

6

Managing the File System

In this Chapter...

- [Tools for Locating Files](#), 6-2
- [Entering the Single-User State to Perform Maintenance](#), 6-3
- [Maintaining File-System Integrity](#), 6-4
- [The Root Directory Environment Variable](#), 6-6
- [The .hp3070 File](#), 6-11
- [Managing Unused Directories and Files](#), 6-16
- [Installing Software Packages](#), 6-18
- [Managing Logical Volumes](#), 6-19

For Help

HP-UX Reference

HP-UX reference information is available on the Internet at:

<http://docs.hp.com/index.html>

System Administrator Manager (SAM)

To start SAM, enter: `/usr/sbin/sam` at a shell window prompt.

For help inside of SAM:

- From the dialog, click **Help**.
- Press F1 for context-sensitive help on a specific field.

3070 Reference

3070 User and Service manuals are located on 3070 system controllers and on factory-supplied updates.

More Help

See [In Case of Difficulty](#) on page 10-1.

Tools for Locating Files

Table 6-1 lists some file-finding shell commands and their descriptions.

Table 6-1 Some file-finding shell commands and their descriptions

Shell Command	Description
<code>find</code>	Searches the file hierarchy
<code>whereis</code>	Locates source, binaries, man pages (only)
<code>which</code>	Locates an executable in your <code>\$PATH</code>
<code>file</code>	Determines file type
<code>strings</code>	Finds printable strings in binary file
<code>type</code>	Alias for <code>whence</code> This is a utility that combines some of the features of <code>whereis</code> and <code>which</code>

Entering the Single-User State to Perform Maintenance

This section contains:

- [Introduction](#), 6-3
- [Enter to and Return from the Single-User State](#), 6-3

Introduction

Enter the single-user state to ensure no other users are on the system while maintenance tasks such as backing up the system are performed.

When the system is in the single-user state:

- The only access is through the console by the `root` user.
- The only processes running are:
 - The shell on the system console.
 - Background daemon processes started by `/sbin/rc`
 - Processes started by the **root** user.

Enter to and Return from the Single-User State

- Use SAM.

Maintaining File-System Integrity

This section contains:


- [Introduction](#), 6-4
- [fsck](#), 6-4
- [Conditions for Manually Running fsck](#), 6-4
- [Control File Access](#), 6-5

Introduction

It is possible for the file system to become corrupt in ways from which the system cannot recover. Possible causes include:

- an unanticipated power failure.
- an improper system shutdown.
- a hardware failure.

CAUTION

 A corrupted file system should always be repaired immediately. Allowing a known corrupted file system to exist can be disastrous.

`fsck`

To maintain the integrity of a file system and to repair inconsistencies, use the `fsck` command routinely.


See the man pages (enter: `man fsck`) for switches to the `fsck` command.

NOTE


If the system is rebooted after an improper shutdown (such as a power failure), `fsck` will run automatically.

Conditions for Manually Running `fsck`

CAUTION

 Run `fsck` ONLY when the system is in the single-user state and the file system is static. See [Entering the Single-User State to Perform Maintenance](#) on page 6-3.

CAUTION

 DO NOT run `fsck` on any **mounted** file system other than `/` (root).

NOTE

`fsck` can be run on **unmounted** file systems other than root.

Control File Access

Both file-access permissions and file ownership determine who can access a given file. HP-UX file-access commands are summarized in [Table 6-2](#).

Table 6-2 Some HP-UX commands to change file-access permissions

HP-UX File Access Command	What it Does
<code>chmod</code>	Changes the type of access privileges (read, write, and execute) for a file's owner or group, and for all other users. Only the file owner or superuser can change a file's read, write, and execute privileges.
<code>umask</code>	Changes default file permissions. By default, new files have read/write permissions for everyone (<code>-rw-rw-rw</code>), and new directories have read/write/execute permissions for everyone (<code>drwxrwxrwx</code>).
<code>chown</code>	Changes the ownership for a file. Only the file owner or superuser can change file ownership.
<code>chgrp</code>	Changes file group ownership. Only the file owner or superuser can change group ownership.

The Root Directory Environment Variable

This section contains:

- [Introduction](#), 6-6
- [The \\$AGILENT3070_ROOT Environment Variable](#), 6-6
- [\\$AGILENT3070_ROOT on UNIX](#), 6-6
- [\\$AGILENT3070_ROOT on MS Windows](#), 6-9

Introduction

3070 systems are now available with a choice of two operating systems:

- UNIX
- MS Windows

3070 application file path usage is different between the two operating systems.

The \$AGILENT3070_ROOT Environment Variable

Beginning with software revision 3070 04.00pa, an environment variable is used to allow 3070 board files to be easily transferred between 3070 systems running either MS Windows or UNIX.

The environment variable is **\$AGILENT3070_ROOT**. It replaces the root directory path (upper path names) on both operating systems.

All subdirectories under `/opt/hp3070/././.` will exist on UNIX systems for the foreseeable future.

NOTE

`/opt/hp3070` is replaced by **\$AGILENT3070_ROOT** on all MS Windows systems.

