Block special file path of secondary disk section.
sstate	Same as *pstate* above except indicates secondary disk section state.
fail	*FAIL*—Hardware fail flag; set automatically on failure or by *mirror -cf* command. *GOOD*—Hardware status flag displayed if *FAIL* flag not set.

- Example:

  # /etc/mirror -l

  /dev/dsk/c0d0s4 ONLINE /dev/dsk/c1d0s4 OFFLINE GOOD
  /dev/dsk/c0d0s10 ONLINE /dev/dsk/c1d0s10 REIMAGE GOOD

Setting Mirror Disk Sections Offline

- Only one section of a mirror disk may be taken offline at a time; the other must remain online.
- Use the *mirror -o* command to take a section offline (see */etc/mirror* command earlier in this section).
- Example:

  \# */etc/mirror -l*

  /dev/dsk/c0d0s10 ONLINE /dev/dsk/c1d0s10 ONLINE GOOD

  \# */etc/mirror -os /dev/dsk/c0d0s10* (set secondary *OFFLINE*)

  \# */etc/mirror -l*

  /dev/dsk/c0d0s10 ONLINE /dev/dsk/c1d0s10 OFFLINE GOOD

  \# */etc/mirror -r /dev/dsk/c0d0s10* (reimage, set *ONLINE*)

  \# */etc/mirror -l*

  /dev/dsk/c0d0s10 ONLINE /dev/dsk/c1d0s10 ONLINE GOOD

  \# */etc/mirror -op /dev/dsk/c0d0s10* (set primary *OFFLINE*)

  \# */etc/mirror -l*

  /dev/dsk/c0d0s10 OFFLINE /dev/dsk/c1d0s10 ONLINE GOOD

6

File Systems

Series 600/800 Conventional Disk Sectioning Scheme

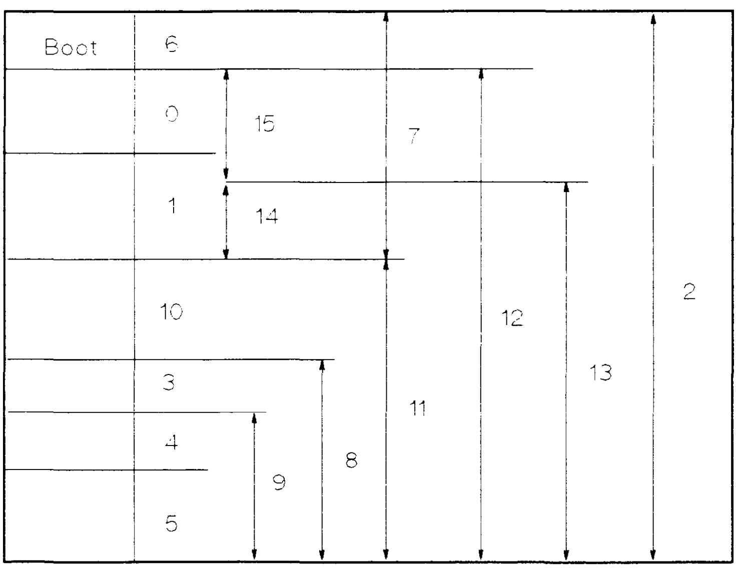


Figure 6-1. Disk Sectioning Scheme

Note Do not allocate file systems on sections that overlap.

Creating Conventional Series HP-UX File Systems

1. Initialize media using *mediainit*:

   \# *mediainit /dev/diag/dsk/c1d0*

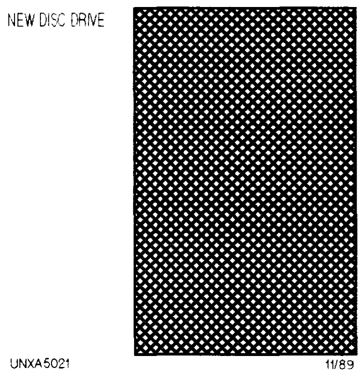


Figure 6-2. Initializing the Media

2. Make a new file system using *newfs*:

   \# *newfs /dev/rdsk/c1d0s7 hp7937*

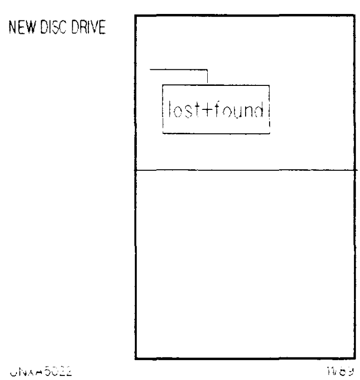


Figure 6-3. Making a New File System

3. Make a mount point directory using *mkdir*:

   # *mkdir /newdir*

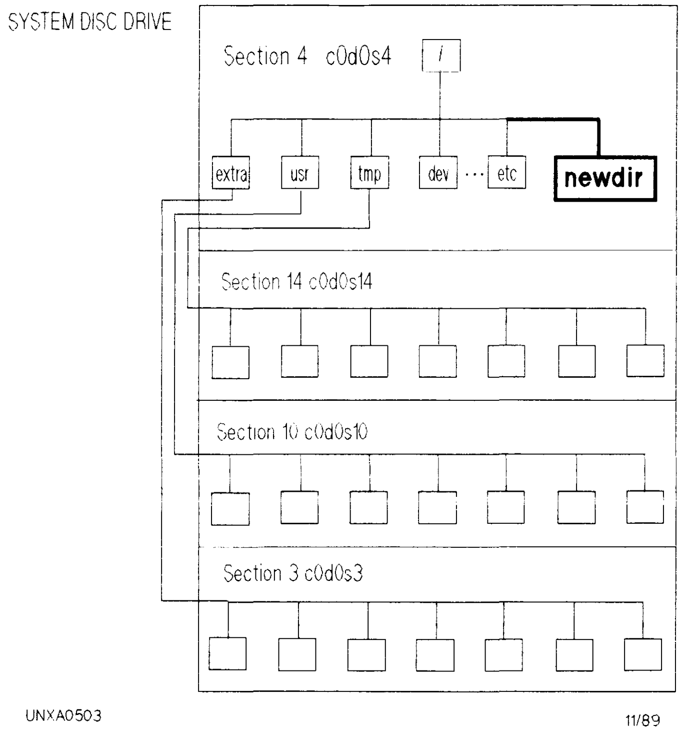


Figure 6-4. Adding a New Directory

4. Mount the new file system using *mount*:

   # mount /dev/dsk/c1d0s7 /newdir

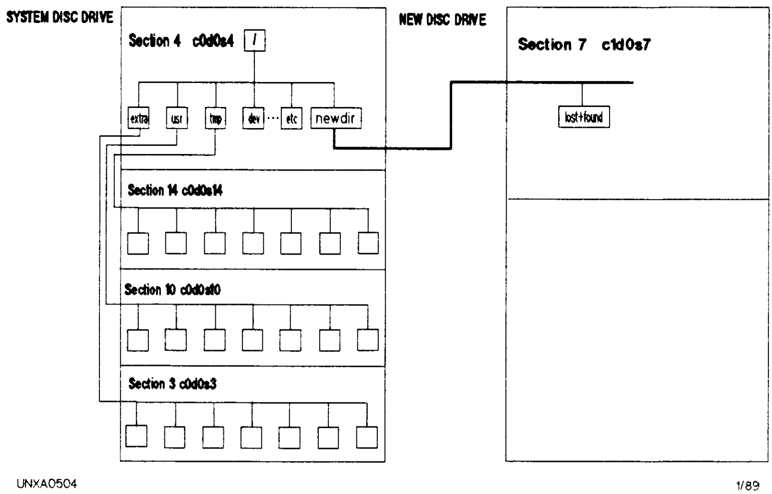


Figure 6-5. Mounting the New File System

Creating LVM File Systems

1. Initialize the media using *mediainit:*

   # mediainit /dev/diag/dsk/c1d0

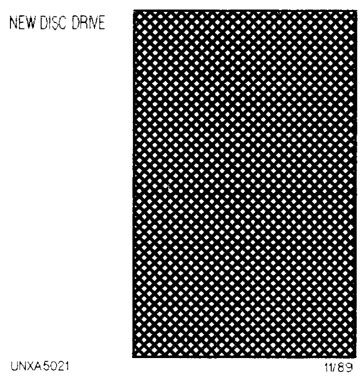


Figure 6-6. Initializing the Media

2. Create a physical volume for use as a volume group:

   # pvcreate -f /dev/rdsk/c1d0s2

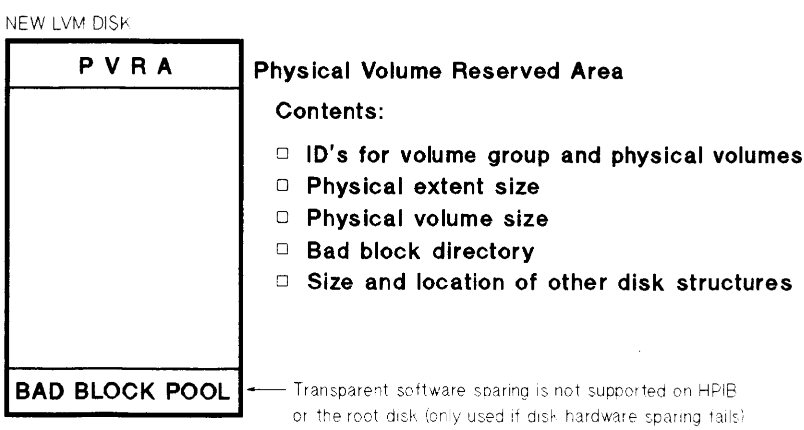


Figure 6-7. Creating a Physical Volume

3. Make a directory under */dev* for the volume group:

 # mkdir /dev/vg01

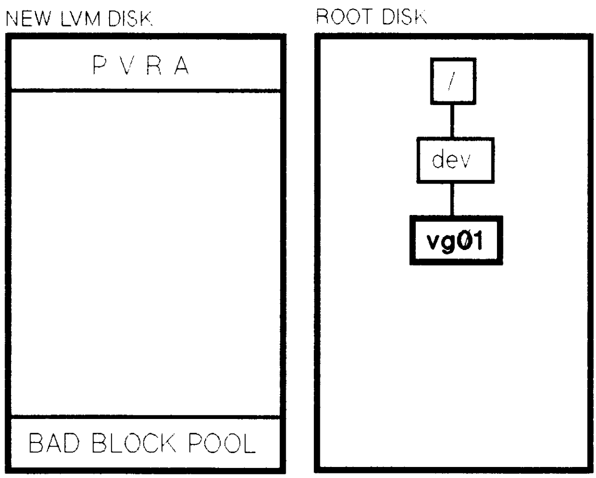


Figure 6-8. Creating a */dev* **Subdirectory for the Volume Group**

4. Make a group device file with *mknod*:

 # mknod /dev/vg01/group c 64 0x010000

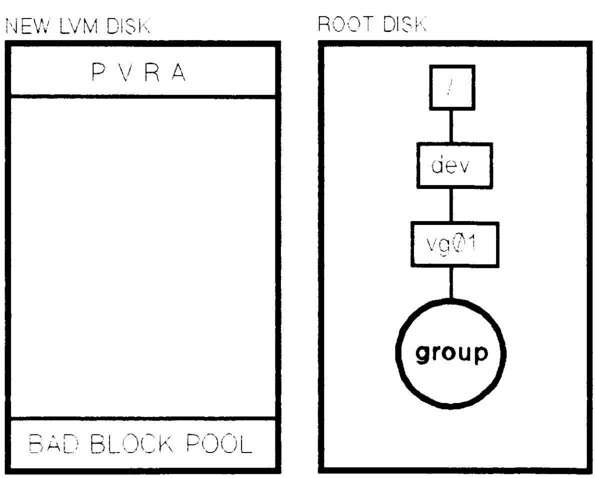
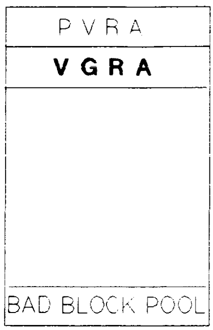


Figure 6-9. Creating a *group* **Device File**

5. Create a volume group:

 # vgcreate /dev/vg01 /dev/dsk/c1d0s2



Volume Group Reserved Area Contents

- **Volume Group Descriptor Area (VGDA)**
  Identifies logical and physical volumes
  Physical to logical extent mapping
- **Volume Group Status Area (VGSA)**
  Physical volume status (missing/present)
  Physical extent status (stale/ok)
- **Mirror Consistency Record (MCR)**
  Lists disk writes in progress

'vgcreate' creates or updates '/etc/lvmtab'

Adds volume group information to '/etc/lvmtab'

Figure 6-10. Creating a Volume Group

6. Create a 100Mb logical volume in volume group one:

 # lvcreate -L 100 vg01

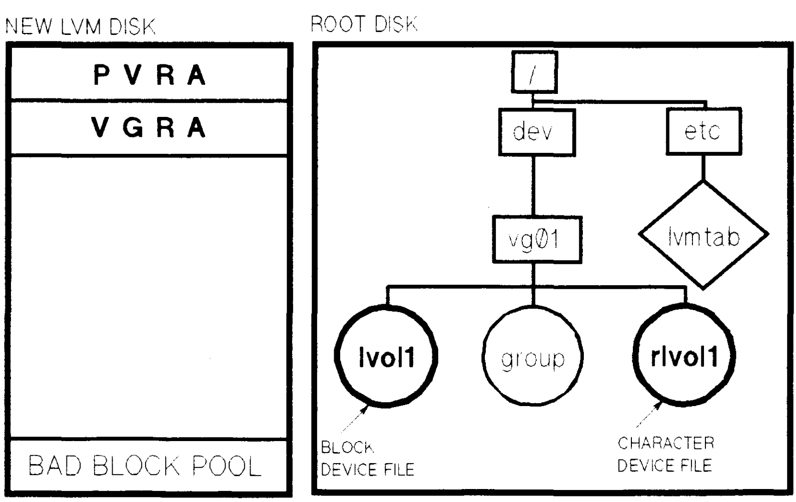


Figure 6-11. Creating a Logical Volume

7. Create a physical file system for logical volume one:

   # newfs /dev/vg01/rlvol1 hp7937

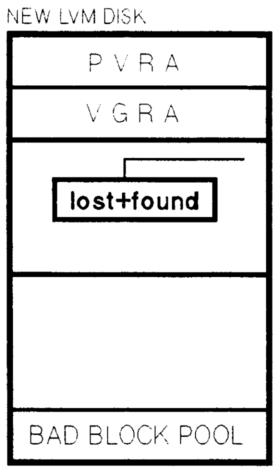


Figure 6-12. Creating a Physical File System

8. Make a mount point directory using *mkdir*:

   # mkdir /newdir

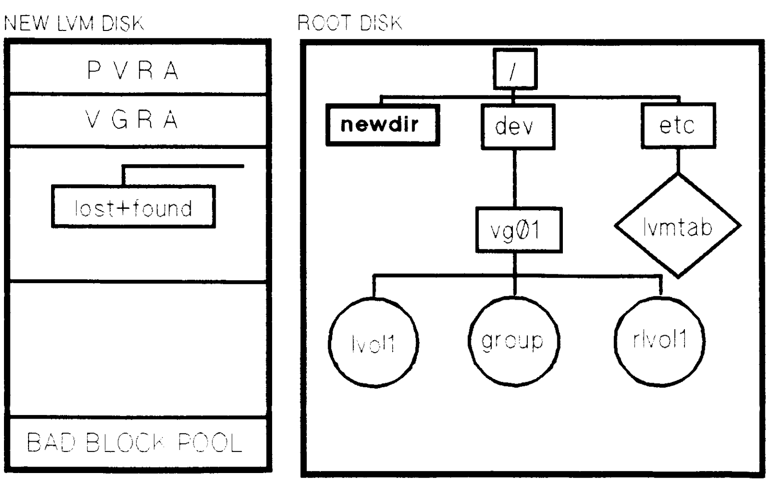


Figure 6-13. Creating a Mount Point Directory

9. Mount the new file system using *mount*:

   # mount /dev/vg01/lvol1 /newdir

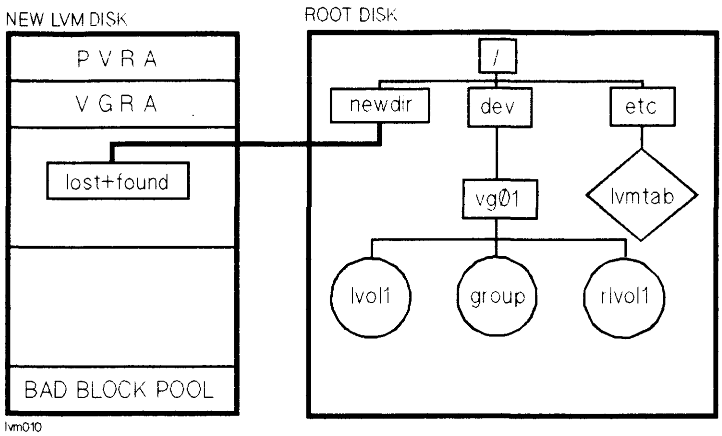


Figure 6-14. Mounting the New LVM File System

/etc/newfs

- Creates a file system using default values from */etc/disktab*.
- Calls *mkfs* to construct the file system:

  # /etc/newfs [-v] [mkfs-options] devicefile model

  Where:

Key	Operation
-v	Verbose mode.
mkfs-options	Any option available with *mkfs*; the options override the default values.
devicefile	Special device file associated with the device.
model	HP model number of the disk (see */etc/disktab*).

