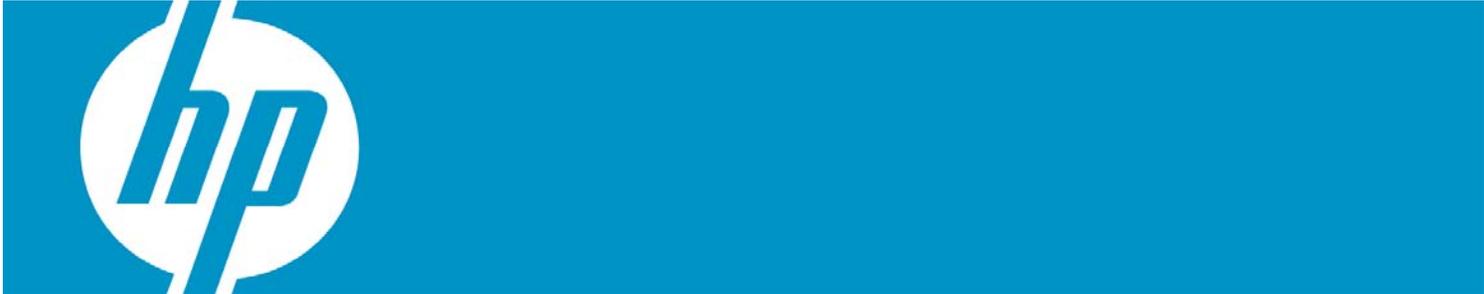


When Good Disks Go Bad: Dealing with Disk Failures Under LVM



Abstract.....	3
Background	3
1. Preparing for Disk Recovery	4
Defining a Recovery Strategy.....	4
Using Hot-Swappable Disks.....	4
Using Alternate Links (PVLlinks)	4
LVM Online Disk Replacement (LVM OLR)	5
Mirroring Critical Information, Especially the Root Volume Group	5
Creating Recovery Media	6
Other Recommendations for Optimal System Recovery	6
2. Recognizing a Failing Disk.....	9
I/O Errors in the System Log.....	9
Disk Failure Notification Messages from Diagnostics	10
LVM Command Errors.....	10
3. Confirming Disk Failure	12
4. Gathering Information About a Failing Disk	15
5. Removing the Disk	18
Removing a Mirror Copy from a Disk	18
Moving the Physical Extents to Another Disk	19
Removing the Disk from the Volume Group.....	20
Replacing a LVM Disk in an HP Serviceguard Cluster Volume Group.....	25
Disk Replacement Scenarios	25
Disk Replacement Process Flowchart.....	28
Replacing a Mirrored Nonboot Disk.....	31
Replacing an Unmirrored Nonboot Disk	33
Disk Replacement Flowchart	39
Conclusion.....	42
Appendix A: Using Device File Types.....	43
Appendix B: Device Special File Naming Model	44
New Options to Specify the DSF Naming Model	44

Behavioral Differences of Commands After Disabling the Legacy Naming Model	45
Appendix C: Volume Group Versions and LVM Configuration Files	46
Volume Group Version	46
Device Special Files	46
lvmtab, lvmtab_p	47
Appendix D: Procedures.....	48
Mirroring the Root Volume on PA-RISC Servers	48
Mirroring the Root Volume on Integrity Servers	50
Appendix E: LVM Error Messages.....	54
LVM Command Error Messages.....	54
All LVM commands.....	54
lvchange	54
lvextend	54
lvlnboot	55
pvchange	56
vgcfgbackup	56
vgcfgrestore	57
vgchange	57
vgcreate	58
vgdisplay	59
vgextend	59
vgimport	60
Syslog Error Messages.....	60
Appendix F: Moving a Root Disk to a New Disk or Another Disk.....	61
Appendix G: Recreating Volume Group Information	62
Appendix H: Disk Relocation and Recovery Using vgexport and vgimport	63
Appendix I: Splitting Mirrors to Perform Backups	65
Appendix J: Moving an Existing Root Disk to a New Hardware Path	66
For more information.....	67
Call to Action	67

Abstract

This white paper discusses how to deal with disk failures under the HP-UX Logical Volume Manager (LVM). It is intended for system administrators or operators who have experience with LVM. It includes strategies to prepare for disk failure, ways to recognize that a disk has failed, and steps to remove or replace a failed disk.