\$AGILENT3070_ROOT on UNIX

On UNIX systems, the value of **\$AGILENT3070_ROOT** is `/var/hp3070`

Identify the Value of \$AGILENT3070_ROOT

To identify the value of **\$AGILENT3070_ROOT** on a UNIX system, at a shell window prompt enter:

- `echo $AGILENT3070_ROOT`

File Path Usage in a UNIX terminal Window

Table 6-3 on page 6-7 illustrates path equivalents using the system config file when working in a UNIX terminal window.

Table 6-3 File path usage in a UNIX shell window

Pre 3070 Software Release 3070 04.00 pa	At and After 3070 Software Release 3070 04.00 pa
/var/hp3070/diagnostics/th1	\$AGILENT3070_ROOT/diagnostics/th1
/hp3070/diagnostics/th1	\$AGILENT3070_ROOT/diagnostics/th1

File Path Usage in a UNIX BT-BASIC (or MS Windows BT-BASIC) Window

Table 6-4 illustrates path equivalents using the system config file when working in a UNIX **BT-BASIC** (or MS Windows **BT-BASIC**) window.

NOTE

BT-BASIC usage is the same in both UNIX and MS Windows.

Table 6-4 File path usage in a UNIX BT-BASIC (or MS Windows BT-BASIC) window

Pre 3070 Software Release 3070 04.00pa	3070 Software Release 3070 05.00p	At and After 3070 Software Release 3070 04.00pa
msi "D:/Agilent3070/diagnostics/th1/config"	msi "C:/Agilent3070/diagnostics/th1/config"	msi btgetenv\$ ("AGILENT3070_ROOT") & "/diagnostics/th1"
get "D:/Agilent3070/diagnostics/th1/config"	get "C:/Agilent3070/diagnostics/th1/config"	get btgetenv\$ ("AGILENT3070_ROOT") & "/diagnostics/th1/config"

NOTE

The `btgetenv$ ("AGILENT3070_ROOT") &` is only required for BT-BASIC commands which are referenced to the root.

If the text does a BT-BASIC `msi btgetenv$ ("AGILENT3070_ROOT") & <command>` prior to the next BT-BASIC command (for example `compile` or `faon`), then using the environment variable which defines the path from the root is unnecessary. BT-BASIC commands which normally contain paths (`msi`, `load`, `copy`, `save`, `get`, `store`, `unlink`, `rcall`) for example, will require `btgetenv$ ("AGILENT3070_ROOT") & <rest of path>`

\$AGILENT3070_ROOT on MS Windows

Identify the Value of \$AGILENT3070_ROOT

To identify the value of **\$AGILENT3070_ROOT** on a MS Windows system, see *Administering 3070 MS Windows Systems* E9970-90000.

File Path Usage in a MS Windows Korn Shell Window

When working in a **Korn shell** window, follow the UNIX syntax by:

- Using `$variable` (instead of `%variable%`).
- Using the correct case.
- Using `/` (forward slash) instead of `\` (backslash).

Table 6-5 illustrates path equivalents when working in a MS Windows **Korn shell** window.

Table 6-5 File path usage in a MS Windows **Korn shell** window

Pre 3070 Software Release 3070 04.00pa	3070 Software Release 3070 05.00p	At and After 3070 Software Release 3070 04.00pa
D:/Agilent3070/diagnostics/th1	C:/Agilent3070/diagnostics/th1	\$AGILENT3070_ROOT/diagnostics/th1
/opt/hp3070/help/C/SERVICE	/opt/hp3070/help/C/SERVICE	\$AGILENT3070_ROOT/Documentation/SERVICE

File Path Usage in a MS Windows BT-BASIC or UNIX BT-BASIC Window

Table 6-4 on page 6-8 shows path equivalents to the previous path usage when working in either a MS Windows **BT-BASIC** or UNIX **BT-BASIC** window.

NOTE

BT-BASIC usage is the same in both MS Windows and UNIX.

File Path Usage in a MS-DOS Command Prompt Window

When working in a **MS-DOS Command Prompt** window:

- Use %variable% (instead of \$variable).

- Use \ (backslash) instead of / (forward slash).

Table 6-6 illustrates path equivalents using the dev directory when working in a **MS-DOS Command Prompt** window.

Table 6-6 File path usage in a **MS-DOS Command Prompt** window

Pre 3070 Software Release 3070 04.00pa	At and After 3070 Software Release 3070 04.00pa
D:\Agilent3070\dev	%AGILENT3070_ROOT%\dev

The .hp3070 File

The system first searches the current working directory for a .hp3070 file. If one is not found, the user's home directory is searched. This method allows a .hp3070 file for each board.

The .hp3070 file can affect system behavior in many ways.

Some Descriptions of .hp3070 File Keywords

Some descriptions of .hp3070 file keywords are given in [Table 6-7](#). This is not a complete description of this file. Other options are described beneath appropriate topics throughout the 3070 User documentation.