- Example (*newfs* with default values):

  # newfs /dev/rdsk/c1d0s3 hp7935

- Example (*newfs* specifying file system characteristics; overrides */etc/disktab*):

  # newfs -b 4096 -f 2048 -m 15 -i 4096 /dev/rdsk/c1d0s3 hp7935

  Where:

Key	Operation
-b 4096	Specifies block size in bytes.
-f 2048	Specifies fragment size in bytes.
-m 15	Specifies the minfree value in percent.
-i 4096	Specifies one inode for every 4096 bytes of file space (default: one inode for every 2048 bytes of file space).

The */etc/disktab* File

- *disktab* defines supported disk drive geometries and disk section characteristics.
- Used by */etc/newfs* to create HP-UX file systems.
- Options entered with the *newfs* command override parameters contained in */etc/disktab*.
- For each supported HP model number, *disktab* lists:

ty	Type of disk (removable or Winchester).
ns	Number of sectors per track.
nt	Number of tracks per cylinder.
nc	Total number of cylinders on the disk.
s[0-n]	Section size in sectors; file system size = DEV_BSIZE * the number of sectors (where DEV_BSIZE = 1024 bytes).
b[0-n]	Section block sizes in bytes.
f[0-n]	Section fragment sizes in bytes.
se	Number of bytes per sector.
rm	Number of revolutions per minute.

- Example (a sample entry for an HP7937 disk):

```
hp7937:\
        :ty=winchester:ns#30:nt#13:nc#1396:rm#3600:\
        :s0#24280:b0#8192:f0#1024:\
        :s1#48560:b1#8192:f1#1024:\
        :s2#558051:b2#8192:f2#1024:\
        :s3#29298:b3#8192:f3#1024:\
        :s4#107426:b4#8192:f4#1024:\
        :s5#216664:b5#8192:f5#1024:\
        :s6#1998:b6#8192:f6#1024:\
        :s7#75152:b7#8192:f7#1024:\
        :s8#353778:b8#8192:f8#1024:\
        :s9#324196:b9#8192:f9#1024:\
        :s10#129024:b10#8192:f10#1024:\
        :s11#482898:b11#8192:f11#1024:\
        :s12#556052:b12#8192:f12#1024:\
        :s13#507282:b13#8192:f13#1024:\
        :s14#24280:b14#8192:f14#1024:\
        :s15#48560:b15#8192:f15#1024:\
```

A Conventional File System */etc/checklist* Example

 # cat /etc/checklist

/dev/dsk/c0d0s4	/	hfs	rw	0[1]	1[2]	# root
/dev/dsk/c0d0s3	/extra	hfs	rw	0	2	# extra
/dev/dsk/c0d0s5	/mnt	hfs	rw	0	3	# /mnt
/dev/dsk/c0d0s0	/tmp	hfs	rw	0	4	# /tmp
/dev/dsk/c0d0s10	/usr	hfs	rw	0	5	# /usr
/dev/dsk/c0d0s1	swap	ignore	sw	0	0	# swap

1 This column: Backup frequency, not implemented, set to 0.

2 Root should be assigned a pass number value of 1 (*fsck* ignores file systems having a pass number value of 0).

A Logical Volume Manager File System */etc/checklist* Example

```
# cat /etc/checklist
/dev/vg00/lvol1    /        hfs   rw  0¹ 1² # root
/dev/vg00/lvol2    /swap    swap  sw  0  0  # primary swap
/dev/vg00/lvol3    /usr     hfs   rw  0  2  # /usr
/dev/vg01/lvol1    /data    hfs   rw  0  3  # /data
```

1 This column: Backup frequency, not implemented, set to 0.

2 Root should be assigned a pass number value of 1 (*fsck* ignores file systems having a pass number value of 0).

/etc/mount

- The */etc/mount* command announces to the system that a removable file system is to be attached to the file tree at *directory*.
- Executing *mount* with no parameters shows all file systems currently mounted by printing the table contained in */etc/mnttab*.
- The command and command parameters are shown below:

  # */etc/mount -a*
  # */etc/mount [fsname directory [-f] [-o options] [-t type]]*
  # */etc/mount [-p] [-l] [-L] [-s] [-u]*

  Where:

Key	**Operation**
-a	Attempt to mount all file systems listed in */etc/checklist*.
fsname	Full path name of block special device file associated with file system.
directory	Full path of existing directory where the file system is to be mounted.
-f	Force the file system to be mounted.
-o options	Defaults—use options specified, or one or more of the following separated by commas: *rw*—Read/Write (default). *ro*—Read only. *suid*—Set user ID execution allowed (default). *nosuid*—Set user ID execution denied.
-l	Displays local mount information.
-L	Displays local mount information, plus cluster-wide *NFS* mounts.
-s	7.0 treatment of */etc/mnttab* file: does not add kernel mount information to */etc/mnttab*.
-p	Print the list of mounted file systems in a format suitable for use in */etc/checklist*.
-t type	Specifies a file system type. Acceptable types are *hfs*, *cdfs*, and *nfs*.

/etc/umount

- Detaches a file system from the HP-UX tree.
- File system must be quiescent.
- The root file system and the file systems used in conjunction with dynamic swapping cannot be unmounted.
- The command and common parameters are shown below:

  # /etc/umount -a [-v] [-s] [-h host] [-t type]
  # /etc/umount fsname | mount_point_dir

  Where:

Key	**Operation**
-a	Attempt to unmount all file systems listed in */etc/mnttab*.
fsname	Full path name of special device file associated with the file system, or the full path name of the mount-point-directory associated with the file system to be unmounted.
mount_point_dir	Full path name of the system's mount-point-directory.
-v	Verbose.
-s	7.0 treatment of */etc/mnttab* file (does not add kernel mount information to */etc/mnttab*.
-h host	Unmount only those file systems in */etc/mnttab* that are remote-mounted from the host.
-t type	Unmount only file systems mounted with a given type.

/usr/bin/bdf

- Prints the amount of free disk space available on the specified file system.
- *bdf* with no options prints information on all mounted file systems.
- The command and parameters are shown below:

  $ */usr/bin/bdf [-b] [-i] [-l|-L] [-t type | [filesystem] | file] ...]*

  Where:

Key	Operation
-b	Report on the file systems; include dynamic swap information.
-i	Report the number of used and free inodes.
-t type	Report on the file systems of a given type (*hfs* or *nfs*).
filesystem	Special device file associated with file system (for example */dev/dsk/c0d0s4*).
file	Print information on the file system that contains *file*.
-l	Display disk-space-available information for a locally mounted file system.
-L	Display information for the file system that can be unmounted from the local cnode (includes file systems mounted on the local node and cluster-wide *NFS* mounts).

- Example (*bdf -b* prints the following information):

Filesystem	**kbytes**	**used**	**avail**	**capacity**	**Mount**
/dev/dsk/c0d0s4	102512	26665	65596	29%	/
/dev/dsk/c0s10	123295	42739	68226	39%	/usr
/dev/dsk/c0d0s0	23168	32	20816	0%	/tmp
/dev/dsk/c0d0s5	207127	15	186399	0%	/mnt
swapping	4096	2048	2048	50%	/mnt
/dev/dsk/c0d0s3	27912	26100	1812	102%[1]	/extra

1 File system */extra* is filled beyond MINFREE by 2%.

MINFREE Space vs. User File System Space

- File system performance rapidly decreases when the file system is filled beyond 90% of its total capacity.
- To prevent this performance degradation, HP-UX compares the actual file system fill to the file system's minimum allowed free space (MINFREE) value. When a file system's fill value leaves less than the MINFREE value, only a superuser can write on the remaining free file space.
- The default value of MINFREE is set by HP-UX to 10. MINFREE can be set by */etc/newfs* when the file system is created, or altered using the */etc/tunefs* command. For example:

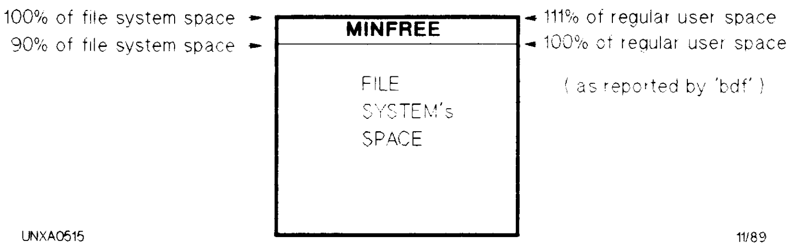


Figure 6-15. MINFREE Space vs. User File System Space

To calculate user writable space in percent:

% user_space = 100 - MINFREE

File System Organization

HP-UX Conventional 600/800 Boot Section Organization

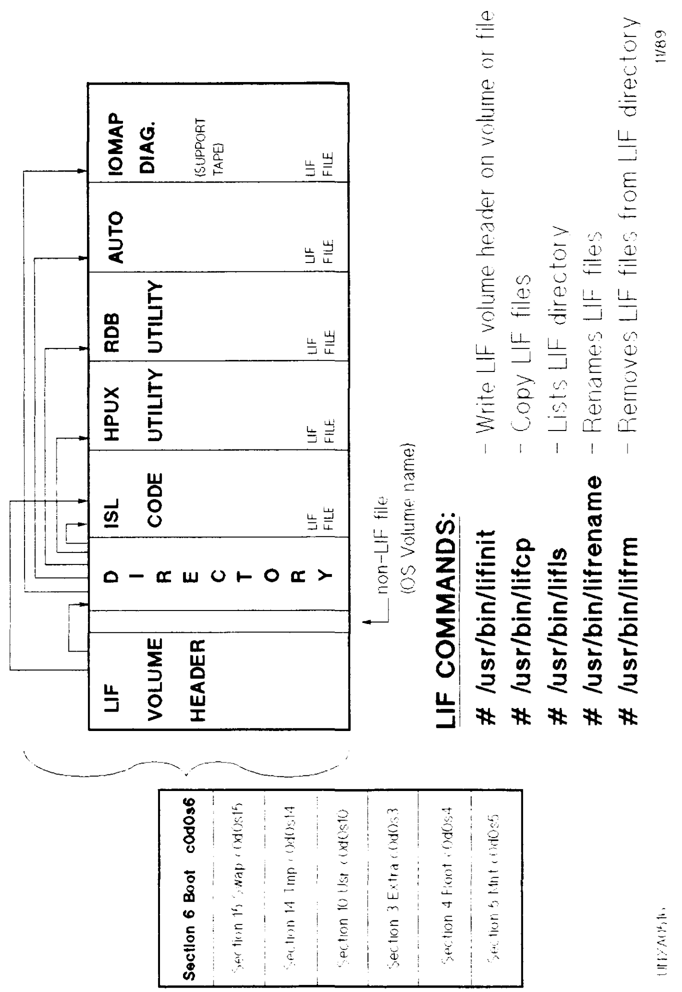
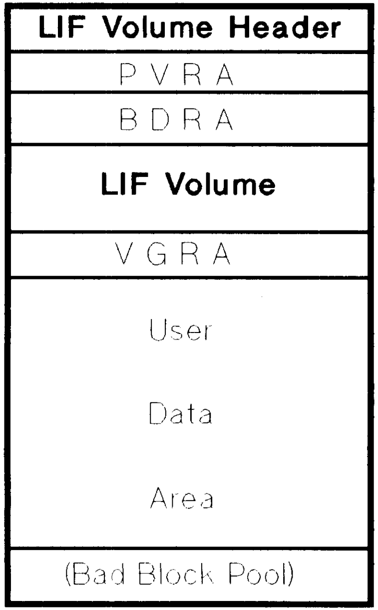


Figure 6-16. Conventional 600/800 Boot Section Organization



lvm030

Figure 6-17. LVM 600/800 Boot Section Organization

LIF Volume Header Contents

- Eight blocks used for SwitchOver/UX
- Label information for LIF Volume

LIF Volume Contents

- LIF Utilities/Files

ISL	← Initial System Loader
HPUX	← loads HP-UX kernel file
AUTO	← contains the autoboot string
LABEL	← used by HPUX to locate the root logical volume and during maintenance mode boot to configure root and primary swap

HP-UX High-Performance File System (HFS) Cylinder Groups

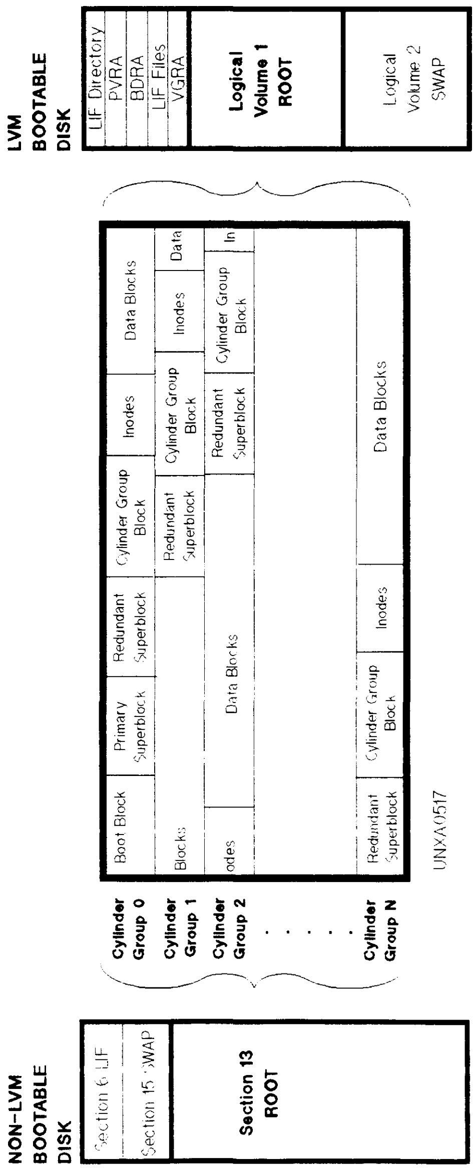
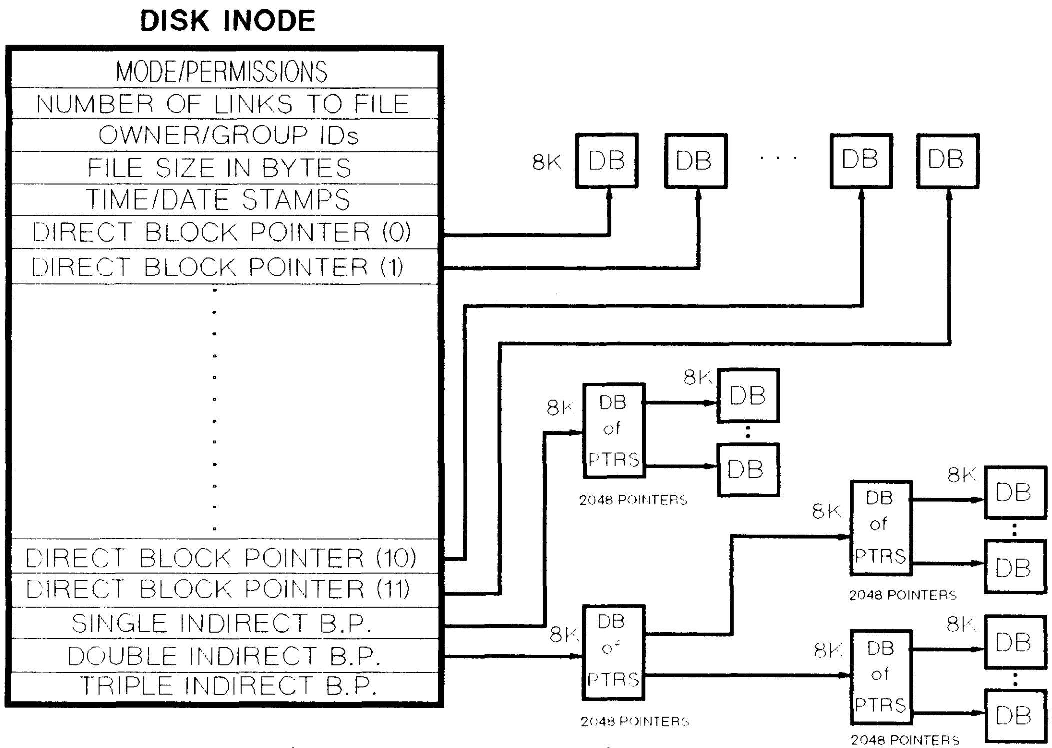


Figure 6-18. HP-UX High-Performance File System (HFS) Section Layout



UNIX MAXIMUM FILE SIZE = 8K [12 + 2048 + (2048^2) + (2048^3)]

CURRENT SPECTRUM MAXIMUM = DOUBLE INDIRECT (4GB maximum per file)

UNXA0521 1/89

Figure 6-19. Inodes

Causes of File System Corruption

- Improper shutdown procedures.
  - Not using *shutdown* or *reboot* to halt the CPU.
  - Physically write-protecting a mounted file system.
  - Taking a mounted file system off-line.
- Improper startup procedures.
  - Not checking a file system for inconsistencies.
  - Not repairing inconsistencies found.
- Hardware failure.
  - Disk I/O subsystem failure.

File System Checker

/etc/fsck File System Checker

- *fsck* checks for file system corruption by comparing the customer file system to an internal *fsck* standard that defines how a high performance HP-UX file system should be constructed.