Background

Whether managing a workstation or server, your goals include minimizing system downtime and maximizing data availability. Hardware problems such as disk failures can disrupt those goals. Replacing disks can be a daunting task, given the variety of hardware features such as hot-swappable disks, and software features such as mirroring or online disk replacement you can encounter.

LVM provides features to let you maximize data availability and improve system uptime. This paper explains how you can use LVM to minimize the impact of disk failures to your system and your data. It also addresses the following topics:

- [Preparing for Disk Recovery](#): what you can do *before* a disk goes bad. This includes guidelines on logical volume and volume group organization, software features to install, and other best practices.
- [Recognizing a Failing Disk](#): how you can tell that a disk is having problems. This covers some of the error messages related to disk failure you might encounter in the system's error log, in your electronic mail, or from LVM commands.
- [Confirming Disk Failure](#): what you should check to make sure the disk is failing. This includes a simple three-step approach to validate a disk failure if you do not have online diagnostics.
- [Gathering Information About a Failing Disk](#): what you must know before you remove or replace the disk. This includes whether the disk is hot-swappable, what logical volumes are located on the disk, and what recovery options are available for the data.
- [Removing the Disk](#): how to permanently remove the disk from your LVM configuration, rather than replace it.
- [Replacing the Disk](#): how to replace a failing disk while minimizing system downtime and data loss. This section provides a high-level overview of the process and the specifics of each step. The exact procedure varies, depending on your LVM configuration and what hardware and software features you have installed, so several disk replacement scenarios are included. The section concludes with a flowchart of the disk replacement process.

You do not have to wait for a disk failure to begin preparing for failure recovery. This paper can help you be ready when a failure does occur.

1. Preparing for Disk Recovery

Forewarned is forearmed. Knowing that hard disks will fail eventually, you can take some precautionary measures to minimize your downtime, maximize your data availability, and simplify the recovery process. Consider the following guidelines before you experience a disk failure.

Defining a Recovery Strategy

As you create logical volumes, choose one of the following recovery strategies. Each choice strikes a balance between cost, data availability, and speed of data recovery.

- **Mirroring:** If you mirror a logical volume on a separate disk, the mirror copy is online and available while recovering from a disk failure. With hot-swappable disks, users will have no indication that a disk was lost.
- **Restoring from backup:** If you choose not to mirror, make sure you have a consistent backup plan for any important logical volumes. The tradeoff is that you will need fewer disks, but you will lose time while you restore data from backup media, and you will lose any data changed since your last backup.
- **Initializing from scratch:** If you do not mirror or back up a logical volume, be aware that you will lose data if the underlying hard disk fails. This can be acceptable in some cases, such as a temporary or scratch volume.

Using Hot-Swappable Disks

The hot-swap feature implies the ability to remove or add an inactive hard disk drive module to a system while power is still on and the SCSI bus is still active. In other words, you can replace or remove a hot-swappable disk from a system without turning off the power to the entire system.

Consult your system hardware manuals for information about which disks in your system are hot-swappable. Specifications for other hard disks are available in their installation manuals at <http://docs.hp.com>.

Using Alternate Links (PVLlinks)

On all supported HP-UX releases, LVM supports Alternate Links to a device to enable continuous access to the device if the primary link fails. This multiple link or multipath solution increases data availability, but does not allow the multiple paths to be used simultaneously. In such cases, the device naming model used for the representation of the mass storage devices is called the legacy naming model.