Table 6-7 Some descriptions of .hp3070 file keywords

keyword	Description
.Experience	<p>Use this option to globally restrict the user to a subset — an “experience level” — of the full system features. .Experience can be standard, advanced, or expert. An example looks like this – .Experience: advanced</p> <p>Besides the global value for an experience level, you can individually specify an experience level for software modules in a system. For example – Mpa.Experience: standard</p> <p>overrides the global default and sets the experience level for the pin assignment software to standard.</p>

NOTE

Agilent IPG Test Consultant does not make use of experience levels.

Table 6-7 Some descriptions of .hp3070 file keywords (continued)

keyword	Description
.BackupLevel	<p>The value of this option sets the global backup style for this user and determines whether the system compilers keep an unchanged copy (a backup) of files before modifying them, and how the backup is stored. The backup style can be:</p> <p>none – No file backup is made.</p> <p>numbered – Multiple backups are made as files change. To identify a numbered backup file, its name has a period, a tilde (~), and a unique number from 1 to 9 appended to its name; for example, <code>file.1~</code>. Number 1 is the most recent backup, and number 9 is the oldest. When more than 9 backups occur, the oldest backup file in the set is discarded and those remaining are renumbered.</p> <p>unnumbered – A single backup is made as files change. Each new backup file overwrites the contents of the previous backup file. To identify an unnumbered backup file, a tilde (~) is appended to its name; for example, <code>file~</code>. For example, <code>.BackupLevel: unnumbered</code></p> <p>Besides the global value for backup style, you can individually specify a backup style for some of the software modules in your system. For example, <code>Mpa.BackupLevel: numbered</code> overrides the global default and sets the backup style for the pin assignment software to numbered.</p>
.ProgramAction	<p>The value of this option determines whether a new window is automatically opened when some commands are executed (such as <code>execute</code> – see Syntax Reference. <code>.ProgramAction</code> can be either <code>window</code> (a new window is automatically opened) or <code>nowindow</code> (a new window is not automatically opened). For example, <code>.ProgramAction: nowindow</code></p>
Debug.Source	<p>The value of this option determines whether the Agilent Pushbutton Debug environment is automatically invoked when a <code>debug</code> statement is executed on the BT-BASIC command line. The value of this option can be:</p> <p>Debug.Source: no – Use the standard debug environment by default.</p> <p>Debug.Source: yes – Use the Agilent Pushbutton Debug environment by default.</p> <p>For more information, see Test Methods: Digital.</p>

Table 6-7 Some descriptions of .hp3070 file keywords (continued)

keyword	Description
FXT.WIRECOLORS	<p>This option lets you specify user-defined wire colors for fixturing. The values following this variable are the colors that are requested in fixture building reports. This lets you customize wiring reports so they ask for colors (in any language) matching the colors of the wires being used.</p> <p>The first color is used for all non-ground wiring (positive and negative) and should be the local word for red. The second color is used for all ground wiring and should be the local word for black. The remaining colors are used in sequence, one per node. The sequence of colors repeat after the last color has been used. For example, <code>FXT.WIRECOLORS: "red black blue green yellow aqua white"</code></p> <p>In the example, <code>red</code> is used for all non-ground wires, and <code>black</code> is used for all ground wires. The color of wires specified for wiring nodes cycle through the list from <code>blue</code> to <code>white</code>. After <code>white</code> has been used, the sequence starts over with <code>blue</code>.</p> <p>For more information, see Test and Fixture Development.</p>
Operator.ForceWidgets	<p>This option lets you specify whether the operator keypad appears on the screen for operator logins. Specify <code>Yes</code> to have the keypad automatically appear, or <code>No</code> to have it not appear. For example,</p> <pre>Operator.ForceWidgets: Yes</pre>
Operator.Footswitch	<p>This option lets you specify whether the foot switch on the Agilent 3070 is enabled. Specify <code>Yes</code> to have the foot switch enabled, or <code>No</code> to have it disabled. For example, <code>Operator.footswitch: Yes</code></p>

The majority of the .hp3070 file contains definitions for the operator keypad, which are invoked by an `operator` statement if the `Operator.ForceWidgets` option is set to `Yes` (see [Syntax Reference](#)). When a set of labels is specified in the `operator` statement — for example, `operator waitforstart` invokes the set of labels and functions defined as `waitforstart` — that set of label

definitions becomes active. The boxes in the operator keypad are labeled with those definitions, and selecting a box (with the mouse or the touchscreen) invokes the function associated with the label in that box.

If no label specifier is included in an `operator` statement, the default label definition is determined by the value of either of two variables in this file. The

`Operator.Default` variable sets the default for a user who is not using a board handler with the system, and the `Operator.ABH_Default` variable sets the default for a user who is. The values of both of these variables are typically defined as `standard`.

The label and function definitions are arranged into groups that each contain three specifications:

- A physical description of the operator keypad, including the X and Y coordinates and how many boxes should appear in the keypad.
- What label (text) should appear in each box. Labels can contain any combination of upper or lowercase letters and are treated as lowercase when invoked in an `operator` statement.
- Which function is invoked by selecting a particular box. Function names are case-sensitive.