  If corruption is detected, *fsck* will attempt to repair the damage.
- *fsck* has two operating modes:

preen	Automatically corrects inconsistencies that will not result in data loss.
interactive	Prints a brief error message for each inconsistency and prompts the user for the corrective action.

/etc/fsck Syntax

Caution

fsck should not be run on a mounted file systems (except root which cannot be unmounted). Users must not access files while *fsck* is running so use *shutdown* before invoking *fsck* when running *fsck* online.

fsck should be run on character device files.

- Preen Mode Run String:

  # *fsck -p [device_file]*

  # *fsck -P [device_file]*

- Interactive Mode Run String:

  # *fsck [-y] [-n] [-b block] [-q] [device_file]*

Where:

Key	**Operation**
-p	Check file system for inconsistencies.
-P	Check file system specified if not cleanly unmounted.
-b block	Use redundant superblock specified in *block* variable to check file system. Use when primary superblock is corrupted.
-y	Assume *yes* response to all *fsck* questions (use with caution).
-n	Assume *no* response to all *fsck* questions (do not write to file system).
-q	Fix counts in superblock and cylinder groups; print brief message.
device_file	Device file of the file system to be checked (for example */dev/[r]dsk/cXd0sY* or */dev/vgXX/rlvolY*). If not specified, *fsck* runs on high performance systems in */etc/checklist*.

Five Basic Steps to Repairing File Systems

1. Run *fsck* in preen mode to repair simple file system errors.
2. If file system inconsistencies still exist, run *fsck* in the interactive mode with the *-n* option. Redirect the output to a printer or file.
3. Analyze the error printout or the file created in step 2 to determine the problem.
4. Mount the file system (for example, # *mount -f /dev/dsk/c0d0s10 /mount_point_dir*) and copy the files to be removed by *fsck* to a clean file system.
5. Invoke *fsck* interactively and repair the damage.

Logical Volume Manager Basics

- Using LVM, you can combine one or more disks (physical volumes) into a volume group, which can then be subdivided into one or more logical volumes.
- LVM is not a file system. It is a manager that points to the start and end of logical volume data space for each physical disk that the logical volume happens to span.
- Logical volumes resemble disk sections, but with some important differences:
  - Logical volumes can range in size from 1 MB to 4 GB.

    Maximum size: File system 4 GB, Root 2 GB, Raw file system data 2 GB
  - Logical volumes can be expanded or reduced in size as needs change.

Warning **Reducing a logical volume's size will result in data loss. Always back up logical volume data before reducing logical volume size.**

LVM Spans Disks

- File systems can be larger than a physical disk's size.
- Logical volumes can grow and shrink allowing more efficient space usage. Volume Group

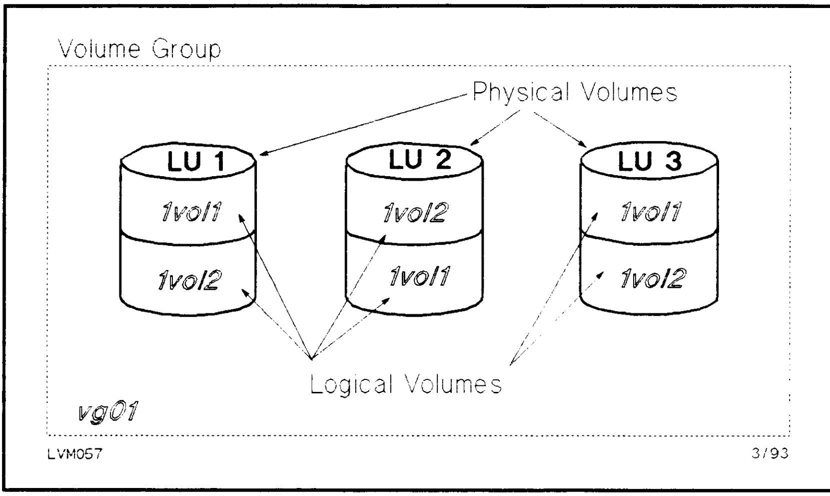


Figure 6-20. LVM Spanning Disks

LVM's Logical to Physical Extent Mapping

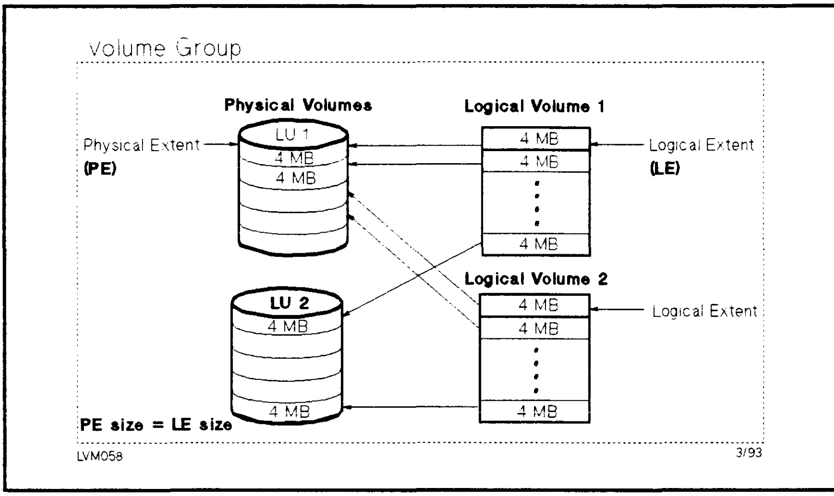


Figure 6-21. LVM Logical to Physical Extent Mapping

Extent size must be a power of 2 (range: 1 - 256 MB). Set by *vgcreate -s*

Displaying LVM Information

- For Physical Volumes

  \# *pvdisplay [-v] /dev/dsk/c*d0s2* (displays all physical disks searching for LVM physical volume information)

- For Volume Groups

  \# *vgdisplay [-v] /dev/vg** (displays information on all volume groups)

- For Logical Volumes

  \# *lvdisplay [-v] /dev/vg*/lvol** (displays all logical volumes in all volume groups)

- To display kernel devices on LVM Bootable Disks

  \# *lvlnboot -v* (displays root, primary swap, and dump logical volumes)

- For Swap Information (all swap including LVM)

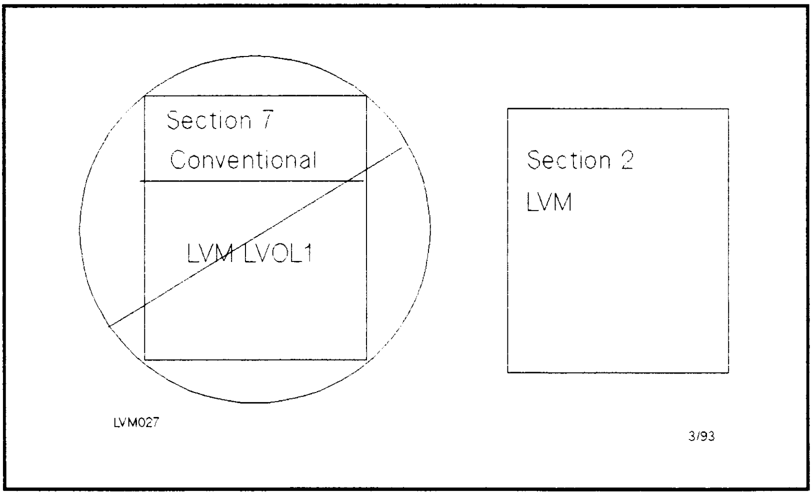
  \# *swapinfo*

- The contents of */etc/lvmtab*

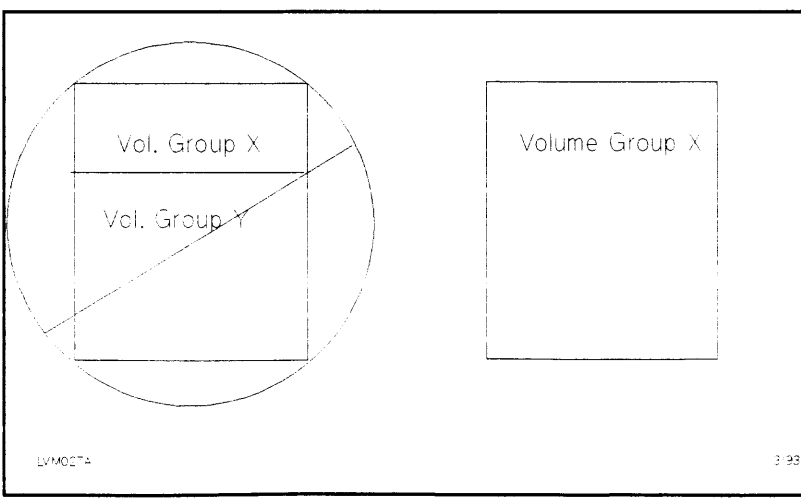
  \# *strings /etc/lvmtab* (displays volume group/physical volume relationships)

Logical Volume Manager Rules

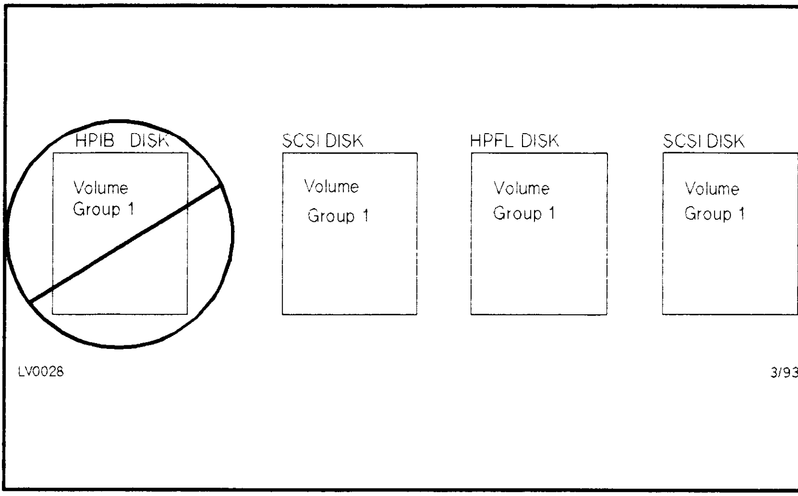
1. A disk drive must be dedicated exclusively to the LVM.



2. A disk drive can be a member of only one volume group.

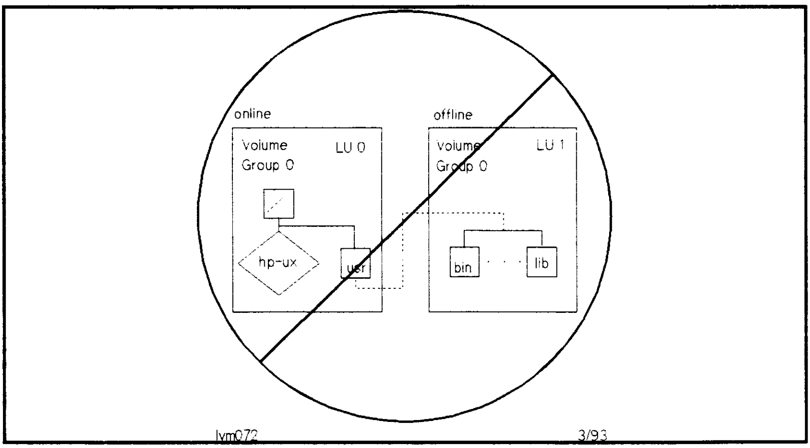


3. HPIB interfaced drives must not be mixed with HPFL or SCSI drives. HPIB disks must reside in their own exclusive volume group(s).



4. The maximum number of logical volumes in a volume group is 255. (Range is 1 - 255; Default is 255; Kernel tunable parameter: maxlvs)
5. The maximum number of physical extents allowed per physical volume is 65,535. (Range is 1 - 65,535; Default is 1016; Kernel tunable parameter: maxpxs)
6. Maximum number of physical volumes per volume group is 255. (Range is 1 - 255; Default is 16; Kernel tunable parameter: maxpvs)

7. More than half of the configured LVM disks in a volume group must be present to change or activate that volume group. Quorum is checked both during configuration changes (for example, when creating a logical volume) and at state changes (for example, if a disk fails). If quorum is not maintained, LVM will not acknowledge the change. To override quorum check use *vgchange* with the *-q* option. If quorum is not met on the *root* volume group, the system will not boot. This makes it wise to override quorum on root using *vgchange -q n*. If this change is not made and quorum is not met, use *ISL > hpux -lq* to boot.



8. Any time a change is made to the root volume group (i.e., root, swap, dump, or file system) Boot Data Reserved Area (BDRA) must be updated (see the *lvlnboot* for details on accomplishing this task). Failure to do this may result in an unbootable system.

LVM Device Files

- The */etc/lssf* command does NOT list LVM special files. Use *ll*.
- Physical volume

  /dev/[r]dsk/cXd0s2

Key	Operation
[r]	If present, indicates character (raw) access
X	Integer logical unit (lu) number

- Volume Group

  /dev/vgXX/group

Key	Operation
XX	Integer volume group number (0 ... 255)
group	Must be called *group*

- Logical Volume

  /dev/vgXX/[r]lvolY

Key	Operation
XX	Integer volume group number (0 ... 255)
[r]	If present, indicates character (raw) access
Y	Integer logical volume number (1 ... 255)

Example:

```
# ll /dev/vg02
crw-rw-rw-    1 root root   64 0x020000 Sep 21 10:59 group
brw-r-----    1 root root   64 0x020001 Sep 21 10:59 lvol1
crw-r-----    1 root root   64 0x020001 Sep 21 10:59 rlvol1
```

Key	Operation
64	Major number always 64
02	HEXADECIMAL volume group number
0000	Always zeroes for *group*
01	HEXADECIMAL logical volume number

LVM Physical Volume Commands

pvcreate	Makes a disk an LVM disk (a physical volume).
pvdisplay	Displays information about physical volumes in a volume group.
pvchange	Sets physical volume characteristics to allow or deny allocation of additional physical extents on this disk.
extendfs	Extends the size of a logical volume at the physical volume level.
pvmove	Moves allocated physical extents from source to destination within a volume group.

LVM Volume Group Commands

vgcreate	Creates a volume group.
vgdisplay	Displays information about volume groups.
vgchange	Activates or deactivates one or more volume groups. Allows a volume group to mount with or without a quorum.
vgextend	Extends a volume group by adding disks to it.
vgreduce	Reduces a volume group by removing one or more disks from it.
vgscan	Scans all disks and looks for logical volume groups.
vgsync	Synchronizes mirrors that are stale in one or more logical volumes.
vgremove	Removes definition(s) of volume group(s) from the system.
vgexport	Removes a volume group from the system without modifying the information found on the physical volume(s).

vgimport — Adds a volume group to the system by scanning physical volumes which have been exported using *vgexport*.

LVM Logical Volume Commands

lvcreate — Creates a logical volume.

lvdisplay — Displays information about logical volumes.

lvchange — Changes characteristics of logical volume including availability, scheduling policy, permissions, block relocation policy, allocation policy, mirror cache availability.

lvextend — Increases disk space allocated to a logical volume.

lvmerge — Merges two logical volumes into one logical volume.

lvreduce — Decreases disk space allocated to a logical volume.

lvremove — Removes one or more logical volumes from a volume group.

lvsplit — Splits a mirrored logical volume into two logical volumes.

lvsync — Synchronizes mirrors that are stale in one or more logical volumes.

Creating Boot Disks in the Root Volume Group

- Creating a bootable disk is useful for two specific cases:

  Creating a mirror for the root logical volume.
  Creating a new root logical volume (moving root from one disk to another).

- Briefly, the steps to add a boot disk to the root volume group are:

  1. Use *pvcreate -B* when you make the disk a physical volume.
  2. Add the disk to a volume group with *vgextend*.
  3. Use *mkboot* to place boot utilities (LIF) in the boot area.
  4. Use *mkboot -a* to modify the AUTO file in the boot LIF area.

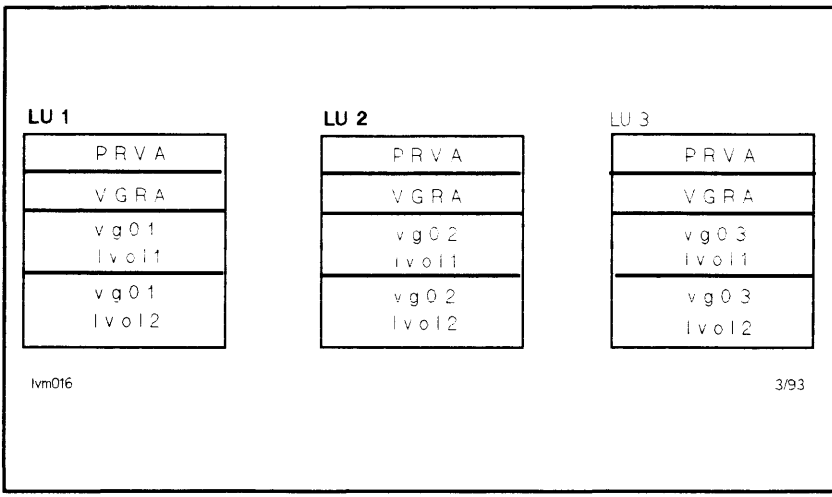
Note Other procedures will apply. See System Administration Tasks, Managing Logical Volumes, Chapter 8.