Starting with the HP-UX 11i v3 release, there is a new feature introduced in the Mass Storage Subsystem that also supports multiple paths to a device and allows access to multiple paths simultaneously. The device naming model used in this case to represent the mass storage devices is called the agile naming model. The management of the multipathed devices is available outside of LVM using the next generation mass storage stack. Agile addressing creates a single persistent DSF for each mass storage device regardless of the number of hardware paths to the disk. The mass storage stack in HP-UX 11i v3 uses this agility to provide transparent multipathing. When the new mass storage subsystem multipath behavior is enabled on the system (HP-UX 11i v3 and later), the mass storage subsystem balances the I/O load across the valid paths.

You can enable and disable the new mass storage subsystem multipath behavior and disabled through the use of the `scsimgr` command. For more information, see [scsimgr\(1M\)](#).

Starting with the HP-UX 11i v3 release, HP no longer requires or recommends that you configure LVM with alternate links. However, it is possible to maintain the traditional LVM behavior. To do so, both of the following criteria must be met:

- Only the legacy device special file naming convention is used in the LVM volume group configuration.
- The `scsimgr` command is used to disable the Mass Storage Subsystem multipath behavior.

See the following appendices for more information:

- [Appendix A](#) documents the two different types of device files supported starting with HP-UX 11i v3 release
- [Appendix B](#) documents the two different types of device special naming models supported starting HP-UX 11i v3 release

Also, see the [LVM Migration from legacy to agile naming model HP-UX 11i v3 release](#) white paper. This white paper discusses the migration of LVM volume group configurations from legacy to the agile naming model.

LVM Online Disk Replacement (LVM OLR)

LVM online disk replacement (LVM OLR) simplifies the replacement of disks under LVM. With LVM OLR, you can temporarily disable LVM use of a disk in an active volume group. Without it, you cannot keep LVM from accessing a disk unless you deactivate the volume group or remove the logical volumes on the disk.

The LVM OLR feature introduces a new option, `-a`, to `pvchange` command. The `-a` option disables or re-enables a specified path to an LVM disk. For more information on LVM OLR, see the [LVM Online Disk Replacement \(LVM OLR\)](#) white paper.

Starting with the HP-UX 11i v3 release, when the Mass Storage Subsystem multipath behavior is enabled on the system and LVM is configured with persistent device files, disabling specific paths to a device using `pvchange -a n` command does not stop I/Os to that path as they did in earlier releases because of the Mass Storage Stack native multipath functionality. Detaching an entire physical volume (all paths to the physical volume) using the `pvchange -a N` command is still available in such cases to perform Online Disk Replacement. When the Mass Storage Subsystem multipath behavior is disabled and legacy DSFs are used to configure LVM volume groups, the traditional LVM OLR behavior is maintained.

On HP-UX 11i v1 and HP-UX 11i v2 releases, LVM OLR is delivered in two patches: one patch for the kernel and one patch for the `pvchange` command.

Both command and kernel components are required to enable LVM OLR (applicable for 11i v1 and 11i v2 releases):

- For HP-UX 11i v1, install patches PHKL_31216 and PHCO_30698 or their superseding patches.
- For HP-UX 11i v2, install patches PHKL_32095 and PHCO_31709 or their superseding patches.

Note: Starting with HP-UX 11i v3, the LVM OLR feature is available as part of base operating system.

Mirroring Critical Information, Especially the Root Volume Group

By using mirror copies of the root, boot, and primary swap logical volumes on another disk, you can use the copies to keep your system in operation if any of these logical volumes fail.

Mirroring requires the add-on product HP MirrorDisk/UX (B2491BA). This is an optional product available on the HP-UX 11i application release media. To confirm that you have HP MirrorDisk/UX installed on your system, enter the `swlist` command. For example:

```
# swlist -l fileset | grep -i mirror
```

```
LVM.LVM-MIRROR-RUN B.11.23 LVM Mirror
```

The process of mirroring is usually straightforward, and can be easily accomplished using the system administration manager SAM, or with a single `lvextend` command. These processes are documented in [Managing Systems and Workgroups](#) (11i v1 and v2) and [System Administrator's Guide: Logical Volume Management](#) (11i v3). The only mirroring setup task that takes several steps is mirroring the root disk. See [Appendix D](#) for the recommended procedure to add a root disk mirror. .