For example, `standard`, which is the default definition for the operator keypad, might look like the following.

```
Standard.Boxes:      7
Standard.X:         73
Standard.Y:         50
Standard.Columns:   7

Standard.Label1:    start
Standard.Label2:    stop
Standard.Label3:    yes
Standard.Label4:    no
Standard.Label5:    faon
Standard.Label6:    faoff
Standard.Label7:    exit

Standard.Command1:  CHAR_START
Standard.Command2:  CHAR_STOP
Standard.Command3:  CHAR_YES
Standard.Command4:  CHAR_NO
Standard.Command5:  CHAR_FAON
Standard.Command6:  CHAR_FAOFF
Standard.Command7:  CHAR_EXIT
```

This example defines seven boxes, arranged in seven columns. When the operator keypad is invoked by an `operator` statement that specifies this definition, the boxes are labeled `start`, `stop`, `yes`, `no`, `faonn`, `faoff`, and `exit`. When box #1 (`Label1` or `start`) is selected, the characters `START` are executed on the command line; `start` is the function passed to BT-BASIC.

The following keywords have special meanings when they appear in definitions for the operator keypad:

- BREAK
- EXIT
- FAON
- FAOFF
- FBON
- FBOFF
- FCON
- FCOFF
- FDON
- FDOFF
- START
- STOP

In each case, selecting the label associated with the keyword passes that keyword's function to the BT-BASIC command line and executes it immediately.

Managing Unused Directories and Files

This section contains:

- [Introduction](#), 6-16
- [Remove Unneeded HP-UX Filesets](#), 6-17

Introduction

Periodically monitor the use of storage space on each hard drive. Remove unused files and unneeded HP-UX filesets to a storage media in the case it may be needed later.

With use, files will be added to the file system. If unused files are not managed, they could eventually overrun a disk drive rendering it inoperable.

Depending on the amount and type of system activity, the file system may need attention as often as daily or weekly. Use features in SAM to examine, clean up (such as trimming or removing unwanted files), and reconfigure so that the user's needs are met.

CAUTION



Truncating (trimming) a log file to zero bytes with the shell command, `>(filename)`, is always safer than removing it with `rm`. Make sure you plan to backup any files you remove before you remove them.

When managing hard drive usage, pay attention to:

- Log directories where files can accumulate unless they are manually removed. Watch the size of these directories and either delete or truncate files (after backing them up) as necessary.
- Backup files created automatically when IPG Test Consultant is run if test developers have **Backup Style** set to anything other than **No backup**.

These backup files can be named two ways:

- the filename ends with a tilde (~), such as `file~`
- the filename ends with a period, a number in the range from 1 to 9, and a tilde, such as `file.7~`

These backup files can accumulate in each board directory. If test developers are using the automatic file backup feature (especially **Numbered backup** that can save as many as nine copies of each file), consider moving unwanted backup files to a backup device to reclaim system disk space. Test developers can also manage disk space through **IPG Test Consultant**.

NOTE

For a complete list of 3070 files that should be watched, refer to the following Website (accessible with your testhead serial number), the disk full FAQ:

<http://www.agilent.com/key/boardtest>

Remove Unneeded HP-UX Filesets

The only factory-loaded filesets on the system controller that can be safely removed are the on-line documentation sets:

- Agilent 3070UD (User documentation).
- Agilent 3070SD (Service documentation).

These documentation sets can have a high reference value so you probably do not want to remove them. However, if you decide to remove the User and Service documentation:

-
- 1 **Login as `root`.**

 - 2 **Start SAM.**

 - 3 **Click Routine Tasks > Find and Remove Unneeded Filesets.**

Installing Software Packages

Introduction

The factory routinely releases new software packages that improve and add capabilities.

Some software packages depend on a previous software package being installed.

Always notify system users when a new software package is installed.

Install a Software Package

Installation instructions accompany software packages. Follow the instructions carefully.

Using Software Distributor

If you have multiple 3070 systems to update, you can designate one system as an update server from which the other systems can update.

- 1 Insert the update tape in the tape drive of the update server.
- 2 Execute on one line: `swcopy -s /dev/rmt/0m -x enforce_dependencies=false * @ /<location on disk>.`
- 3 From a system to be updated, execute on one line: `swinstall -s <servername>:/<location on disk>.`

Managing Logical Volumes

CAUTION



This section contains **unsupported** procedures:

- [Introduction](#), 6-19
- [Possible Causes for lvol Size Problems](#), 6-20
- [Resolutions That Do Not Involve Actual lvol Resizing](#), 6-20
- [Before You Begin an Unsupported Procedure to Resize a Logical Volume](#), 6-21
- [Extending all Logical Volumes Except /var, /usr, and /root](#), 6-22
- [Extending the /var and /usr Logical Volumes](#), 6-25
- [Extending the /root Logical Volume](#), 6-30

Introduction

A logical volume is a collection of disk space from one or more hard drives. Each collection appears to the operating system like a single hard drive.

Like hard drives, logical volumes can hold file systems, raw data areas, dump areas, and swap areas.

Unlike hard drives, the size of a logical volume can be adjusted after it has been created.