LVM Data Structures Backup

- Run *vgcfgbackup* periodically (at least once per day) and whenever a change is made to the LVM configuration including:

  Adding or removing disks to or from a volume group
  Changing boot disks in a volume group
  Creating or removing logical volumes
  Extending or reducing logical volumes

Shown below is a LVM system that needs LVM data structure backup.



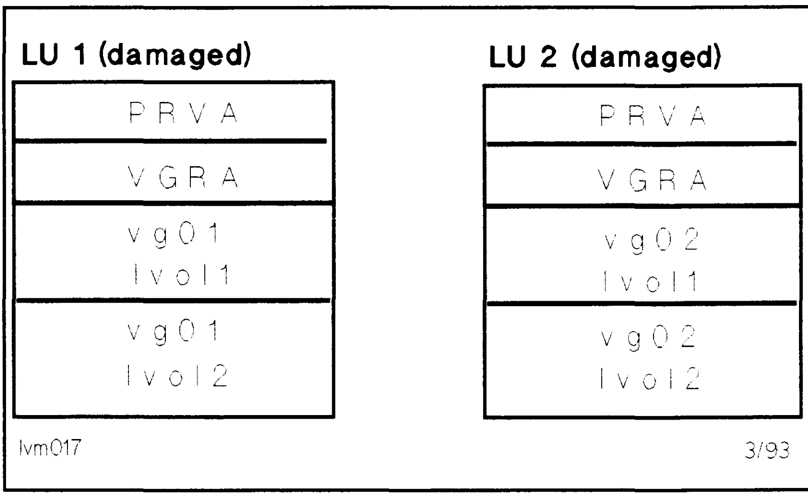
With all disks in the volume group online, run:

\# *vgcfgbackup vg01* (backup placed in */etc/lvmconf/vg01.conf)*
\# *vgcfgbackup vg02* (backup placed in */etc/lvmconf/vg02.conf)*
\# *vgcfgbackup vg03* (backup placed in */etc/lvmconf/vg03.conf)*

Warning *vgcfgbackup* **does not back up user data! It only backs up LVM data structures.**

LVM Data Structures Restore

- Use *vgcfgrestore* to restore LVM data structures after a disk hardware failure (one of many steps), or the result of improper use of HP-UX commands such as *dd*, *mkfs*, or *newfs*.
- Shown below is the LVM system that needs LVM data structure restored:



Note User data is not restored by the *vgcfgrestore* command.

- Make sure that all volume groups involved are made not available:

  # vgchange -a n vg01
  # vgchange -a n vg02

- Restore LVM data structures on all damaged physical volumes:

  # vgcfgrestore -n vg01 /dev/rdsk/c1d0s2
  # vgcfgrestore -n vg02 /dev/rdsk/c2d0s2

- Make all volume groups involved available:

  # vgchange -a y vg01
  # vgchange -a y vg02

Is */etc/lvmtab* Blown? */etc/vgscan* to the Rescue!

- The */etc/lvmtab* file contains important LVM volume group information. If this file is missing or corrupted, *vgscan* can restore it based on LVM structures on the LVM disk drives and device file information.
- The three steps shown below will produce valid information in */etc/lvmtab*.

  1. If */etc/lvmtab* exists, rename it */etc/lvmtab.old* with the *mv* command.
  2. Make sure all physical volumes are powered up and on-line.
  3. Run *vgscan -v* to create and update */etc/lvmtab*.

Logical Volume Manager Commands

/etc/extendfs

- If the original hfs filesystem created on *logical_vol_path* does not make use of all available space, *extendfs* can be used to increase the *hfs* filesystem size. The root (/) system cannot be extended.
- The *extendfs* command and options are shown below:

  # extendfs [-q] [-v] [-s size] logical_vol_path

Key	**Operation**
-q	Query the size of *char_dev_file*. Do not extend the volume specified by *logical_vol_path*.
-v	Verbose.
-s size	Integer number of 1024 byte blocks to add to *hfs* file system. If *-s size* is not specified, maximum possible size is used.
logical_vol_path	Character device file of disk to be extended. Example: */dev/vgXX/rlvolY*

To extend a generic file system:

 # umount /dev/vgXX/lvolY

(filesystem must be unmounted before *extendfs*)

 # lvextend -L larger_size_in_MB /dev/vgXX/rlvolY

 # extendfs /dev/vgXXrlvolY

/etc/lvchange

- Changes the characteristics of a logical volume. Optional command-line options and parameters specify the type and extent of the change. All options take effect immediately. Before deactivating a logical volume, unmount the logical volume using *umount* before using *lvchange*.
- The *lvchange* command and non-mirror options are shown below:

  # lvchange [-a availability] [-p permission] [-r relocate] [-C contiguous] lv_path

Key	**Operation**
-a availability	Activate or deactivate the availability of a logical volume. Availability has two allowed values: *y* = logical number is available, *n* = logical volume is temporarily unavailable (all current processes using the logical volume will still have access).
-p permissions	Sets read/write permissions. Permissions has two allowed values: *r* = read-only, *w* = read/write. Default is read/write allowed.
-r relocate	Set bad block relocation. *y* = relocate blocks (default), *n* = do not allow bad block relocation.
-C contiguous	Set allocation policy. *y* = allocate extents in ascending order without gap, *n* = allocate extents where available.
lv_path	Path of logical volume. Example: */dev/vgXX/lvolY*

/etc/lvcreate

- Creates a new logical volume within a volume group specified by *vol_group_name*. If *lv_name* is not specified, a system-generated name of the form lvolN is created, where N is the minor number of the new volume. Once a logical volume is created, it can be altered by *lvchange*, *lvextend*, and *lvreduce*.
- The *lvcreate* command and common options are shown below:

  # lvcreate [-l num_logical_extents | -L size_in_MB] [-m num_mirrors] [-n lv_name] [-p permissions] [-r relocate] [- C contiguous]

Key	**Operation**
-l num_logical_extents	Integer number of logical extents to be allocated to the created logical volume.
-L size_in_MB	Integer number of megabytes to be allocated to the created logical volume. May be rounded up by system.
-m num_mirrors	Specifies number of mirrors that contain the same data. Value can be 1 or 2.
-n lv_name	Simple file name (not path) of logical volume to be created.
-p permissions	Permissions is *w* (read-write) or *r* (read-only). Default is read-write.
-r relocate	Set bad block relocation. *y* = relocate (default), *n* = do not relocate bad blocks.
-C contiguous	Allocate physical extents with no gaps (required by root and dumps devices), *n* = allow gaps between physical extents (default), *y* = do not allow gaps between physical extents.

/etc/lvdisplay

- Displays the characteristics and status of each logical volume specified by the *lv_path* parameter. If the *-v* option is specified, *lvdisplay* displays physical volume (PV) distribution information and map information about the physical extents corresponding to logical extents of the logical volume or volumes.
- The *lvdisplay* command and options are shown below:

  # lvdisplay [-v] lv_path ...

Key	**Operation**
-v	Verbose. Displays extensive information about the logical volume(s) specified by *lv_path*.
lv_path	Path(s) of logical volume to be displayed. Example: */dev/vgXX/lvolY*

To display a generic logical volume:

 # lvdisplay -v /dev/vgXX/lvolY

/etc/lvextend

- Increases the number of physical extents allocated to a logical volume. To limit the allocation to specific physical volumes, use the names of one or more physical volumes in *phys_vol_path*.
- The *lvextend* command and options are shown below:

  #lvextend {-l num_logical_extents | -L size_in_MB | -m num_mirrors} lv_path [phys_vol_path ...]

Key	Operation
-l num_logical_extents	Total integer number of all logical extents (existing + desired extension size). Number must be LARGER than the current logical extent allocation.
-L size_in_MB	Integer number of Megabytes to extend the logical volume. Number must be LARGER than the current size of the logical volume (existing + desired extension size). System may round up.
-m num_mirrors	Sets the number of physical extents allocated for each logical extent. Values can be 1 or 2. *num_mirrors* must be greater that the current number of mirrors for the logical volume. Data in the new copies is synchronized—this can be time consuming. Note: *-m* cannot be used on HP-IB devices.
lv_path	Path of logical volume to be extended. Example: */dev/vgXX/lvolY*
phys_vol_path	Path of physical volume(s) to extend on. Example: */dev/dsk/cXd0s2*

/etc/lvlnboot

- Prepares a logical volume to be root, primary swap, or dump volume. If a non-existent logical volume is specified, this command fails. If a different logical volume is already linked to the root or primary swap, the command fails.
- The *lvlnboot* command and options are shown below:

  # lvlnboot [-v] [-r root_lv] [-s swap_lv] [-d dump_lv] [-R vol_group_name]

Key	**Operation**
-v	Verbose. With no other arguments, prints information on all root volume groups.
-r root_lv	Defines *root_lv* to be the root volume the next time the system is booted on the volume group. Updates the Boot Data Reserved Area such that the volume group is used to locate the root file system. This allows the *root_lv* to be used as the root volume during a maintenance mode boot (ISL>hpux -lm (;0)hp-ux). The physical volumes must have been created using *pvcreate -B*, indicating that the physical volume is to be used as a Bootable Physical Volume. Also, *mkboot* must have been run on the Physical Volume. The *root_lv* must be a contiguous logical volume and cannot have Bad Block Relocation enabled. Example: *-r /dev/vgXX/lvolY*

Key	Operation
-*s swap_lv*	Defines *swap_lv* to be the primary swap volume next time the system is booted on the volume group. Updates the Boot Data Reserved Area. Any existing swap areas previously defined are removed. The *swap_lv* must be a contiguous logical volume, and a Root Logical Volume must have been previously defined by use of this command. Example: -*s /dev/vgXX/lvolY*
-*d dump_lv*	Defines *dump_lv* to be one of the dump volumes next time the system is booted on the volume group. Updates the Boot Data Reserved Area. The combined size of all the dump volume should be at least 2048 bytes larger than the total memory of the system. The additional 2 Kbytes is used to safeguard against dump to the bottom of the disk. Multiple dump devices can be configured, but each *dump_lv* must be entered on a separate *lvlnboot* line. The *dump_lv* must be a non-mirrored, contiguous logical volume. Example: -*d /dev/vgXX/lvolY*
-*R vol_group_ name*	Recovers any missing links to all of the logical volumes specified in Boot Data Reserved Area on each of the physical volumes in the volume group. Updates or rebuilds LIF's LABEL file. Example: -*R /dev/vgXX*

Warning

Any time a change is made to the Root Volume Group (typically vg00), run *lvlnboot* to update BDRA. If this is not done, the disk may not meet QUORUM requirements making the root disk unbootable.

/etc/lvmerge

- Merge to logical volumes of the same size, increasing the number of mirrored copies of the *master_lv_path* by the number of copies in the *backup_lv_path*. The data contained in the *backup_lv_path* is resynchronized using the data in the *master_lv_path*. All new data on the *backup_lv_path* is lost.
- Whenever a mirrored logical volume is split into two logical volumes, a bitmap is stored that keeps track of all writes to either logical volume in the split pair. When the two logical volumes are subsequently merged using lvmerge, the bitmap is used to decide which areas of the logical volumes need to be resynchronized. The bitmap continues to exist until the merge is completed, or until either of the logical volumes is extended or reduced, or the system is rebooted.
- If no bitmap is available, the entire logical volume is resynchronized.
- The normal usage for this command is to merge previously mirrored logical volumes that have been split (to allow off-line access to data for backup) using the *lvsplit* command. However, the two logical volumes are not required to have been the result of a previous *lvsplit* operation.
- The *lvmerge* command and options are shown below:

  # lvmerge backup_lv_path master_lv_path

Key	**Operation**
backup_lv_path	Logical volume path to be merged with master logical volume.
master_lv_path	Mirrored master volume path.

To merge *vg00 lvol1b* (copy) with *vg00 lvol1* (master):

 # lvmerge /dev/vg00/lvol1b /dev/vg00/lvol1

/etc/lvreduce

- Reduces the number of physical extents allocated to a logical volume.
- The *lvreduce* command and common options are shown below:

  # lvreduce {-l num_logical_extents | -L size_in_MB | -m num_mirrors} lv_path

Key	**Operation**
-l num_logical_extents	Integer number of logical extents to reduce logical volume.
-L size_in_MB	Integer number of Megabytes to reduce the logical volume.
-m num_mirrors	Set the number of physical extents (mirrors) allocated for each logical extent. Value may be 0 = no mirror; or 1 = one mirror copy. *num_mirrors* must be less than the current number of mirror copies.
lv_path	Path of logical volume to be reduced. Example: */dev/vgXX/lvolY*

To reduce vgXX, logical volume Y by 100 MB:

 # lvreduce -L 100 /dev/vgXX/lvolY

Warning **Reduce the size of a logical volume ONLY if you no longer need its current contents, or if you have safely backed up the contents to tape or to another logical volume.**

/etc/lvremove

- Removes the logical volume or volumes specified by *lv_path*. Logical volumes must be unmounted before removal.
- The *lvremove* command and options are shown below:

  # lvremove [-f] lv_path ...

Key	Operation
-f	Specifies that no user confirmation is required.
lv_path	Path(s) of logical volume to be displayed.
	Example: */dev/vgXX/lvolY*

To remove a generic logical volume:

 # umount /dev/vgXX/llvolY

 # lvremove /dev/vgXX/lvolY

/etc/lvrmboot

- Updates all physical volumes contained in volume group such that logical volume is removed as the root, primary swap, or a dump volume when the system is next booted on the volume group.
- The *lvrmboot* command and options are shown below:

  # lvrmboot [-v] [-r] [-s] [-d dump_lv] vol_group_name

Key	**Operation**
-v	Prints verbose messages.
-r	Removes definitions of the root, primary swap, or dump volumes from the given volume group. Updates Boot Data Reserved Area.
-s	Remove definition of the primary swap volume and all dump volumes from the given volume group. Updates the Boot Data Reserved Area.
-d	Remove definition of dump_lv as one of the dump volumes. Updates the Boot Data Reserved Area.
vol_group_name	Name of affected volume group. See examples below.

To remove a dump logical volume:

 # lvrmboot -v -d /dev/vg00/lvol6

To remove a root from volume group zero:

 # lvrmboot -r /dev/vg00

/etc/lvsplit

- *lvsplit* splits singly- or doubly mirrored *lv_path* into two logical volumes. A new logical volumes is created containing one copy of the data.
- If *suffix* is specified, the new logical volume is given the name *lv_pathsuffix*. If *suffix* is not specified, *lvsplit* assigns a new name using the suffix *b*.
- Whenever a mirrored logical volume is split into two logical volumes, a bitmap is stored that keeps track of all writes to either logical volume in the split pair. When the two logical volumes are subsequently merged using *lvmerge*, the bitmap is used to decide which areas of the logical volumes need to be resynchronized. This bitmap remains in existence until the merge is completed, or either of the logical volumes is extended, reduced, or split again, or the system is rebooted.
- The new logical volume must be checked with *fsck* before mounting. *lvsplit* flushes the file system to a consistent state except for pipes and unlinked but open files.
- The *lvsplit* command and options are shown below:

  # lvsplit [-s suffix] lv_path

Key	**Operation**
-s suffix	Assigns the characters supplied in *suffix* to the end of the logical volume name. Example: *-s backup*

To split an on-line logical volume which is currently mounted on */usr* so that a backup can take place:

 # lvsplit /dev/vg00/lvol3

 # fsck /dev/vg00/lvol3b

 # mkdir /lvol3b

 # mount /dev/vg00/lvol3b /lvol3b

 # fbackup -f /dev/rmt/0h -0vHi /lvol3b (backup using appropriate backup utility and string)

To merge the split pair after backup:

 # umount /lvol3b

 # lvmerge /dev/vg00/lvol3b /dev/vg00/lvol3

/etc/lvsync

- The *lvsync* command synchronizes the physical extents of the logical volume specified by *lv_path*. Synchronization occurs only on physical extents that are stale mirrors of the original logical extent. The synchronization process can be time consuming depending on the hardware and amount of data.
- The *lvsplit* command and options are shown below:

  # lvsync lv_path

Key	**Operation**
lvpath	The logical volume path of the master logical volume. Example: */dev/vg00/lvol3*

To resynchronize the mirrors on a logical volume:

 # lvsync /dev/vg00/lvol3

/etc/mkboot

- The *mkboot* command is used to install or update boot programs on the disk specified by the associated device file. Must be in single-user mode.
- The *mkboot* command and common Series 800 options are shown below:

  # mkboot [-b boot_file_path] device

  # mkboot [-b boot_file_path] [-i included_lif_file] [-l] [-p preserved_lif_file] [-v] device

  # mkboot [-a auto_file_string*] device*

Key	**Operation**
-b boot_file_path	Boot programs in the pathname specified are installed on *device*.
device	Character or Block special file associated with the disk.
-i included_lif_file	If specified one or more times, mkboot copies each file specified by *included_lif_file* and ignores any other LIF files in *boot_file_path*. The sole exceptions are the files ISL and HPUX, which are always copied. If LABEL is specified in *included_lif_file* and LABEL is not present on the boot disk's LIF, a minimal LABEL file is created allowing boot possibly without swap or dump.
-l	Treat disk specified by *device* as an LVM physical volume, regardless of whether or not it is currently set up as one.