There are three corollaries to the mirroring recommendation:

1. Use the strict allocation policy for all mirrored logical volumes. Strict allocation forces mirrors to occupy different disks. Without strict allocation, you can have multiple mirror copies on the same disk; if that disk fails, you will lose all your copies. To control the allocation policy, use the `-s` option with the `lvcreate` and `lvchange` commands. By default, strict allocation is enabled.
2. To improve the availability of your system, keep mirror copies of logical volumes on separate I/O buses if possible. With multiple mirror copies on the same bus, the bus controller becomes a single point of failure—if the controller fails, you lose access to all the disks on that bus, and thus access to your data. If you create physical volume groups and set the allocation policy to *PVG-strict*, LVM helps you avoid inadvertently creating multiple mirror copies on a single bus. For more information about physical volume groups, see `lvmpvg(4)`.
3. Consider using one or more free disks within each volume group as spares. If you configure a disk as a spare, then a disk failure causes LVM to reconfigure the volume group so that the spare disk takes place of the failed one. That is, all the logical volumes that were mirrored on the failed disk are automatically mirrored and resynchronized on the spare, while the logical volume remains available to users. You can then schedule the replacement of the failed disk at a time of minimal inconvenience to you and your users. Sparing is particularly useful for maintaining data redundancy when your disks are not hot-swappable, since the replacement process may have to wait until your next scheduled maintenance interval. Disk sparing is discussed in [Managing Systems and Workgroups](#) (11i v1 and v2) and [System Administrator's Guide: Logical Volume Management](#) (11i v3).

Note: The sparing feature is one where you can use a spare physical volume to replace an existing physical volume within a volume group when mirroring is in effect, in the event the existing physical volume fails. The sparing feature is available for version 1.0 volume groups (legacy volume group). Version 2.x volume groups do not support sparing.

Creating Recovery Media

Ignite/UX lets you create a consistent, reliable recovery mechanism in the event of a catastrophic failure of a system disk or root volume group. You can back up essential system data to a tape device, CD, DVD, or a network repository, and quickly recover the system configuration. While Ignite/UX is not intended to be used to back up all system data, you can use it with other data recovery applications to create a means of total system recovery.

Ignite/UX is a free add-on product, available from www.hp.com/go/softwaredepot. Documentation is available from the [Ignite/UX](#) website.

Other Recommendations for Optimal System Recovery

Here are some other recommendations, summarized from the [Managing Systems and Workgroups](#) and [System Administrator's Guide: Logical Volume Management](#) manuals that simplify recoveries after catastrophic system failures:

- Keep the number of disks in the root volume group to a minimum (no more than three), even if the root volume group is mirrored. The benefits of a small root volume group are threefold: First, fewer disks in the root volume group means less opportunities for disk failure in that group. Second, more

disks in any volume group leads to a more complex LVM configuration, which will be more difficult to recreate after a catastrophic failure. Finally, a small root volume group is quickly recovered. In some cases, you can reinstall a minimal system, restore a backup, and be back online within three hours of diagnosis and replacement of hardware.

Three disks in the root volume group are better than two due to quorum restrictions. With a two-disk root volume group, a loss of one disk can require you to override quorum to activate the volume group; if you must reboot to replace the disk, you must interrupt the boot process and use the `-lq` boot option. If you have three disks in the volume group, and they are isolated from each other such that a hardware failure only affects one of them, then failure of only one disk enables the system to maintain quorum.

- Keep your other volume groups small, if possible. Many small volume groups are preferable to a few large volume groups, for most of the same reasons mentioned previously. In addition, with a very large volume group, the impact of a single disk failure can be widespread, especially if you must deactivate the volume group. With a smaller volume group, the amount of data that is unavailable during recovery is much smaller, and you will spend less time reloading from backup. If you are moving disks between systems, it is easier to track, export, and import smaller volume groups. Several small volume groups often have better performance than a single large one. Finally, if you ever have to recreate all the disk layouts, a smaller volume group is easier to map. Consider organizing your volume groups so that the data in each volume group is dedicated to a particular task. If a disk failure makes a volume group unavailable, then only its associated task is affected during the recovery process.
- Maintain adequate documentation of your I/O and LVM configuration, specifically the outputs from the following commands:

Command	Scope	Purpose
<code>ioscan -f</code>		Print I/O configuration
<code>lvinboot -v</code>	for all volume groups	Print information on root, boot, swap, and dump logical volumes
<code>vgcfgrestore -l</code>	for all volume groups	Print volume group configuration from backup file
<code>vgdisplay -v</code>	for all logical volumes	Print volume group information, including status of logical volumes and physical volumes
<code>lvdisplay -v</code>	for all logical volumes	Print logical volume information, including mapping and status of logical extents
<code>pvdisplay -v</code>	for all physical volumes	Print physical volume information, including status of physical extents
<code>ioscan -m lun (11i v3 onwards)</code>		Print I/O configuration listing the hardware path to the disk, LUN instance, LUN hardware path and lunpath hardware path to the disk

With this information in hand, you or your HP support representative may be able to reconstruct a lost configuration, even if the LVM disks have corrupted headers. A hard copy is not required or even necessarily practical, but accessibility during recovery is important and you should plan for this.

- Make sure that your LVM configuration backups are up-to-date. Make an explicit configuration backup using the `vgcfgbackup` command immediately after importing any volume group or activating any shared volume group for the first time. Normally, LVM backs up a volume group configuration whenever you run a command to change that configuration; if an LVM command prints a warning that the `vgcfgbackup` command failed, be sure to investigate it.

While this list of preparatory actions does not keep a disk from failing, it makes it easier for you to deal with failures when they occur.

2. Recognizing a Failing Disk

The guidelines in the previous section will not prevent disk failures on your system. Assuming you follow all the recommendations, how can you tell when a disk has failed? This section explains how to look for signs that one of your disks is having problems, and how to determine which disk it is.

I/O Errors in the System Log

Often an error message in the system log file is your first indication of a disk problem. In `/var/adm/syslog/syslog.log`, you might see the following error:

HP-UX versions prior to 11.31:

```
SCSI: Request Timeout -- lbolt: 329741615, dev: 1f022000
```

To map this error message to a specific disk, look under the `/dev` directory for a device file with a device number that matches the printed value. More specifically, search for a file whose minor number matches the lower six digits of the number following `dev:`. The device number in this example is `1f022000`; its lower six digits are `022000`, so search for that value using the following command:

```
# ll /dev/*dsk | grep 022000
brw-r----- 1 bin      sys          31 0x022000 Sep 22  2002 c2t2d0
crw-r----- 1 bin      sys         188 0x022000 Sep 25  2002 c2t2d0
```

HP-UX 11.31 and later:

```
Asynchronous write failed on LUN (dev=0x3000015)
IO details : blkno : 2345, sector no : 23
```

To map this error message to a specific disk, look under the `/dev` directory for a device file with a device number that matches the printed value. More specifically, search for a file whose minor number matches the lower six digits of the number following `dev:`. The device number in this example is `3000015`; its lower six digits are `000015`, so search for that value using the following command:

```
# ll /dev/*disk | grep 000015
brw-r----- 1 bin      sys          3 0x000015 May 26 20:01 disk43
crw-r----- 1 bin      sys         23 0x000015 May 26 20:01 disk43
```

To confirm if the specific disk is under the LVM control, use the `pvdisplay -l` command. Even if the disk is not accessible but has an entry in the LVM configuration file (`/etc/lvmtab`), the `pvdisplay -l` command output is `LVM_Disk=yes` or `LVM_Disk=no` based on whether disk belongs to LVM or not, respectively.

```
# pvdisplay -l /dev/dsk/c2t2d0
/dev/dsk/c11t1d7:LVM_Disk=yes
```

This gives you a device file to use for further investigation. If it is found that the disk does not belong to LVM, see the appropriate manual pages or documentation for information on how to proceed.