Logical volumes can be spread across multiple hard drives. Logical volumes exist only on hard drives; they do not exist on floppy disks, optical disks, or CD ROMs.

Logical volumes are created and managed by the Logical Volume Manager (LVM). Both LVM disks and non-LVM disks can exist simultaneously on the system, but they cannot be combined on a single disk.

The 3070 controller was preset with logical volumes when you either:

- upgraded to HP-UX 10.20 or
- purchased a new 3070 system.

In this section Agilent has provided some **unsupported** solutions for managing and resizing logical volumes when disks become overcrowded.

CAUTION



Agilent's supported and preferred solution for logical volume (**lvol**) size problems is a re-install of HP-UX.

In some cases, Agilent clients and users have requested alternatives to a full re-install. Those solutions are presented here with the stipulation that these are not supported solutions.

Possible Causes for lvol Size Problems

There are several reasons to manage information on or actually resize logical volumes. Most are common-sense causes:

- installations of newer, larger operating system versions, or
- the addition of data or software patches to existing drives.

Changes in the **lvol** layout can also affect the required size of logical volumes. For example `/opt`, which at one time was its own logical volume, was moved at B.03.40 and made part of the `/root` directory structure, causing an increase in the size of `/root`

Resolutions That Do Not Involve Actual lvol Resizing

CAUTION



The safe, supported method for solving **lvol** size problems is a complete re-install of HP-UX, during which you can specify an increase or decrease in the default logical volume sizes.

These resolutions may be used if you do not want to re-install HP-UX and you do not want to attempt the complicated resizing procedure:

- Copy the file system to another disk.

If a larger hard disk is available, copy the root file system from the original disk to the spare disk.

Refer to **Tasks You Can Perform Only with HP-UX Commands** in Chapter 5 of *Managing Systems and Workgroups* B23355-90157 for details on creating alternate boot disks.

- Use your recovery tape created with `make_recovery` (refer to man section **1m**).

This is the least time-consuming process. `make_recovery` is provided as a part of the product Ignite-UX, which is free of charge and available on the Applications Release Media or on the Internet at: <http://www.software.hp.com>

- Move a directory to another disk drive.

This option does not expand the logical volume, but should be considered as an optional solution.


- Create another logical volume with unused hard drive space.

Mount the new volume and move the directories (and their subordinate files) onto the new volume.


Create a symbolic / soft link from the new volume to the old location.

Before You Begin an Unsupported Procedure to Resize a Logical Volume


CAUTION

 This information is for experienced system administrators only. If you are an end user, do not for any reason attempt any of these procedures. Many of the non-preferred solutions detailed here could cause permanent data loss or total system failure. If you have any questions whatsoever, call your Agilent SE (system engineer) or FE (field engineer).


CAUTION

 Always do a complete system backup and a `make_recovery` before attempting any of these procedures.

CAUTION

 These instructions contain information on how to boot into single-user mode. This allows users to completely bypass all system security. Treat these instructions with care. Do not allow end users access to these instructions.

CAUTION

 For all these processes, never use `su` or `su-`. Login as `root` for all procedures.

Extending all Logical Volumes Except `/var`, `/usr`, and `/root`

CAUTION



This is an **unsupported** procedure.

CAUTION



This procedure must be done from the testhead controller. Do not attempt it from X on a PC.

CAUTION



The size of a logical volume cannot be decreased unless the disk is cleared and the procedure is started over.

This procedure is illustrated with the use of an example where the `/home` **lvol** is being extended.

Follow the instructions in [Table 6-8](#) on page 6-23 to extend any logical volume other than `/var`, `/usr`, or `/root`.

Table 6-8 Extending all logical volumes except `/var`, `/usr`, and `/root`


Task	Step
<p>1 Make sure nobody else is logged in:</p>	<p>a From the testhead controller, login as <code>root</code></p> <p>b At the <code>#</code> prompt, enter:</p> <ol style="list-style-type: none"> 1) <code>finger</code> 2) <code>fuser -cu <lv name></code> <p>where <code><lv name></code> is the logical volume name (for example, <code>/home</code>).</p> <div style="background-color: yellow; padding: 5px; text-align: center;">CAUTION</div> <p> Any process not owned by <code>root</code> will void this procedure.</p> <hr/> <p>c Did any process appear that is not owned by <code>root</code>?</p> <ul style="list-style-type: none"> ■ If no, continue with Task 2. ■ If yes: <ol style="list-style-type: none"> 1) Kill or force the closure of the process by shutting down the controller. Enter: <code>shutdown -r 0</code> 2) After the reboot, when the CDE screen re-appears, login as <code>root</code> and continue.

Table 6-8 Extending all logical volumes except `/var`, `/usr`, and `/root`

Task	Step
2 Extend the logical volume:	<ul style="list-style-type: none">a Start SAM. Enter: <code>sam</code>b Click Disks and File Systems > Logical Volumes.c Highlight the logical volume to be extended (in this case, lv015 for /home).d Click Actions Menu > Increase Size.e Noting the space available for use then choose a larger size for the logical volume.f Click OK.