Key	Operation
-p preserved_lif_file	If specified one or more times, mkboot will not overwrite the LIF file specified by *preserved_lif_file*. Typically used to prevent overwriting AUTO and with LVM and SwitchOver/UX file LABEL. If *preserved_lif_file* is not on the specified disk, mkboot fails.
-a "auto_boot_string"	If the *-a* option is specified, mkboot creates or updates the AUTO file on the disk specified by *device*. The autoboot string must have quotes if it contains blank spaces. Example: *-a "hpux (52.6.0;0)/hp-ux"*

Note

The default file used as source for *mkboot* is */usr/lib/uxbootlf*.

To copy all LIF utilities from */usr/lib/uxbootlf* to boot disk LU 0:

 # mkboot /dev/rdsk/c0d0s2 (LVM File System)

To copy and update LIF's AUTO file with boot string information on LU 0:

 # mkboot -a "hpux (;0)/hp-ux" /dev/rdsk/c0d0s2 (LVM File System)

/etc/pvcreate

- Initializes a raw disk device for use as a physical volume in a volume group. If *pvcreate* recognizes a pre-existing file system, it asks for confirmation. Command fails if the disk already belongs to a volume group.
- The *pvcreate* command and options are shown below:

  # pvcreate [-b] [-f] [-B] [-t disk_type] [-d soft_defects] phys_vol_path

Key	**Operation**
-b	Used to specify the numbers that correspond indexes of all known bad blocks on physical volume *phys_vol_path*. Specify indexes using decimal, octal, or hexadecimal numbers in standard C-language notation with numbers separated by new-line, tab, or form-feed character. If this option is not used, pvcreate assumes that the physical volume contains no bad blocks.
-f	Force creation of a physical volume (thus deleting any file system present) without requesting information.
-B	Make a physical volume bootable.
-t disk_type	Retrieve configuration information about disk_type from */etc/disktab*.
-d soft_defects	Specify minimum number of bad blocks that LVM should reserve for bad block relocation. Number <= 7039. Not supported on HPIB disks.
phys_vol_path	Character device file for target disk. Example: */dev/rdsk/c1d0s2*

/etc/pvchange

- Changes the characteristics and state of a physical volume by setting the allocation permission to allow or deny allocation of additional physical extents.
- The *pvchange* command and options are shown below:

  # pvchange -x extensibility phys_vol_path

Key	**Operation**
-x extensibility	Set allocation permission for additional physical extents on the physical volume specified. Extensibility has two allowed values: *y* = allow extension, *n* = do not allow extension.
phys_vol_path	Path of physical volume to be added to the volume group. Example: */dev/dsk/cXd0s2*

To deny allocation on a generic physical volume:

 # pvchange -x n /dev/dsk/cXd0s2

/etc/pvdisplay

- Displays information about the physical volume or volumes specified by the *phys_vol_path* parameter. If the *-v* option is specified, pvdisplay displays a map of the logical extents that correspond to the physical extents of each physical volume.
- The *pvdisplay* command and options are shown below:

  # pvdisplay [-v] phys_vol_path ...

Key	**Operation**
-v	Verbose. Displays extensive information about the physical volume(s) specified by *phys_vol_path*.
phys_vol_path	Path(s) of physical volume(s) to be displayed. Example: */dev/dsk/cXd0s2*

To display extensive information on LU3 and LU4:

 # pvdisplay -v /dev/dsk/c3d0s2 /dev/dsk/c4d0s2

/etc/pvmove

- Move allocated physical extents from one LVM physical volume to one or more LVM physical volumes. The *pvmove* command succeeds only if there is enough space on the destination(s) to hold all of the source physical extents. Source and destination must reside in the same volume group.
- The *pvmove* command and options are shown below:

  # pvmove [-n lv_path] source_pv_path [dest_pv_path ...]

Key	**Operation**
-n lv_path	Moves only allocated physical extents associated with *lv_path*. Example: */dev/vgXX/lvolY*
source_pv_path	Block device file of source physical extents. Example: */dev/dsk/cXd0s2*
dest_pv_path	Block device file(s) of destination device(s). Example: */dev/dsk/cYd0s2*

Backup of LVM Data Structures With */etc/vgcfgbackup*

- The *vgcfgbackup* command creates or updates a backup of LVM volume group data structures (PVRA, VGRA, and where applicable on a bootable LVM disk, BDRA) into a file.

Warning

Without an up to date copy of the LVM data structures *vgcfgbackup* creates, it is IMPOSSIBLE to recover from most, if not all, LVM problems. Volume groups MUST be backed up with *vgcfgbackup* after ANY changes to the group. It is highly recommended that all volume groups be backed up daily using *cron(1M)*.

- The default path for the backup is

  /etc/lvmconf/vol_group_name.conf

- The *vgcfgbackup* command and options are shown below:

  # vgcfgbackup [-u] [-f vg_conf_path] vol_group_name

Key	Operation
-u	Updates configuration file with latest LVM configuration.
-f vg_conf_path	Saves configuration in file specified by *vg_conf_path*. Example: *-f /my/backup/path/myvol.conf*
vol_group_name	Name of the volume group to backup. Example: */dev/vgXX*

Recovery of LVM Data Structures With */etc/vgcfgrestore*

- The *vgcfgrestore* command restores LVM data structures (PVRA, VGRA, and where applicable on a bootable LVM disk, BDRA) from a *vgcfgbackup* created backup file.
- The default path for the backup file to be used by *vgcfgrestore* is */etc/lvmconf/vol_group_name.conf*.
- The *vgcfgrestore* command and options are shown below:

  # vgcfgrestore {-n vol_group_name | -f vg_conf_path} -l

  # vgcfgrestore {-n vol_group_name | -f vg_conf_path} [-o old_pv_path] pv_path

Key	**Operation**
-n vol_group_name	Specify the volume group name of the backup file. Example: *-n vg01*
-f vg_conf_path	Specifies the entire path to the *vgcfgbackup* file. Do not use with *-n*. Example: *-f /my/backup/path/myvol.conf* (created with *vgcnfgbackup -f*).
-l	List information in *vgcfgbackup* backup file.
-o old_vol_path	Used to restore configuration information saved for physical volume old path pv path. This is used when a physical volume's name has changed since backup.
pv_path	Physical device path volume group file is to be restored. Example: */dev/rdsk/c1d0s2*

To restore the LVM data structures on volume group *vgYY*:

 # vgchange -a n vgYY

 # vgcfgrestore -n vgYY /dev/rdsk/cXd0s2

 # vgchange -a y vgYY

/etc/vgchange

- Activates or deactivates one or more volume groups as specified by the *-a* option; namely *y* or *n*.
- Before deactivating a volume group, make sure that all logical volumes in the volume group(s) are unmounted with the *umount* command.
- The *vgchange* command and non-mirror options are shown below:

  # vgchange -a availability [-q quorum] [-l] [-p] [-s] vol_group_name ...

Key	**Operation**
-a availability	Activate or deactivate the volume group. Availability has two allowed values: *y* = volume group is available, *n* = volume group is temporarily unavailable. When mirrors are involved, activating a group will perform the necessary mirror consistency recovery based on the state of the Mirror Write Cache and Mirror Consistency Recovery section of *lvdisplay*. See vgchange(1M) and lvdisplay(1M) for details.
-q quorum	Set activation of quorum requirement for the volume group. Qdefault), *n* = ignore quorum requirement.
-l	Do not allow open of logical volumes that belong to the volume group(s) specified.
-p	Activate volume group only if all physical volumes in group are available.
-s	Disable synchronization of stale physical extents within the volume group. Only applies to LVM mirrored disks.

/etc/vgcreate

- Creates a new volume group. *vol_group_name* is a symbolic name for the volume group and must be used in all references to it. *vol_group_name* is the path to a directory under */dev* which must contain a character special file named *group*. Except for the *group* entry, no other files must be present on */dev/vol_group_name*.
- The *vgcreate* command and common options are shown below:

  # vgcreate [-x extensibility] [-e max_phys_extents] [-l max_logical_vols] [-p max_logical_vols] [-s phys_extent_size] vol_group_name phys_vol_path ...

Key	Operation
-x extensibility	Set allocation permission for additional physical extents on the physical volume specified. Extensibility has two allowed values: *y* = allow extension, *n* = do not allow extension.
-e max_phys_ extents	Integer number sets maximum number of physical extents. Default maximum 1016. Kernel tunable with *maxpxs*.
-l max_logical_ vols	Integer number sets maximum number of logical volumes that a volume group can contain. Default maximum is 255. Kernel tunable with *maxlvs*.
-p max_logical_ vols	Integer number sets maximum number of physical volumes allowed in a volume group. Default is 255. Kernel tunable with *maxpvs*.
-s phys_extent_ size	Integer number sets the number of megabytes in each physical extent. Values in Mbytes may range from 1- 256. Value must be a power of 2 (i.e., 1, 2, 4, 8, 16 ... 128, 256).
vol_group_name	Name of volume group to be created. Example: */dev/vgXX*
phys_vol_ path	Block device file(s) associated with the disk drive(s) to be added to the volume group. Example: */dev/dsk/cXd0s2*

To create a generic volume group on two disk drives:

```
# vgcreate /dev/vgXX /dev/dsk/cYd0s2 /dev/dsk/cZd0s2
```

/etc/vgdisplay

- Displays information about volume groups. If the volume group name is specified, ***vgdisplay*** displays information for that volume group only. If no volume group name is specified, ***vgdisplay*** displays names and corresponding information for all defined volume groups.
- The *vgdisplay* command and options are shown below:

  # vgdisplay [-v] [vol_group_name ...]

Key	**Operation**
-v	Verbose. Displays extensive information about the volume group(s).
vol_group_name	Path(s) of volume group to be displayed. Example: */dev/vgXX*

To display extensive information from a generic volume group:

 # vgdisplay -v /dev/vgXX

/etc/vgexport

- Removes a volume group from the system without modifying the logical volume information as found on the physical volume(s). Volume group information is removed from */etc/lvmtab* file, and the associated device files including the volume_group_name directory are removed from the system. Useful in moving a volume group from one system to another.
- The *vgexport* command and options are shown below:

  # vgexport [-p] [-v] [-m mapfile] vol_group_name

Key	**Operation**
-p	Preview actions to be taken but do not make any changes. Best used with the *-v* option.
-v	Verbose. Prints the name(s) of the physical volumes associated with the volume group specified by *vol_group_name*.
-m mapfile	Specify the name of the file to which logical volume name(s) and numbers are to be written. If no name is specified, no file is written. *mapfile* is used as input to the *vgimport* command. Example: */vgXXmap*
vol_group_name	Name of volume group to export. Example: */dev/vgXX*

/etc/vgextend

- Extends a volume group by adding physical volumes to it.
- The *vgextend* command and common options are shown below:

  # vgextend [-x extensibility] vol_group_name phys_vol_path

Key	**Operation**
-x extensibility	Set allocation permission for additional physical extents on the physical volume specified. Extensibility has two allowed values: *y* = allow extension, *n* = do not allow extension.
vol_group_name	Name of affected volume group. Example: */dev/vgXX*
phys_vol_path	Path of physical volume to be added to the volume group. Example: */dev/dsk/cXd0s2*

/etc/vgimport

- Adds the specified volume group to the system. The physical volumes to be added are scanned to obtain volume group and physical volume information. A volume group directory (/dev/vgXX) and character *group* file must have been created prior to invoking this command. *vgimport* works in a similar manner to *vgcreate*, but does not activate the group (use *vgchange -a* to activate). */etc/lvmtab* is updated to reflect changes.
- The *vgimport* command and options are shown below:

  # vgimport [-p] [-v] [-m mapfile] vol_group_name phys_vol_path ...

Key	**Operation**
-p	Preview actions to be taken but do not make any changes. Best used with the *-v* option.
-v	Verbose. Prints the name(s) of the physical volumes associated with the volume group specified by *vol_group_name*.
-m mapfile	Name of file created by *vgexport*. If not specified, import with standard naming conventions (i.e., *[r]lvolX*). Example: */vgXXmap*
vol_group_name	Name of volume group to import. Example: */dev/vgXX*
phys_vol_path	Block device files of physical volumes to be imported. Example: */dev/dsk/cXd0s2*

/etc/vgreduce

- Reduce a volume group by removing one or more physical volumes. All but one physical volume can be removed. The last physical volume can be removed by *vgremove*. All logical volumes residing on the physical volume must be removed by executing *lvremove* before executing *vgreduce*.
- The *vgreduce* command and options are shown below:

  # vgreduce vol_group_name phys_vol_path

Key	**Operation**
vol_group_name	Name of the volume group to reduce. Example: */dev/vgXX*
phys_vol_group	Block device file(s) associated with the disk drive(s) to be removed from the volume group. Example: */dev/dsk/cXd0s2*

To remove a generic physical volume from a volume group:

 # vgreduce /dev/vgXX /dev/dsk/cYd0s2

/etc/vgremove

- Removes the last physical volume from the volume group and the definition of the volume of the group(s) specified. When completed, the volume can no longer be accessed.
- Before removing a volume group, two steps are necessary:
  1. Remove all but one of the logical volumes belonging to the volume group with *lvremove*.
  2. Remove the physical volume(s) in the volume group with *vgreduce*.
- The *vgremove* command and options are shown below:

  \# *vgremove vol_group_name* ...

Key	**Operation**
vol_group_name	Name(s) of volume group(s) to be removed. Example: */dev/vgXX*

To remove two generic volume groups:

\# *vgremove /dev/vgXX /dev/vgYY*

/etc/vgscan

- Searches physical volumes connected to the system, looking for logical volumes. It then groups these into volume groups. It searches the */dev* directory for all group volume files with LVM major numbers (64). After analyzing the information, it updates */etc/lvmtab* with this information.
- Move */etc/lvmtab* before invoking *vgscan*.
- The *vgscan* command and options are shown below:

  # vgscan [-p] [-v]

Key	**Operation**
-p	Preview the actions taken but do not update */etc/lvmtab*. This option is best used in conjunction with the *-v* option.
-v	Verbose.

To update */etc/lvmtab* by scanning the system:

 # mv /etc/lvmtab lvmtab.old

 # vgscan -p -v (preview changes to */etc/lvmtab*)

 # vgscan -v (create and update */etc/lvmtab* with current system information)

/etc/vgsync

- *vgsysnc* synchronizes the physical extents in each mirrored logical volume in the volume group specified by *vol_group_name*. Synchronization occurs only on the physical extents that are stale mirrors of the original logical extent.

  Unless disabled, the mirrors within a volume group are synchronized automatically when the volume group is activated by the *vgchange -a y* command.

- The *vgsync* command and options are shown below:

  # vgsync vol_group_name

Key	**Operation**
vol_group_name	The path of the volume group to be synchronized. Example: */dev/vgXX*

To synchronize vg03:

 # vgsync /dev/vg03

7

Cookbook Procedures

Spool-A-Printer Cookbook

1. Verify that the kernel contains the required interface and printer driver(s) (see *Installing Peripherals HP 9000 Series 800 Computers, HP Part Number B3108-90004* for driver information):

   # *lsdev*

   If the required device driver(s) are not present in the kernel, add driver include statement(s) (e.g., *include lpr0;*) to the */etc/conf/gen/S800* file. Use *uxgen* to generate a new kernel. Once this has been accomplished, reboot the computer on the new kernel.

2. Use *ioscan* to bind the necessary driver(s) to the new printer:

   # *ioscan -f*

   Verify that the hardware and software status is *ok* for the printer's hardware path.

3. Assign logical unit numbers and make default device file(s) for any new hardware detected by the *ioscan* command:

   # *cd /dev ; insf*

4. Find the printer's model/interface script:

   # *ll /usr/spool/lp/model | more*

5. Identify the printer's device file:

   # *lssf /dev/lp** (for non-serial printer)

   # *lssf /dev/tty?p?* (for serial printer)

6. Verify that the printer's device file is owned by *'lp'*:

   \# *ll /dev/devicefile* (use device file found in Step 5 above)

   To change ownership type: # *chown lp /dev/devicefile* (use device file found in Step 5 above).

7. Configure the printer into the LP Spooler:

   \# *lpshut*

   \# *lpadmin* -*p*printer_name (user's choice) -*v*/dev/devicefile (from Step 5 above) -*m*hpMODEL (from Step 4 above)

   \# *accept* printer_name

   \# *enable* printer_name

   \# *lpadmin* -*d*printer_name (execute this line if the spooled printer is the default printer)

   \# *lpsched*

8. Test the printer using LP Spooler, then check LP Spooler's status:

   \# *lp* -*d*printer_name */etc/passwd*

   \# *lpstat -t*

Add-A-Serial-Printer Cookbook

1. Connect the serial printer to an available serial port on a multiplexer panel (do not use the Access Port).

2. Verify that a device file exists for the new printer:

   # lssf /dev/ttyXpY

   Where:
   X = logical unit (lu) number of the mux assigned by *insf*.
   Y = port number the printer is attached to.

   mux0 lu X port Y hardwired address mod.slot */dev/ttyXpY*.

   If *lssf* fails to find the desired device file, check to see that the insf process has been correctly followed. If the kernel (hp-ux) has been configured correctly, *mksf* can create a new device file for the printer.

   # cd /dev

   # /etc/mksf -d muxN -l X -p Y -h /dev/ttyXpY

   Where:
   muxN = mux driver name (e.g., mux0, mux1, mux2)
   X = logical unit (lu) number of the mux assigned by *insf*.
   Y = port number the printer is attached to.

3. Edit */etc/inittab* using vi or ed. Delete any existing lines which refer to the device file that will be used by the new serial printer.