The `pvdisplay` command supporting the new `-l` option, which detects whether the disk is under the LVM control or not, is delivered as part of the LVM command component in these releases:

- For HP-UX 11i v1, install patch PHCO_35313 or their superseding patches.
- For HP-UX 11i v2, install patch PHCO_34421 or their superseding patches.

Note: Starting with HP-UX 11i v3, the `-l` option to the `pvdisplay` command is available as part of the base operating system.

Disk Failure Notification Messages from Diagnostics

If you have Event Monitoring Service (EMS) hardware monitors installed on your system, and you enabled the disk monitor `disk_em`, a failing disk can trigger an event to the (EMS). Depending on how you configured EMS, you might get an email message, information in `/var/adm/syslog/syslog.log`, or messages in another log file. EMS error messages identify a hardware problem, what caused it, and what must be done to correct it. The following example is part of an error message:

```
Event Time.....: Tue Oct 26 14:06:00 2004
Severity.....: CRITICAL
Monitor.....: disk_em
Event #.....: 18
System.....: myhost
```

Summary:

Disk at hardware path 0/2/1/0.2.0 : Drive is not responding.

Description of Error:

The hardware did not respond to the request by the driver. The I/O request was not completed.

Probable Cause / Recommended Action:

The I/O request that the monitor made to this device failed because the device timed-out. Check cables, power supply, ensure the drive is powered ON, and if needed contact your HP support representative to check the drive.

For more information on EMS, see the [diagnostics section](#) on the [docs.hp.com](#) website.

LVM Command Errors

Sometimes LVM commands, such as `vgdisplay`, return an error suggesting that a disk has problems.

For example:

```
# vgdisplay -v | more
...
--- Physical volumes ---
PV Name                /dev/dsk/c0t3d0
PV Status              unavailable
Total PE               1023
Free PE                173
...
```

The physical volume status of `unavailable` indicates that LVM is having problems with the disk. You can get the same status information from `pvdisplay`.

The next two examples are warnings from `vgdisplay` and `vgchange` indicating that LVM has no contact with a disk:

```
# vgdisplay -v vg
```

```
vgdisplay: Warning: couldn't query physical volume "/dev/dsk/c0t3d0": The
specified path does not correspond to physical volume attached to this
volume group vgdisplay: Warning: couldn't query all of the physical
volumes.
```

```
# vgchange -a y /dev/vg01
```

```
vgchange: Warning: Couldn't attach to the volume group physical volume
"/dev/dsk/c0t3d0": A component of the path of the physical volume does
not exist. Volume group "/dev/vg01" has been successfully changed.
```

Another sign that you might have a disk problem is seeing stale extents in the output from `lvdisplay`. If you have stale extents on a logical volume even after running the `vgsync` or `lvsync` commands, you might have an issue with an I/O path or one of the disks used by the logical volume, but not necessarily the disk showing stale extents. For example:

```
# lvdisplay -v /dev/vg01/lvol3 | more
```

```
...
```

```
LV Status                               available/stale
```

```
...
```

```
--- Logical extents ---
```

LE	PV1	PE1	Status 1	PV2	PE2	Status 2
0000	/dev/dsk/c0t3d0	0000	current	/dev/dsk/c1t3d0	0100	current
0001	/dev/dsk/c0t3d0	0001	current	/dev/dsk/c1t3d0	0101	current
0002	/dev/dsk/c0t3d0	0002	current	/dev/dsk/c1t3d0	0102	stale
0003	/dev/dsk/c0t3d0	0003	current	/dev/dsk/c1t3d0	0103	stale

```
...
```

All LVM error messages tell you which device file is associated with the problematic disk. This is useful for the next step, confirming disk failure.

3. Confirming Disk Failure

Once you suspect a disk has failed or is failing, make certain that the suspect disk is indeed failing. Replacing or removing the incorrect disk makes the recovery process take longer. It can even cause data loss. For example, in a mirrored configuration, if you were to replace the wrong disk—the one holding the current good copy rather than the failing disk—the mirrored data on the good disk is lost.