Extending the `/var` and `/usr` Logical Volumes

This procedure is illustrated with the use of an example where `/dev/vg00/lvol19`, the `/var` directory, will be extended by 1 GB.

CAUTION



This is an **unsupported** procedure. Make sure you understand the cautionary statements in **Before You Begin an Unsupported Procedure to Resize a Logical Volume** on page 6-21.

CAUTION



Files critical to the operating system may reside on these logical volumes. Use extreme caution when resizing **lvols** or relocating or deleting files.

Before You Begin

Keep the following things in mind:

- This procedure may vary depending on the type of controller.
- You may see the term ISL or IPL in this example. Which of these you see on screen will also vary, depending on the type of controller.
- On a C240 or B180L system, the disk drive referenced should match the SCSI address that MTD ships for the C240s.
- On a 712, 725, or C110 with MTD pre-loaded disks, there are notes for places where you may do things differently for a dual boot system.

You must know:

- For the particular mount point (directory) (for example, `/var`, `/home`, or `/usr`), what is the block device file for the logical volume?

In this example `/var` is being increased. The logical volume block device file is

```
/dev/vg00/lvol19
```

Find information about which logical volume is associated with which mount point by running `bdf`

Make sure you write down the block device file name for each logical volume for which you want to increase the size.

- What is the physical volume (physical disk drive) for the logical volume?

Enter: `vgdisplay -v <volume group>`

where `<volume group>` is usually `vg00`

In the above example, the volume group is `vg00` because it is part of the block device file name `/dev/vg00/lvol9`

At the end of the `vgdisplay` will be about five lines describing the physical volumes for the volume group. Look for the **PV Name** field. This should be something like `/dev/dsk/c0t6d0`. This is the physical volume name that will be used later.

Also note the **Free PE** field. This tells you the number of free physical extents (1 extent = 4Mb).

CAUTION



If the value in the **Free PE** field is 0, then you cannot extend the logical volume.

Follow the instructions in [Table 6-9](#) on page 6-27 to extend any logical volume of the `/var` or `/usr` file systems.

Table 6-9 Resizing the `/var` and `/usr` logical volumes

Task	Step
<p>1 Boot into single user mode.</p>	<p>a Shut down the system using the <code>shutdown</code> or <code>reboot</code> command.</p> <p>b As the system begins to come back up, press the Esc key about once per second. You should arrive at the <code>BOOT_ADMIN</code> prompt.</p> <p>c Choose option 1 or 2 below, depending on your system:</p> <p>1) On the C240 or B180L, at the <code>BOOT_ADMIN</code> prompt, enter:</p> <pre>boot pri</pre> <p>2) When asked if you want to interact with IPL or ISL, choose Y. At the ISL (or IPL) prompt, enter:</p> <pre>hpux -is</pre> <p>This will bring up HP-UX in single user mode.</p> <p>3) On the 712, 725, or C110, with the internal disk at Unix 9.x and the external disk (SCSI.4.0) at UNIX 10.20, at the <code>BOOT_ADMIN</code> prompt, enter:</p> <pre>boot scsi.4.0 isl</pre> <p>4) At the ISL prompt, enter:</p> <pre>hpux -is /stand/vmunix</pre> <p>This will bring up HPUX 10.20 on the external disk.</p>

Table 6-9 Resizing the `/var` and `/usr` logical volumes (continued)

Task	Step
<p>NOTE</p> <p>If you get to a CDE login screen, then you have not booted into single user mode. If this is the case, start over and at the ISL prompt enter:</p> <pre>isl> hpux -is /stand/vmunix</pre> <p>When a root prompt with no login required appears, you are in single user mode.</p>	
<p>NOTE</p> <p>If you are not sure where your disk drives are, enter:</p> <pre>ls /dev/dsk</pre>	
<p>2 Use <code>lvextend</code> to extend the logical volume:</p>	<p>In this example, the logical volume is being extended from 250 MB (the default size for a 2-GB root disk) to 1.25 GB. You can specify that you want the increased disk space allocated to a specific disk (physical volume). If you do not, LVM will determine where to allocate the space.</p> <ul style="list-style-type: none"> Enter: <pre>lvextend -L <new size> <logical volume></pre> <p>For this example, the syntax would look like:</p> <pre>lvextend -L 1250 /dev/vg00/lvol9 /dev/dsk/c0t3d0</pre> <p>where the disk you are adding to is <code>/dev/dsk/c0t3d0</code></p>

Table 6-9 Resizing the `/var` and `/usr` logical volumes (continued)

Task	Step
3 Check the logical volume with <code>lvdisplay</code> :	<ul style="list-style-type: none"> Enter: <code>lvdisplay /dev/vg00/lvol9</code>
4 Use <code>extendfs</code> to extend the file system:	<p>At this point, the size of the file system is still the same, so extend it.</p> <ul style="list-style-type: none"> Enter: <code>extendfs <logical volume></code> <p>In this example, the syntax would look like:</p> <pre>extendfs -F hfs /dev/vg00/r1vol9</pre> <div style="background-color: #e0f0ff; padding: 5px; margin-top: 10px;"> <p>NOTE</p> <p>Note that the device file you extend is the character (rather than the block device file). This is somewhat subtle <code>r1vol</code> instead of <code>lv01</code>.</p> <p>If you don't specify a size (for more information, enter: <code>man extendfs</code>), this command by default will extend the filesystem to use all the available space on that logical volume.</p> </div>
5 Mount the resized volume by rebooting the system:	<p>You need to remount the filesystem. Since you are now in single user mode, the easiest way to do this is to reboot the system.</p> <ul style="list-style-type: none"> Enter: <code>reboot</code>

Extending the `/root` Logical Volume

Use the procedures in this section to solve size issues on the `/root` file system.