4. Configure the printer into the LP spooler using the steps in the Spool-A-Printer Cookbook earlier in this section.

Printer/Spooler Troubleshooting Cookbook

1. Check the hardware to see that it is cabled correctly, powered on, addressed correctly, and on-line. If possible, run the printer's self-test.

2. Check the printer's hardware and software status in the kernel:

   # ioscan -kf

   If the kernel does not indicate that the status of the printer's path is *ok*, check to see that the kernel has the necessary drivers using *lsdev*. If a drivers is missing, see the Spool-A-Printer Cookbook for details on adding drivers to the kernel.

3. Check the printer's device file for ownership and permissions:

   # ll /dev/printers_device_file

   If not owned by *lp*, change device file ownership by typing: *chown lp /dev/printers_device_file.*

4. Attempt to bypass the lp spooler by outputting directly to the printer's device file:

   - For HP-IB printers, type:

     # cat /etc/passwd > /dev/lpX

   - For serial printers, type the following:

     # nohup sleep 2000000000 < /dev/ttyXpY & stty -parenb \ -ienqak cs8 9600 -cstopb -clocal ixon -opost < /dev/ttyXpY

     Where: */dev/ttyXpY* = the printer's device file. *9600* = the printer's baud rate.

     # cat /etc/passwd > /dev/ttyXpY

   - If you receive some form of printout (perhaps without proper line feeds, etc.), the fundamental path to the printer is working. The LP spooler has software configuration problems.

5. Determine the LP Spooler's Status:

   # lpstat -t

   - If there is no evidence of the troubled printer, it has not been configured into the LP Spooler. Add it using the Add-A-Serial-Printer Cookbook and/or Spool-A-Printer Cookbook in this section.
   - If the printer has been spooled but fails to print, try the following:
     - If *lpstat* indicates scheduler is not running, start it using *lpsched*:

       # lpsched

     - If *lpstat* indicates that the system has no default destination printer, assign one using *lpadmin -d*:

       # lpadmin -dprinter_name

     - If *lpstat* indicates that the printer is rejecting requests, using the *accept* command:

       # accept printer_name

     - If *lpstat* indicates that the printer is disabled, enable it using the *enable* command:

       # enable printer_name

6. If the above procedure fails to remedy the problem, try removing the printer from the LP Spooler using the Remove-A-Spooled Printer Cookbook, then add the printer again using the Add-A-Serial Printer Cookbook and/or the Spool-A-Printer Cookbook.

Add-A-Terminal Cookbook

1. Connect the terminal to an available port on one of the mux panels (do not use the Access Port for this purpose).
2. Verify that a special device file exists for the new terminal:

   # *lssf /dev/ttyXpY*

   Where:

   X = logical unit (lu) number of the mux assigned by *insf*.
   Y = port number the terminal is attached to.

   muxN lu X port Y hardwired address mod.slot */dev/ttyXpY*.

   If *insf* fails to find the desired special device file, run *ioscan* to see if the MUX card has been configured correctly. If it has, *mksf* can create a new special device file for the terminal:

   # *cd /dev*

   # */etc/mksf -d muxN -l X -p Y*

   Where:
   muxN = name of the mux driver (e.g., mux0, mux1, mux2) *X* and *Y* have the same meaning as above.
3. Modify */etc/inittab* using vi or ed. Delete any existing lines which refer to the special device file that will be used by the new terminal.

4. Add the following line to */etc/inittab*:

   xx:2:respawn:/etc/getty -h ttyXpY 9600

   Where:

Key	**Operation**
xx	Unique one or two character identification.
2	Run level (typically *2* for multiuser terminal operation).
X	Multiplexer's logical unit (lu) number assigned by *insf*.
Y	Port number terminal is attached to.
9600	Terminal baud rate.

5. Force *init* to read the */etc/inittab* file:

   # *telinit q*

   If communication with the terminal fails, verify the hardware, terminal configuration, and */etc/inittab* file making any necessary changes. After making any necessary changes invoke the following commands:

   # *ps -ef* (to locate getty processes running against new terminal).

   # *kill -9 PID#* (kill process ID(s) associated with *ttyXpY*). This will automatically spawn a new getty process for the terminal.

6. Test the new terminal by attempting to login.

Add-A-Dial-In Modem Cookbook

1. Connect the MODEM to an available port on one of the mux panels (do not use the Access Port or ports 3 - 5 on the synapse panel).

2. Create a special device file for the new dial-in modem:

   # mksf [-d muxN [-l lu] [-p port] [-h | -i | -o] [-c] [path]

   Where:

Key	**Operation**
-d muxN	Specifies mux driver name to be used.
-l	Logical unit number (lu) of the device assigned by *insf*.
path	Default path name is tty<lu>p<port>.
-c	CCITT (European Standard).
-h	Hardwired (direct connect).
-i	Callin MODEM.
-o	Callout MODEM.
-p	Multiplexer port number to which the MODEM is attached.

   For example: to make a dial-in MODEM special device file for mux lu 2, port 2:

   # cd /dev

   # mksf -d mux0 -l 2 -p 2 -i ttyd2p2

   Where:

   -i specifies *callin* MODEM.
   ttyd2p2 is the special device file name.

3. Edit */etc/inittab* using vi or ed. Delete any existing lines which refer to the mux and port used by the new MODEM.

4. Add the following line to */etc/inittab*:

   xx:2:respawn:/etc/getty -h -t ZZZ ttydXpY 2400

   Where:

Key	**Operation**
xx	Unique one or two character identification.
2	Run level (typically *2* for multiuser MODEM operation).
ZZZ	Integer timeout value in seconds.
X	Multiplexer's logical unit (lu) number assigned by *insf*.
Y	Port number MODEM is attached to.
2400	MODEM baud rate (e.g, 1200, 2400, 9600).

5. Force *init* to read the */etc/inittab* file:

   # telinit q

   If communication with the MODEM fails, verify the hardware connections, MODEM configuration switches, and the associated */etc/inittab* getty line. Make any necessary changes. After making necessary changes invoke the following commands:

   # ps -ef (locate getty processes running against new MODEM).

   # kill -9 PID# (kill process ID(s) associated with *ttydXpY*). This will automatically spawn a new getty process for the MODEM.

6. Test the new dial-in modem by attempting to login from a remote location.

HP-UX Network Installation Cookbook

1. Install networking software by running */etc/update*. Load all software in the NETWORKING partition.

2. Edit */etc/rc* using vi or ed. Make the following change to the *initialize0* function:

   From—

   SYSTEM_NAME=unknown

   To read—

   SYSTEM_NAME=bashful

   Where: *bashful* is the user chosen system host name. This name must be unique among all the machines connected to this LAN.

Caution

Networking errors may result if the system name specified by */etc/rc* and */etc/src.sh* do not agree!

---

3. Edit */etc/hosts*. Add a line for each host in the network to be accessed. Include a line for the computer you are on. Each line in */etc/hosts* must begin with Internet Protocol (IP) address, followed by the official hostname, then any hostname aliases.

Caution

- Each Internet Protocol address must begin in column 1 of the */etc/hosts* file.
- Internet addresses must begin in column 1 of the line.
- Do not use leading zeros in the Internet Protocol Address. As an example *015.032.064.001* should be entered as *15.32.64.1*
- The last number following the final decimal (.) cannot be zero. As an example *15.32.64.0* has a final zero and is NOT allowed.

For example:

192.6.1.1	*bashful*	*shy*
192.6.1.2	*grumpy*	*maddog*
192.6.1.3	*doc*	
192.6.1.4	*sleepy*	

4. Edit the first if/else clause in the */etc/netlinkrc* script. Making the following changes to set the customer chosen DOMAIN and ORGANIZATION that this host will belongs to:

   From—

   ROOTSERVER=‘hostname‘
   NODENAME=$ROOTSERVER
   DOMAIN=‘bin/uname -n‘
   ORGANIZATION=standalone

   To read—

   ROOTSERVER=‘hostname‘
   NODENAME=$ROOTSERVER
   DOMAIN=*customer_chosen_domain_name*
   ORGANIZATION=*customer_chosen_organization_name*

5. Shutdown and reboot the system:

   # cd /

   # shutdown -r 0

6. Verify the hardware connection by running the *landad* diagnostic:

   a. Run *landad* using the default sections to obtain the LAN card's hex station address:

      # suplicen

      # sysdiag

      DUI> landad pdev=2/4.3 (2/4.3 = the LAN interface's hardware address)

   b. Run *landad* again specifying section 9. This will test the MAU. Use the hex station address obtained in the previous Step:

      # /usr/diag/bin/sysdiag

      DUI> run landad pdev=2/4.3 section=9

      If desired, repeat this for other hosts with known hex station addresses on the LAN. Each host must be connected to the LAN and powered on. It is not necessary for these hosts to be running HP- UX for this test to pass.

7. Check the network packet passing capability by using *ping*. First *ping* the host your are on, if this is successful, *ping* host directly connected to the LAN. Each host must be configured as above and booted up.

   # ping bashful

   # ping grumpy (Note: Use *Ctrl-c* to stop *ping*)

   # ping doc

   # ping sleepy

8. Verify the Telnet feature of ARPA:

   # telnet grumpy

   login: root

   password:

   (allows session from terminal with no screen mode capability)

   Ctrl-d (exits Telnet)

Add-A-User Cookbook

1. Add a login line to */etc/passwd* using vi or ed:

   caitlin::215:200:caitlin x8174:/mnt/users/caitlin:/bin/sh

   /etc/passwd format:

   user_name:password:user_id:group_id:comment_field:\
   login_directory:command

2. Add or modify a line in */etc/group* using vi or ed (optional):

   ces::200: (May exist if others in new user's group)

   lab::300:caitlin (Add to access other group privileges)

   /etc/group format:

   group_name:password:group_id:member1,member2

3. Make the user's login (or home) directory:

   # *mkdir /mnt/users/caitlin*

4. Change the owner of */mnt/users/caitlin* from *root* to *caitlin*:

   # *chown caitlin /mnt/users/caitlin*

   chown syntax:

   # *chown owner file*

5. Change the group ownership of */mnt/users/caitlin* from *other* to *ces* (or 200):

   # *chgrp ces /mnt/users/caitlin*

   chgrp syntax:

   # *chgrp group file*

6. Login to the new account.

7. Copy or create a shell customization script (e.g., *cp /etc/d.profile $HOME/d.profile*).

Memory Core Dumps

- If HP-UX gets a High Priority Machine Check (HPMC) or panics, the kernel will automatically write an image of the entire physical memory onto the primary swap device.
- After the HPMC or panic and the memory write to swap, the kernel halts present processes and attempts to reboot. On reboot */etc/rc* will invoke */etc/savecore*. *savecore* recognizes the attempt to save the memory image on the primary swap or dumps device into two files on the *tmp* file system. Normally the destination of the files, *hp-ux.X* and *hp-core.X*, is the */tmp/syscore* directory.
- The size of *tmp* should be larger than physical memory to enable it to hold a full memory dump. If the size of *tmp* is smaller than physical memory, *savecore* will store only a portion of the dump on *tmp*; the rest will be lost. To clear a dump without saving it, at the shell prompt type: */etc/savecore -c /tmp*
- */etc/savecore* will clear special bits on the primary swap device when the entire dump has been saved to files on the file system. On subsequent reboots, */etc/savecore* will not recognize that a dump resides on the primary swap device.

Modifying */etc/rc* to Save Memory Dumps

1. Execute the *bdf* command. Select a file system which contains more free space than the size of physical memory. Note the associated mount_point_directory:

   # */usr/bin/bdf* (see *bdf* command described earlier)

2. Make a *syscore* directory on the mount_point_directory selected in step 1 above:

   # */bin/mkdir /mount_point_directory/syscore*

3. Using vi or ed, make the following changes to the *save_core* function in the */etc/rc* file:

   From—

   if [-x /etc/savecore] && [-d /tmp/syscore]
   then
   /etc/savecore/tmp/syscore

   To read—

   if [-x /etc/savecore] && [-d /mount_point_directory/syscore]
   then
   /etc/savecore /mount_point_directory/syscore]

4. After a memory dump and subsequent reboot, two files will be saved on the directory created in step 2. These are:

   /mount_point_directory/syscore/hp-ux.X

   and

   /mount_point_directory/syscore/hp-core.X

   Where: The trailing *X* in the file name is an integer number; the first dump saved is zero (0), the next 1, etc.

Add Dynamic Swap Cookbook

To add dynamic swap space on a high performance file system (HFS), perform the following steps:

1. Find a file system with sufficient free space to add dynamic swap space, without filling the file system and preventing users from storing files:

   # */usr/bin/bdf -b*

2. Use *swapon* to create the desired dynamic swap space on the file system selected in the previous Step:

   # */etc/swapon mount_point_dir min_blocks max_blocks* \
   fs_reserve_blocks priority

   Where:

Key	**Operation**
mount_point_dir	Full path name of mount_point_directory where the file system dynamic swap is to reside.
min_blocks	Minimum number of file system blocks to be allocated for dynamic swap (0 = do not take any blocks at *swapon*).
max_blocks	Maximum size in blocks dynamic swap is allowed to take from the file system (0 = take as many blocks as necessary).
fs_reserve_blocks	Number of file system blocks that are saved for file system use only (0 = no blocks reserved for the file system).
priority	Indicate the order space is taken from the file system for use as swap (priority zero file systems are taken first).

   min, *max*, and *fs_reserve_blocks* size must be specified as an integer number of file system blocks. The size of the file system blocks can be found in */etc/disktab* for the model and section number of the disk that attaches to the mount_point_directory.

Note

Once swap is enabled by *swapon*, *min_blocks* and *fs_reserve_blocks* cannot be changed, but *max_blocks* can be increased in size.

3. If dynamic swapping is to be made permanent (enabled on system start-up) add a line to */etc/checklist* in the following format using vi or ed:

   */dev/dsk/cXd0sY mount_point_dir swapfs min=A, lim=B, *
   res=C, pri=P 0 0

   Where:

Key	**Operation**
/dev/dsk/cXd0sY	Full path name of block special file where dynamic swap is to reside.
mount_point_dir	Full path name of mount_point_directory where the file system dynamic swap is to reside.
A	Integer minimum number of blocks taken by swap (0 = do not take any blocks at *swapon*).
B	Integer maximum number of blocks swap can take from file system (0 = take as many blocks as necessary).
C	Integer number of blocks reserved for file system use (0 = no reserved blocks).
P	Integer priority (0 = use dynamic swap space first).

4. Verify that dynamic swapping has been enabled:

   # /usr/bin/bdf -b

Note

Once dynamic swapping has been enabled, it cannot be deactivated until *swapfs* lines are removed from */etc/checklist* for the section you wish to deactivate; then reboot the system. If dynamic swapping on a file section was not made *permanent*, simply reboot, and all temporary swapping sections will be deactivated.

Dynamic Swapping Features

- Dynamic swap allows paging and swapping on ordinary high performance file systems on an overflow basis. When dedicated swap section space is exhausted, dynamic swapping on enabled file systems begins according to a predetermined priority scheme.
- Dynamic swap space can be added while HP-UX is running using the *swapon* command. *swapon* allows the user to set sizes and limits to prevent dynamic swap pace from growing so large that user file space is critically reduced or eliminated. *swapon* will allocate no less than the minimum amount of blocks specified, but may allocate more for efficient use of swap space. It will not allocate blocks that are reserved for the file system users.
- Swapping performance is reduced when swapping to file systems. Swapping is done in file system sized blocks (typically 8K or 4K blocks), instead of 64K blocks used on dedicated swap sections. Data is scattered across the file system instead of contiguous as on dedicated swap sections.
- Once dynamic swapping is enabled on a section, it cannot be disabled without rebooting HP-UX. File systems with swapping enabled cannot be unmounted.
- Only one directory per file system can be specified for dynamic swapping.

The */etc/swapon* Command

The *swapon* command enables additional devices (dedicated swap sections) or file systems for paging and swapping. The *swapon* command and options are shown below:

\# */etc/swapon -a*

\# */etc/swapon name | [directory min limit reserve priority]*

Where:

Key	**Operation**
-a	All dedicated swap devices (swap sections) marked as *swap*, and all dynamic swapping file systems marked as *swapfs* in */etc/checklist* are made available for swapping.
name	Full path name of block special file of dedicated swap section of disk. This section must be *uxgen*ed into the system.
directory	Full path name of mount_point_directory of the file system to be enabled for dynamic swapping.
min	Integer number of file system blocks to take from the file system (default = 0). Only valid with *directory* option.
limit	Integer maximum number of blocks swap can take from the file system (default = 0). Only valid with *directory* option.
reserve	Integer number of file system blocks that are reserved for file system only (default = 0). Only valid with *directory* option.
priority	Integer indicates order that space is taken from the file systems for use as swap. Lowest priority taken first (default = 0).

Add-A-DataPair/800-Mirror-Disk Cookbook

To create a non-LVM mirror disk using DataPair/800 on a file system except *root* and *swap* perform the following steps:

1. If the disk drive(s) have never been initialized, initialize the media:

Caution Initialization will destroy any data on the disk drive.

/usr/bin/mediainit /dev/diag/dsk/cXd0 Where: *X* = the logical unit (lu) number of the disk assigned by *insf* (for HP-UX versions prior to 8.0, when addressing an HP-FL device *X* becomes 2000 + the lu number).

2. Create an HP-UX file system on the mirrored primary and secondary disk sections, only if valid file system(s) do not already exist:

   # */etc/newfs /dev/rdsk/cXd0sY hpZZZZ*

   Where:

Key	Operation
X	The logical unit (lu) number of the disk assigned by *insf* (for HP-UX versions prior to 8.0, when addressing an HP-FL device *X* becomes 2000 + the lu number).
Y	Section number of the file system to be created (see */etc/disktab*).
ZZZZ	Model number of HP disk (see */etc/disktab* for supported models).

3. If a mount_point_directory does not already exist for the primary disk section of the mirrored disk, make one using the following command:

   # */bin/mkdir /newdir*

   Where: *newdir* is the full path name of directory the new file system is to be mounted under.

4. Using vi or ed, delete any existing lines in */etc/checklist* that refer to the secondary disk section of the mirror disk pair. If a line does not exist for the mirrored disk primary disk section, add it according to the following format:

   /dev/dsk/cXd0sY /newdir hfs rw 0 P # /newdir

   Where:

Key	**Operation**
/dev../.0sY	Full path name of block special device file for the mirror disk primary disk section.
newdir	The full path name of directory the new file system is to be mounted under.
hfs	Section is a high performance file system.
rw	May be read or write to (default).
0	Backup frequency (set to 0).
P	Integer pass number determines order that *fsck* checks file systems.
#	Begin comment field.

5. If not already mounted, mount the mirrored disk primary disk section:

   # /etc/mount -a

   Where:

   -a attempts to mount all file systems in */etc/checklist*.

6. Mirror the primary and secondary disk sections:

   # */etc/mirror -c /dev/rdsk/cXd0sY online /dev/rdsk/cZd0sY \
   offline*

   Where:

Key	**Operation**
/cXd0sY	*X* = the primary disk logical unit number (see */etc/conf/gen/S800* for lu information). *Y* = section number of the primary disk.
/cZd0sY	*Z* = the secondary disk logical unit number (see */etc/conf/gen/S800* for lu information). *Y* = section number of the secondary disk (must be same as primary).

7. Update the secondary disk section with an exact image of the primary disk section using the *reimage* option (this will also bring the secondary disk online):

   # */etc/mirror -r /dev/dsk/cXd0sY*

   Where:

Key	**Operation**
X	The primary disk logical unit number plus 2000 (see */etc/conf/gen/S800* for lu information).
Y	Section number of the primary disk.

8. After the reimage has completed, verify that both sections of the mirror disk are now ONLINE and GOOD:

   # */etc/mirror -l*

DataPair/800-Mirror *root*-and-*swap* Cookbook

To create a mirror of the *root* or *swap* sections, perform the following steps:

1. If the disk drive(s) to be used as secondary disk section(s) for *root* or *swap* have never been initialized, initialize the media.

Caution Initialization will destroy any data on the disk drive.

   \# */usr/bin/mediainit /dev/diag/dsk/cXd0*

   Where: *X* = the logical unit (lu) number of the disk assigned by *insf* (for HP-UX versions prior to 8.0, when addressing an HP-FL device *X* becomes 2000 + the lu number).

2. Create an HP-UX file system(s) for *root* and/or *swap* mirrored secondary disk section(s):

   \# */etc/newfs /dev/rdsk/cXd0sY hpZZZZ*

   Where:

Key	**Operation**
X	The logical unit (lu) number of the disk assigned by *insf* (for HP-UX versions prior to 8.0: when addressing an HP-FL device *X* becomes 2000 + the lu number).
Y	Section number of *root* or *swap* file system (see */etc/conf/gen/S800* for section).
ZZZZ	Model number of HP disk (see */etc/disktab* for supported models).

3. Using vi or ed, edit the *include* section of */etc/conf/gen/S800*. Uncomment the following line:
   From—

   */*include mirror;*/*

   To read—

   include mirror;

4. Using vi or ed, edit the *kernel devices* section of */etc/conf/gen/S800* file to mirror the *root* section, *swap* section, or both:

   From—

   root on disc2 lu X section Y;

   swap on disc2 lu X section Z;

   To read—

   root on disc2 lu X section Y mirrored on disc2 lu W section Y;

   root on disc2 lu X section Z mirrored on disc2 lu V section Z;

   Where:

Key	**Operation**
W	*root* secondary disk lu number.
Y	Section number of *root* file system.
V	*swap* secondary disk lu number.
ZY	Section number of *swap* file system.

5. Type the following commands to make a special file for mirror disk and change mode, ownership, and group for the special file:

   # /etc/mknod /dev/rdsk/mirconfig c 12 0x7f0000

   # chmod 666 /dev/rdsk/mirconfig

   # chown bin /dev/rdsk/mirconfig

   # chgrp bin /dev/rdsk/mirconfig

6. Follow the *UXGEN Process* to generate a new kernel (hp-ux). After a new kernel is booted, the system will mirror the *root* and/or *swap* file system and automatically re-image the mirrored file system(s) on system boot-up.

Modifying the LIF *auto* File Cookbook

1. At the ISL prompt, execute the *lsautofl* command. This command displays the content of the *autoexecute* file.
2. Execute the *set autofile* utility and make the desired changes.

   For example:

   ISL>hpux set autofile (;6) hpux (2/4.0.0;13)/hp-ux
3. Check the display message to ensure the change was correct.
4. From the *ISL>* prompt, set *Autoboot* ON. Check *Primpath* for proper disk path address.
5. Reset the system. The system should automatically boot up without operator intervention.

Add-A-Conventional-File-System Cookbook

1. Initialize the disk, if it has not been initialized:

Caution Initialization destroys all data on the disk.

/usr/bin/mediainit /dev/diag/dsk/cXd0 Where:

X = the logical unit (lu) number of the disk assigned by *insf*.

2. Create an HP-UX file system with *newfs*:

Caution Making a new file system with *newfs* destroys all data associated with the section specified. Make sure you specify the correct logical unit number and section number when using this command.

/etc/newfs /dev/rdsk/cXd0sY hpZZZZ

Where:

Key	Operation
X	Logical unit number (lu) *X* assigned by *insf*.
Y	Section number of the new file system (see */etc/disktab*).
hpZZZZ	HP disk model number (see */etc/disktab* for supported models).

3. Make a mount-point-directory for the new file system:

/bin/mkdir /newdir

Where:

newdir = full path name of directory the new file system is to be mounted under.

4. Add a new line to */etc/checklist* for the new file system:

   /dev/dsk/cXd0sY /newdir hfs rw 0 P # /newdir

   Where:

Key	**Operation**
/dev/./.0sY	Full path name of block special device file for the file system to be mounted.
/newdir	Full path name of directory the new file system is to be mounted under.
hfs	Type (options): *hfs* - high performance file system, *nfs* - remote NFS file system, *swap* - swap file system, *swapfs* - dynamic swap file system, *ignore* - entry is ignored by *mount* and *fsck*.
rw	Options (use default options, or comma separated list of options): *ro* - read only, *rw* - read write (default), *suid* - Set-user-ID execution allowed (default), *nosuid* - Set-user-ID not allowed.
0	Back frequency (set to 0).
P	Integer pass number *P* determines order *fsck* checks file systems.
#	# begins comment field.

5. Mount new file system:

   # */etc/mount -a*

   Where:

   a = Attempt to mount all file systems in */etc/checklist*.

LVM Cookbooks

Add An LVM Disk Cookbook

1. Verify that the kernel has the correct drivers for the new disk drive. (Consult "Installing Peripherals" if you are unsure of the correct drivers.)

   # lsdev

   If the correct drivers are not displayed, edit */etc/conf/gen/S800*. Add the necessary "include" statements, then use the UXGEN process to generate a new kernel.

2. Halt the system.

   # shutdown -h 0

3. Install the new hardware. Verify that there are no bus address conflicts.

4. Boot the system. Once booted, use the following command to verify detection of the new disk drive. Note the assigned logical unit number for the new disk drive.

   # ioscan -kfC disk

5. Create a physical volume for LVM use. Use the logical unit number obtained in step 4 of this procedure for the *X* in the command below:

   # pvcreate -f /dev/rdsk/cXd0s2

Warning **This will destroy any data on the disk.**

6. Check *dev* to see if the volume group to reside on the new disk has a directory under */dev*. The default name for a volume group is *vgYY* where *YY* is the integer number of the group.

   # ls /dev | grep vg

7. If necessary, create a new directory under */dev* for a new volume group to reside on the disk.

   # mkdir /dev/vgYY

8. If a new volume group is to be created, make a ***group*** character device file for the new volume group.

   # mknod /dev/vgYY/group c 64 0xZZ0000

   Where:

Key	**Operation**
ZZ	HEXADECIMAL group number.

9. Create or extend the volume group to the new disk. Do ONE of the following:

   # vgcreate /dev/vgYY /dev/dsk/cXd0s2

   Which is used to create a new logical volume.

   # vgextend /dev/vgYY /dev/dsk/cXd0s2

   Which is used to extend a logical volume onto a new disk.

10. Create a logical volume on the new disk. Specify the size of the logical volume to be created in megabytes following the *-L* option. The logical volume number will be the next integer not used (begins with 1).

    # lvcreate -L SSS vgYY

    Where:

Key	**Operation**
SSS	The integer number of MB to be allocated for this logical volume.

11. Create the physical file system for the new logical volume.

    # newfs /dev/vgYY/rlvolZ MODEL

    Where:

Key	**Operation**
Z	The integer logical volume number.
MODEL	The HP disk's model number (see */etc/disktab* for supported models, and use *diskinfo /dev/rdsk/cXd0s2* to see model number).

12. Make a mount-point-directory on root (/) for the new logical volume.

    # mkdir /vgYYlvolZ

13. Add a new line in */etc/checklist* using *vi* or *ed* for the new logical volume.

    /dev/vgYY/lvolZ /vgYYlvolZ hfs rw 0 P # /vgYYlvolZ

    Where:

Key	**Operation**
/dev/vgYY /lvolZ	The full path name of the block device file for the logical volume to be mounted.
/vgYYlvolZ	The full path name of the directory that the file system is to be mounted under.
hfs	Type (options): *hfs* - high performance file system, *nfs* - remote *NFS* file system, *swap* - swap file system, *swapfs* - dynamic swap file system, *ignore* - entry is ignored by *mount* and *fsck*.
rw	Options. (use default options, or comma separated list of options): *ro* - read only, *rw* - read/write (default), *suid* - Set-user-ID execution allowed, *nosuid* - Set-user-ID not allowed.
0	Backup frequency (set to 0).
P	Integer pass number *P* which determines the order in which *fsck* checks the file systems.
#	# begins the comment field.

14. Mount the new file system.

    # mount -a

15. Backup the LVM data structures (PVRA and VGRA) for all groups affected. The default path name for the backup will be */etc/lvmconf/vgYY.conf*.

    # vgcfgbackup /dev/vgYY

LVM Example: Adding a New Disk; Volume Group; Logical Volumes

The following example adds a new disk to the system. The new disk will be placed into a new volume group. Next, two new 100MB logical volumes will be created on the new disk drive.

The commands assume the following:

1. The new disk is HP Model Number *C2474S*
2. The new disk's device files are */dev/[r]dsk/c4d0s2*
3. The new volume group is *vg05*
4. The new logical volumes are *lvol1* and *lvol2*

The following commands will accomplish the task:

```
# pvcreate -f /dev/rdsk/c4d0s2
# mkdir /dev/vg05
# mknod /dev/vg05/group c 64 0x010000
# vgcreate /dev/vg05 /dev/dsk/c4d0s2
# lvcreate -L 100 vg05
# lvcreate -L 100 vg05
# newfs /dev/vg05/rlvol1 C2474S
# newfs /dev/vg05/rlvol2 C2474S
# mkdir /vg5lv1 /vg5lv2
# mount /dev/vg05/lvol1 /vg5lv1
# mount /dev/vg05/lvol2 /vg5lv2
# vgcfgbackup vg05
```

Extend An LVM Logical Volume Cookbook

1. Determine the device files for existing physical volumes in the associated volume group.

   # strings /etc/lvmtab

2. Find all physical volumes associated with the volume group that the logical volume to be extended resides in. Look for free space with the following command.

   # pvdisplay /dev/dsk/cXd0s2

   Where:

Key	Operation
X	The physical volume's logical unit number as indicated by *strings /etc/lvmtab*.

   Repeat this for each physical volume in the logical volume's volume group.

   Multiply the free PE extents times the PE extent size to determine if adequate free space is available to satisfy user space requirements.

Note

If sufficient free space is not available within the volume group, use the "Add an LVM Disk Cookbook" to add a new physical volume to the volume group. Once this has been accomplished, use this cookbook to extend the desired logical volume.

---

3. Determine the total size of the logical volume. The size in MB will be displayed. Note this value. It will be used in step 4 below.

   # lvdisplay /dev/vgYY/lvolZ

   Where:

Key	Operation
YY	The integer number of the volume group associated with the logical volume to be extended.
Z	The target logical volume's integer number.

4. Extend the logical volume.

   # lvextend -L total_MB_in_lvolume /dev/vgYY/lvolZ

   Where:

Key	Operation
total_MB_ in_lvolume	Must be a larger number than that found in step 3 above.

5. Unmount the target logical volume.

   # umount /dev/vgYY/lvolZ

   Where:

Key	Operation
YY	The integer number of the volume group associated with the logical volume to be extended.
Z	The target logical volume's integer number.

6. Extend the physical file system.

   # extendfs /dev/vgYY/rlvolZ

7. Mount the extended logical volume.

   # mount -a

8. Backup LVM data structures (PVRA and VGRA).

   # vgcfgbackup vgYY

LVM Example: Extend a Logical Volume

The following example extends a logical volume from 100 MB to 200 MB.

The commands assume the following:

1. The volume group is *vg05*
2. The logical volume to be extended is *lvol2*
3. The logical volume is in */etc/checklist*

The following commands to accomplish the task:

```
# lvextend -L 200 /dev/vg05/lvol2
# umount /dev/vg05/lvol2
# extendfs /dev/vg05/rlvol2
# mount -a
# vgcfgbackup vg05
```

Reduce the Size of an LVM Logical Volume Cookbook

1. Backup all user data in the logical volume to be reduced.

   (Use *fbackup*, *cpio*, or *tar* backup utilities as appropriate or copy user data to another logical volume in the volume group using *pvmove*.)

2. Unmount the logical volume to be reduced using the *umount* command.

   # cd / ; umount /dev/vgXX/lvolY

   Where:

Key	**Operation**
XX	The integer number of the volume group associated with the logical volume to be reduced in size.
Y	The target logical volume's integer number.

3. Determine the present total size of the logical volume to be reduced using *lvdisplay*. This value is presented as *LV Size (Mbytes)*.

   # lvdisplay /dev/vgXX/lvolY

4. Reduce the size of the logical volume.

   # lvreduce -L total_MB_in_lvolume /dev/vgXX/lvolY

   Where:

Key	**Operation**
total_MB_ in_lvolume	Must be a smaller number than that found in step 3.

5. Reduce the size of the physical file system.

   # newfs /dev/vgXX/rlvolY MODEL

Key	**Operation**
Y	The integer logical volume number.
MODEL	The HP disk's model number (see */etc/disktab* for supported models, and user *diskinfo /dev/rdsk/cXd0s2* to see model number).

6. Mount the newly reduced logical volume using the *mount* command.

   # mount -a

7. Backup LVM data structures (PVRA and VGRA).

   # vgcfgbackup /dev/vgXX

Remove an LVM Volume Group Cookbook

1. Backup all user data in the volume to be removed.

   (Use *fbackup*, *cpio* or *tar* backup utilities as appropriate.)

2. Determine the names of the logical volumes residing in the volume groups to be removed.

   *# lvdisplay /dev/vgXX/lvol**

   Where:

Key	**Operation**
XX	The integer number of the volume group associated with the logical volume to be reduced in size.

3. Unmount all logical volumes in the volume group to be removed.

   *# cd / ; umount /dev/vgXX/lvol**

4. Remove the logical volumes found in step 2 from the volume group.

   # lvremove /dev/vgXX/lvolY

Note Multiple logical volume paths (*/dev/vgXX/lvolY*) can be included in the same *lvremove* command line.

5. Determine if multiple physical volumes are associated with the volume group to be removed. This is indicated by the number of block device files listed for the volume group to be removed.

   # strings /etc/lvmtab

6. If more than one physical volume (as indicated by the number of block device files listed for the volume group), REMOVE ALL BUT ONE of the physical volumes using the *vgreduce* command.

   # vgreduce /dev/vgXX /dev/dsk/cYd0s2

Note

Multiple physical volume paths (*/dev/dsk/cYd0s2*) can be included in the same *vgreduce* command line.

7. Now, remove the volume group.

   # vgremove /dev/vgXX

LVM Example: Moving A Logical Volume

The following example moves a logical volume to a new disk that has more room.

1. The volume group is *vg01*
2. The logical volume to be moved is *lvol1*
3. The physical volume associated with *lvol1* is */dev/rdsk/c1d0s2*
4. The destination physical volume is */dev/rdsk/c2d0s2*

The following commands will accomplish the task:

```
# pvcreate -f /dev/rdsk/c2d0s2
# vgextend /dev/vg01 /dev/dsk/c2d0s2
# pvmove -n /dev/vg01/lvol1 /dev/dsk/c1d0s2 /dev/dsk/c2d0s2
# vgcfgbackup vg01
```

Exporting and Importing an LVM Volume Group Cookbook

1. Unmount any logical volumes associated with the volume group you wish to export.

   *# umount /dev/vgXX/lvol**

   Where:

Key	**Operation**
XX	The integer number of the volume group to be exported.