CAUTION



These are **unsupported** procedures.

If you choose not to do a full re-install of HP-UX, use one of the following resolutions to either manage **lvol** size or to actually resize the logical volume.

Make sure you understand the cautionary statements in **Before You Begin an Unsupported Procedure to Resize a Logical Volume** on page 6-21.

CAUTION



Performing any of the following procedures can result in severe system difficulties. Files critical to the operating system reside on `/root`, so use extreme caution when resizing this logical volume or when relocating or deleting files.

CAUTION



This is an unsupported procedure, but it does work most of the time. You assume the risk for any data loss or system malfunction. If you are uncomfortable with these cautions, or are an inexperienced end-user, do not follow this procedure. Contact your Agilent SE for on-site consultation.

Before You Begin

1 Take all the following considerations into account:

- The examples and procedures that follow assume you are running HP-UX 10.20.

NOTE

Note that HP UX 10.20 does not fit on some older 1-GB drives; they cannot be resized to make the operating system fit. In this case, contact Agilent for the preferred solution, which is an external drive that has HP-UX and the 3070 software pre-installed.

- Some of these steps require temporary access to an extra hard drive.

- Make sure you have a current `make_recovery` and full backup of each system being modified. This operation can cause permanent data loss.
- These steps cannot be used to reduce the size of a **lv**. Once you increase their size, the only way to reduce their size is to re-load the operating system, destroying all your existing data. You can use `lvreduce`, but there are limitations and restrictions on its successful operation.

2 Determine if you have enough free space on your hard drive(s).

Enter:

```
bdf -l <enter>
```

where `-l` is a hyphen/minus sign and `l` is the letter `l`.

This returns the allocated, used and free space on your existing local logical volumes.

Compare this to the physical size of your hard drive(s). If the allocated / used space is within 5% of your physical hard drive size, your hard drives are getting too full.

3 Note which logical volume(s) you want to increase.

You can now consider one of the following unsupported procedures for managing or resizing your existing logical volume(s).

4 Keep these things in mind:

- The `root`, primary swap and dump logical volumes are documented as non-extendable. Simply saying that this is a limitation of the implementation of these logical volumes is to some extent misleading. In fact, there are two reasons to disallow extension of `root`:
 - The `extendfs` (refer to man section **1m**) command can only be used on file systems that are not mounted.
 - The `/root` logical volume cannot span disks and must be contiguous.
- To extend a logical root volume, there must be free space available immediately following the logical volume on the disk. You may need a spare disk to use as your temporary root disk.
- Extending `root` can be done with just two reboots. The safest, supported method would be system reinstallation, however this does result in down time.

NOTE

It is always a good idea to keep root as small as possible (even though the 3070 installation places /opt in the /root **lvol**). Avoid using the ability to extend root to make the entire file system fit under one root file system. After system failures, large root file systems can take longer to `fsck` at reboot time and are more time consuming to change.

CAUTION



This is an **unsupported** procedure.

Make sure you understand the cautionary statements in **Before You Begin an Unsupported Procedure to Resize a Logical Volume** on page 6-21.

CAUTION



The following procedures can be **EXTREMELY** dangerous if done incorrectly, and should **NEVER** be carried out by inexperienced end-users. Be prepared to do a reinstall (full backups, etc.).

Extend the /root Logical Volume

This example adds a new LVM disk, **c1d0s2**, a **C2474S** at hardware address `52.3.0` (this will be different, based on the type of controller you are using), to the root volume group to make room on **c0d0s2**, the original root disk, then extends the original root logical volume.

Follow the instructions in **Table 6-10** on page 6-33.

Table 6-10 Extend the `/root` logical volume

Task	Step
1 Create a new bootable LVM disk.	For this example, the new disk is c1d0s2 . The syntax would look like: <pre>pvccreate -B /dev/rdisk/c1d0s2</pre>
2 Include the disk in the current volume group.	For this example, the syntax would look like: <pre>vgextend /dev/vg00 /dev/dsk/c1d0s2</pre> Verify that you have done this correctly. Enter: <pre>vgdisplay -v /dev/vg00</pre>
3 Make the disk a boot disk by adding boot utilities.	(This assumes the new disk is installed at hardware path 8/0/19/0.6.0.) <ul style="list-style-type: none"> • Enter: <ol style="list-style-type: none"> 1) <pre>mkboot /dev/rdisk/c1d0s2</pre> This places the boot utilities in the boot area. 2) <pre>mkboot -a "hpux (8/0/19/0.6.0;2)/stand/vmunix" /dev/rdisk/c1d0s2</pre> This adds an AUTO file in the boot LIF area.
4 Create a logical volume on the new disk that is the same size as the original root disk:	In this example the size should be 104 MB with 26 physical extents. <ul style="list-style-type: none"> • Enter: <ol style="list-style-type: none"> 1) <pre>lvcreate -C y -r n -n lvnew /dev/vg00</pre> 2) <pre>lvextend -l 26 /dev/vg00/lvnew /dev/dsk/c1d0s2</pre>