2. Make the volume group unavailable.

   # vgchange -a n /dev/vgXX

3. Use *vgexport* to remove volume group information from */etc/lvmtab*.

   # vgexport -v /dev/vgXX

4. Using *vi* or *ed*, remove any lines in */etc/checklist* that refer to logical volumes in the volume group to be exported.

5. Shut the system down.

   # cd / ; shutdown -h 0

6. Detach the physical disks from the exported system. Physically attach them to the new host system. Check the bus addresses to assure no conflict, power the new disk(s) on, and boot the new host.

7. Create a new directory under /dev for the volume group you wish to import.

   # mkdir /dev/vgYY

   Where:

Key	**Operation**
YY	The integer number of the volume group to be imported.

8. Make a *group* character device file for the volume group to be imported.

   # mknod /dev/vgYY/group c 64 0xZZ0000

   Where:

Key	Operation
ZZ	HEXADECIMAL group number.

9. Determine the logical unit number(s) for the new disks to be imported.

   # ioscan -kfC disk

   Note the logical unit number(s).

10. Now import the volume group.

    # vgimport -v /dev/vgYY /dev/dsk/cLd0s2

    If multiple disks, include block device file for each disk being imported on this line.

    Where:

Key	Operation
YY	The integer number of the volume group to be imported.
L	The integer logical unit number of the imported disk drive as displayed in step 8 above.

11. Activate the newly imported volume group.

    # vgchange -a y /dev/vgYY

12. Make a mount-point-directory on root (/) for each logical volume in the newly imported volume group.

    # mkdir /vgYYlvolM

    Where:

Key	Operation
M	The integer logical volume number.

13. Add a new line in */etc/checklist* using *vi* or *ed* for the new logical volume(s).

    /dev/vgYY/lvolM /vgYYlvolM hfs rw 0 P # /vgYYlvolM

    Where:

Key	Operation
/dev/vgYY /lvolM	The full path name of the block device file for the logical volume to be mounted.
/vgYYlvolM	The full path name of the directory that the file system is to be mounted under.
hfs	Type (options): *hfs* - high performance file system, *nfs* - remote *NFS* file system, *swap* - swap file system, *swapfs* - dynamic swap file system, *ignore* - entry is ignored by *mount* and *fsck*.
rw	Options. (use default options, or comma separated list of options): *ro* - read only, *rw* - read/write (default), *suid* - Set-user-ID execution allowed, *nosuid* - Set-user-ID not allowed.
0	Backup frequency (set to 0).
P	Integer pass number *P* which determines the order in which *fsck* checks the file systems.
#	# begins the comment field.

14. Mount the new file system(s).

    # mount -a

    Where:

Key	Operation
a	Attempts to mount everything in */etc/checklist*.

15. Backup the LVM data structures (PVRA and VGRA) for all groups affected. The default path name for the backup will be */etc/lvmconf/vgYY.conf*.

    # vgcfgbackup /dev/vgYY

Adding Secondary Device Swap on a Logical Volume Cookbook

All physical extents used for swap (or dumps) must be contiguous, that is, the extents must be allocated in ascending order with no gaps between extents. Due to this restriction, it is rarely possible to extend primary swap space.

Fortunately, adding swap space is simple. Adding swap space consists of creating a logical volume with contiguous extents; adding a line to */etc/checklist*; and finally, executing the *swapon* command.

Calculating the necessary swap space proves more problematic.

A rough calculation of swap space requirements for a small standalone system can be made by determining the total size of the applications likely to be running at peak times and add this to the size of physical memory. A precise and detailed calculation can be made. To do this refer to *System Administration Tasks*, Appendix B, Swap Space Calculation.

As a minimum, it is recommended that swap space be equal to the size of physical memory.

Once the total swap size has been estimated or precisely calculated, proceed with the following:

1. Determine the current state of the system's swap space.

   # swapinfo

   The output of the *swapinfo* command will indicate the type of swap by location, how much of it is available, used, and free. If *hold* is displayed, it indicates how much space the system has reserved based on possible requirements of running processes.

2. If additional swap space is required, perform the first nine steps (when applicable) of the procedure given in *Create a Logical Volume on a New Disk*, found earlier in this section. Do NOT proceed beyond step 9 of this procedure.

3. After the applicable first nine steps of *Create a Logical Volume on a New Disk* have been performed, create a contiguous logical volume of sufficient size (when added to existing swap space) to bring total swap space to the required amount. Specify the size of the logical volume to be created in megabytes following the *-L* parameter.

   # lvcreate -C y -L SSS vgYY

   Where:

Key	Operation
SSS	The integer number of MB to be allocated for this logical volume.
YY	The integer number of the volume group to create the logical volume in.

4. Copy the kernel to the new logical volume.

   # cp /hp-ux /dev/vgYY/rlvolZ

5. Add a new line in */etc/checklist* using *vi* or *ed* for the new logical volume.

   /dev/vgYY/lvolZ /swap swap defaults 0 0

   Where:

Key	Operation
/dev/vgYY/lvolZ	The full path name of the block device file for the new swap logical volume.

6. Enable swapping on the new device.

   # swapon -a

   Where:

Key	Operation
a	The attempt to start swapping on everything listed in */etc/checklist*.

7. Verify that the new swap device is enabled.

   # swapinfo

8. If the a logical volume added is a member of the root volume group (typically vg00), update the Boot Data Reserved Area (BDRA).

   # lvlnboot -R /dev/vgYY

9. Backup the LVM data structures. The default path name for the backup will be */etc/lvmconf/vgYY.conf*.

   # vgcfgbackup /dev/vgYY

Adding Dumps Devices on a Logical Volume Cookbook

Beginning with HP-UX release 9.0, you can designate non-swap disk partition(s) or logical volume(s) for use by the kernel to write, or *dump*, an image of the core memory after a High Priority Machine Check (HPMC) or kernel panic occurrence. The new disk space dedicated to dumps is called the *dumps area*. The required dumps area can consist of one or more disk partitions or logical volumes or a combination of both.

Prior to HP-UX release 9.0, dumps were written only to designated swap devices. If the system was configured to save dump information, any bootup after a crash is forced to wait for *savecore* to finish saving the dump before swapping can be enabled. The time required to save a dump varies with the amount of physical memory and the state of the file system. Often long periods of down time result.

The key advantage of having a dumps area is faster bootup after a crash. Swapping can be enabled immediately since dump information is not in the swap area. The key disadvantage to having a dumps area is that the partitions or logical volume(s) designated as dumps area(s) CANNOT be used for any other purpose.

The system administrator needs to balance down time in the rare event of a crash against the use of disk resources. Mission critical systems are good candidates for configuring dumps areas.

Please note that it is possible to still designate swap as the target for dump information as with past revisions of HP-UX.

Note

- Dumps area(s) must reside in the root volume group (normally *vg00*). The kernel (hp-ux) must know where the dumps area(s) are to use them in a crash.
- The logical volume(s) used as dumps area must have contiguous physical extents in a logical volume, that is, the extents must be allocated in ascending order with no gaps between extents. This must be done when the logical volume(s) are created with *lvcreate*.

To configure dumps area(s), on logical volumes perform the following tasks:

1. Verify the current system dumps area configuration.

   # lvlnboot -v

   This command will display the current root, swap, and dumps devices.

2. If the dumps area has not been configured or if additional dumps area is required it will be necessary to locate or create contiguous logical volume(s) to be configured as dumps area. To determine if an existing logical volume was created in a contiguous manner use the *lvdisplay* command.

   # lvdisplay /dev/vgYY/lvolZ

   Look at the bottom line of the output. The *Allocation* field will indicate *contiguous* if the logical volume was created with contiguous physical extents. If this is not displayed, the logical volume cannot be used for dumps area.

Note

There is no way to convert a logical volume from non-contiguous to contiguous. If this is desired, the logical volume must be removed, and if contiguous space is available on the physical volume, the logical volume can be created again using the *lvcreate -C y* option as shown later in this procedure. If it is decided to use an existing logical volume, care should be taken to remove any reference to it in */etc/checklist*.

---

3. If it becomes necessary to create a contiguous logical volume for dumps area, perform the applicable steps among the first nine steps of the *Create a Logical Volume on a New Disk* cookbook, found earlier in this section. Do NOT proceed beyond step 9 of this procedure.

4. After completing the applicable first nine steps of the *Create a Logical Volume on a New Disk* cookbook, create contiguous logical volume or volumes of sufficient size (when added to any existing dumps area) to bring total dumps area size to an amount which is at least 1 MB more than installed physical memory.

lvcreate -C y -L SSS vgYY

Where:

Key	Operation
SSS	An integer number indicating the size in MB of the logical volume to be created.
YY	The integer number of the volume group the logical volume will be associated with.

5. Update the Boot Data Reserved Area (BDRA) and LIF's LABEL file with the new dumps location(s).

   # *lvlnboot -d /dev/vgYY/lvolZ*

Note Repeat the command for each logical volume to be added as a dumps device.

6. Using the UXGEN process, generate a new kernel with the following line included in the kernel devices section of the */etc/conf/gen/S800* file:

   dumps on lvol;

Note This line may have to be added or an existing line modified using *vi* or *ed*. For examples, refer to *S800—Kernel Devices* and *Kernel Device—Dumps* located in section 1 of this manual.

7. Now, backup the LVM data structures. The default path name for the backup will be */etc/lvmconf/vgYY.conf*.

   # *vgcfgbackup /dev/vgYY*

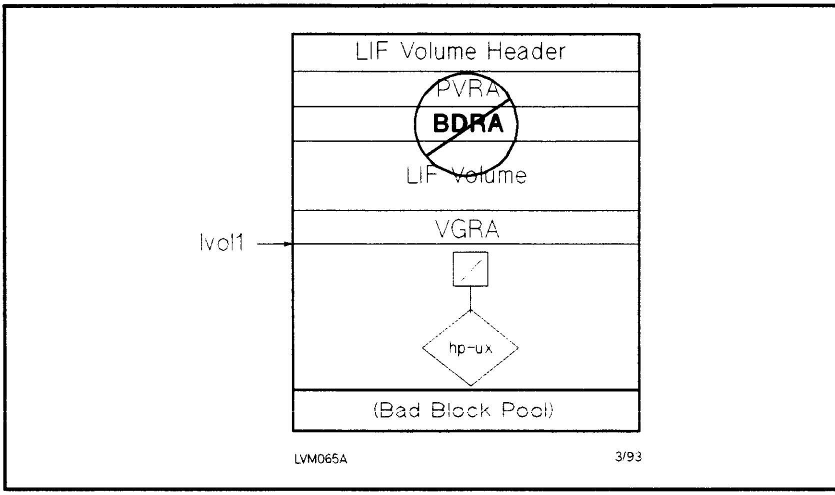
8. Configure */etc/rc* to save dumps. Follow the procedure given in the *Modifying /etc/rc to Save Memory Dumps*, located in the *Cookbooks* section of this manual.

9. Reboot the computer to load the new kernel into memory.

   # *shutdown -r 0*

The system is now configured to save core images to the dumps area.

Booting a Damaged LVM Bootable Disk Cookbook



The following procedure assumes that the Logical Interchange Format (LIF) sections of the bootable LVM disk is functioning to allow the user to load a kernel. If this is not so, use the Support Tools Media (Support Tape) to recover the system.

1. Attempt to boot the disk in LVM maintenance mode.

   ISL> *hpux -lm (BC/X.Y.Z.U;0)hp-ux* (try SYSBCKUP too)

   Where *BC/X.Y.Z.U* is the physical path to the boot disk.

   The steps that follow will not apply in all situations, nor will the order of the commands presented be applicable in all cases. The user must read and analyze error messages to arrive at a solution.

   If the following general troubleshooting suggestions fail to remedy the bootup problem, refer to *Solving HP-UX Problems*, HP Part Number B2355-90030, Chapter 8, or call HP for expert assistance.

The following assumptions will apply to the commands that follow; the root volume group is *vg00*; and the root logical volume is *lvol1*. If this is not the case, substitute your system's root volume group and root logical volume when typing the commands.

The key things to accomplish are:

a. Restore or update LVM data structures.
b. Make the root volume group available.

2. Use *vgcfgrestore* to restore a current backup of LVM data structures (PVRA, VGRA, and BDRA) to *vg00*.

   # vgcfgrestore -n vg00 /dev/rdsk/cXd0s2

   Where: *X* is the logical unit number of the root disk.

3. Attempt to make the root volume group available.

   # vgchange -a y /dev/vg00

4. The following will update LIF's LABEL file with information contained in the Boot Data Reserve Area.

   # lvlnboot -R /dev/vg00

5. Make sure all LVM disks are powered, then use *vgscan* to rebuild the */etc/lvmtab* file. Many LVM commands depend on the information contained in this file.

   # mv /etc/lvmtab /etc/lvmtab.bk
   # vgscan -v

   Once the root volume group LVM data structures have been replaced or updated and the volume group is available, the system should boot normally.

LVM Example: Mirroring a Root Disk

The following example will create a single mirror for the root and primary swap logical volumes. The commands assume the following:

1. The root volume group is *vg00*
2. The root logical volume is *lvol1*
3. The primary swap device is *lvol2*
4. The root bootable system disk is LU *0*
5. The mirror copy disk is LU *1*

The following commands will accomplish the task:

```
# pvcreate -B /dev/rdsk/c1d0s2
# mkboot -l /dev/rdsk/c1d0s2
# mkboot -a "hpux (;0)/hp-ux" /dev/rdsk/c1d0s2

# vgextend /dev/vg00 /dev/dsk/c1d0s2
# lvextend -m 1 /dev/vg00/lvol1 /dev/dsk/c1d0s2
# lvextend -m 1 /dev/vg00/lvol2 /dev/dsk/c1d0s2

# lvlnboot -r /dev/vg00/lvol1
# lvlnboot -s /dev/vg00/lvol2
# vgcfgbackup vg00
```

LVM Example: Backup Up the Mirrored Disk

The following example will split the root mirror created in the previous example, back up the data, and merge the two logical volumes back into the mirror.

Here are the commands to accomplish the backup and resynchronization:

```
# lvsplit -s backup /dev/vg00/lvol1
# fsck -p /dev/vg00/lvol1backup
# mkdir /lvol1backup
# mount /dev/vg00/lvol1backup /lvol1backup
# fbackup -f /dev/rmt/0h -0vHi /lvol1backup
# umount /lvol1backup
# lvmerge /dev/vg00/lvol1backup /dev/vg00/lvol1
# vgcfgbackup vg00
```

A

Other HP-UX Information Sources

Reference Publications

Title	HP Part Number
HP-UX Reference (3-volume set)	B2355-90033
Advanced UNIX Programming (Prentice-Hall, 1985)	92453-90007
Installing and Updating HP-UX	B3108-90006
HP-UX System Administration Tasks Manual, HP 9000 Series 800	B3105-90005
Finding HP-UX Information, HP 9000 Series 800	B3108-90001
HP-UX VI Reference Card	98597-90000
Error Message Catalog	B1862-90004
Support Tools User's Manual	5961-1612
Managing Disk Mirrors Using DataPair/800	92453-90023
Managing SwitchOver/UX	92668-90005
Solving HP-UX Problems	B2355-90030
HP GlancePlus/UX User's Manual	B2660-90001
Installing Peripherals HP 9000 Series 800 Computers	B3108-90004

GSY Information Database System

The GSY Server Mail Information Database System (GSYINFO) is available to the on-line community, and provides a way to obtain and exchange various types of information, including programs, documents, and interesting articles.

To access GSYINFO, use either:

- From HP Desk, at address *GSYINFO/HP4700*

  or

- From HP-UX systems, at address *gsyinfo@hpcugsya*

GSYINFO is easy to use, and has only a few basic rules. These are:

1. Your message *subject* can be anything you choose.
2. Commands (listed below) must be in the body (or text) portion of the message.

Commands

Command	Description
comment	All lines following this are mailed in a separate message to the Server Administrator. You may use *comment* to send Server bug reports, enhancement requests, or use this feature to submit new information to the database.
find pattern	Returns a list of files that match pattern. You may then use the *send* command to have any of these files mailed to you.
grep pattern	Returns a list of files that contain pattern in them, by doing a case-insensitive search. You may then use the *send* command to have any of these files mailed to you.

Commands, cont.

Command	Description
help	Returns help information (this list). When you receive this message from the Server you will be able to select specific help topics by entering an asterisk (*) before the pathname of the topic(s) and then mailing this message back to the Server. Your selection(s) will be mailed back to you in subsequent mail message(s). (See *send* below).
run	This command allows you to run specific programs and have the results of program execution mailed to you. Some programs may require that you send a template to the Server so that the necessary information can be provided to the program you wish to execute.
send [shar]	*index* is an alias for *send*. Returns a complete list of files that can be mailed to you, or use this command to have specific files mailed to you. When you receive the *send* message from the Server, you will be able to select specific files by entering an asterisk (*) before their pathname(s) and then mailing the message back to the Server. Your selection(s) will be mailed to you in subsequent mail message(s). If you use the [shar] option, all the files that you have requested will be "shar"ed, and then mailed to you.

Index

D

E

V

W

HEWLETT PACKARD

Printed in USA 05/93
Edition 2

Manual Part No.
5961-8364