Table 6-10 Extend the `/root` logical volume (continued)

Task	Step
5 Create a file system in the logical volume <code>/dev/vg00/lvn:</code>	Enter: <ul style="list-style-type: none"> • <code>newfs /dev/vg00/rlvnew</code>
6 Create a mount directory for the new file system:	<ul style="list-style-type: none"> • Enter: <ol style="list-style-type: none"> 1) <code>mkdir /newroot</code> 2) <code>mount /dev/vg00/lvnew /newroot</code>
7 Move to <code>/</code> and copy the root file system over to <code>/newroot</code>	<ul style="list-style-type: none"> • Enter: <ol style="list-style-type: none"> 1) <code>cd /</code> 2) <code>find . -xdev -depth -print cpio -pxdm /newroot</code>
8 Modify BDRA (Boot Data Reserved Area) so that the system will use <code>lvnew</code> as the root file system, leaving swap and dump on <code>lvol2</code> as they were before:	<ul style="list-style-type: none"> • Enter: <ol style="list-style-type: none"> 1) <code>lvrmboot -r /dev/vg00</code> 2) <code>lvlnboot -r /dev/vg00/lvnew</code> 3) <code>lvlnboot -s /dev/vg00/lvol2</code> 4) <code>lvlnboot -d /dev/vg00/lvol2</code>
9 Move the physical volume for the swap (and dump) and <code>/usr</code> to new disk.	<ul style="list-style-type: none"> • Enter: <ol style="list-style-type: none"> 1) <code>pvmove -n /dev/vg00/lvol2 /dev/dsk/c0d0s2 /dev/dsk/c1d0s2</code> 2) <code>pvmove -n /dev/vg00/lvol3 /dev/dsk/c0d0s2 /dev/dsk/c1d0s2</code>

Table 6-10 Extend the `/root` logical volume (continued)

Task	Step
10 Verify using <code>lvdisplay</code>	<ul style="list-style-type: none"> Enter: <pre>-v /dev/vg00/lvol2 and lvdisplay -v /dev/vg00/lvol3)</pre>
11 Reboot the system from the new disk in single user mode.	<ul style="list-style-type: none"> At the ISL prompt, enter: <pre>hpux -iS (52.3.0;2)/hp-ux</pre> <p>If this fails, try to boot from the original disk using LVM maintenance mode:</p> <pre>hpux -lm</pre> <p>where <code>-</code> is a hyphen/minus sign, and <code>lm</code> are the letters <code>l</code> and <code>m</code>.</p>
12 Extend the original root logical volume:	<p>In this example, the original root logical volume <code>/dev/vg00/lvol1</code> is extended from the original 104Mb (26 extents) to 200Mb (50 logical extents).</p> <p>For this example, syntax would look like:</p> <pre>lvextend -l 50 /dev/vg00/lvol1</pre>
13 Extend the original root file system:	<p>For this example, syntax would look like:</p> <pre>extendfs /dev/vg00/rlvol1</pre>
14 Make a new directory <code>/origroot</code> and mount to it <code>/dev/vg00/lvol1</code>	<p>For this example, syntax would look like:</p> <pre>mount /dev/vg00/lvol1 /origroot</pre>

Table 6-10 Extend the /root logical volume (continued)

Task	Step
15 Run <code>fsck</code> to extend the root file system:	<p>Because the root file system is mounted, the Superblock is not in sync with the redundant Superblocks created by the <code>extendfs</code> command. Since the redundant Superblocks correctly reflect the new size of <code>root</code>, use one of them when performing <code>fsck</code>. Refer to man section 1m for more information on <code>fsck</code>.</p> <ul style="list-style-type: none"> • Enter: <code>fsck -b (alternate SB from extendfs) /dev/vg00/lvol1</code>
16 Modify BDRA to use the original root, lv01 (now larger):	<ul style="list-style-type: none"> • Enter: a <code>lvrmboot -r /dev/vg00</code> b <code>lvlnboot -r /dev/vg00/lvol1</code> c <code>lvlnboot -s /dev/vg00/lvol2</code> d <code>lvlnboot -d /dev/vg00/lvol2</code>
17 Reboot from the original root disk:	<p>(In this example, <code>/dev/dsk/c0d0s2</code>).</p> <p>At this point, <code>root</code> is now on the original disk, but is using a 200-MB root file system, with primary swap and <code>/usr</code> on the new disk. The new logical volume lvnew can now be removed with <code>lvremove</code> if you wish to reuse that space.</p>