It is also possible that the suspect disk is not failing. What seems to be a disk failure might be a hardware path failure; that is, the I/O card or cable might have failed. If a disk has multiple hardware paths, also known as *pvl*inks, one path can fail while an alternate path continues to work. For such disks, try the following steps on all paths to the disk.

If you have isolated a suspect disk, you can use hardware diagnostic tools, like Support Tools Manager, to get detailed information about it. Use these tools as your first approach to confirm disk failure. They are documented on docs.hp.com in the [diagnostics area](#). If you do not have diagnostic tools available, follow these steps to confirm that a disk has failed or is failing:

1. Use the `ioscan` command to check the S/W state of the disk. Only disks in state CLAIMED are currently accessible by the system. Disks in other states such as NO_HW or disks that are completely missing from the `ioscan` output are suspicious. If the disk is marked as CLAIMED, its controller is responding. For example:

```
# ioscan -fCdisk
Class I H/W Path Driver S/W State H/W Type Description
=====
disk 0 8/4.5.0 sdisk CLAIMED DEVICE SEAGATE ST34572WC
disk 1 8/4.8.0 sdisk UNCLAIMED UNKNOWN SEAGATE ST34572WC
disk 2 8/16/5.2.0 sdisk CLAIMED DEVICE TOSHIBA CD-ROM XM-5401TA
```

In this example, the disk at hardware path 8/4.8.0 is not accessible.

If the disk has multiple hardware paths, be sure to check all the paths.

4. You can use the `pvd`isplay command to check whether the disk is attached or not. A physical volume is considered to be *attached*, if the `pvd`isplay command is able to report a valid status (unavailable/available) for it. Otherwise, the disk is unattached. In that case, the disk was defective or inaccessible at the time the volume group was activated. For example, if `/dev/dsk/c0t5d0` is a path to a physical volume that is attached to LVM, enter:

```
# pvd
```

isplay `/dev/dsk/c0t5d0` | `grep "PV Status"`
PV Status available

If `/dev/dsk/c1t2d3` is a path to a physical volume that is detached from LVM access using a `pvchange -a n` or `pvchange -a N` command, enter:

```
# pvd
```

isplay `/dev/dsk/c1t2d3` | `grep "PV Status"`
PV Status unavailable

If the disk responds to the `ioscan` command, test it with the `diskinfo` command. The reported size must be nonzero; otherwise, the device is not ready. For example:

```
# diskinfo /dev/rdisk/c0t5d0
SCSI describe of /dev/rdisk/c0t5d0:
  vendor: SEAGATE
  product id: ST34572WC
  type: direct access
  size: 0 Kbytes
  bytes per sector: 512
```

In this example the size is 0, so the disk is malfunctioning.

- If both `ioscan` and `diskinfo` succeed, the disk might still be failing. As a final test, try to read from the disk using the `dd` command. Depending on the size of the disk, a comprehensive read can be time-consuming, so you might want to read only a portion of the disk. If the disk is functioning properly, no I/O errors are reported.

The following example shows a successful read of the first 64 megabytes of the disk: When you enter the following command, look for the solid blinking green LED on the disk:

```
# dd if=/dev/rdisk/c0t5d0 of=/dev/null bs=1024k count=64 &
64+0 records in
64+0 records out
```

Note: The previous example recommends running the `dd` command in the background (by adding `&` to the end of the command) because you do not know if the command will hang when it does the read. If the `dd` command is run in the foreground, `Ctrl+C` stops the read on the disk.

The following command shows an unsuccessful read of the whole disk:

```
# dd if=/dev/rdisk/c1t3d0 of=/dev/null bs=1024k &
dd read error: I/O error
0+0 records in 0+0 records out
```

Note: The previous example recommends running the `dd` command in background (by adding `&` at the end of the command) because you do not know if the command will hang when it does the read. If the `dd` command is run in the foreground, `Ctrl+C` stops the read on the disk.

- If the physical volume is attached but cannot be refreshed via an `lvsync`, it is likely there is a media problem at a specific location. Reading only the extents associated with the LE can help isolate the problem. Remember the stale extent might not have the problem.

The `lvsync` command starts refreshing extents at LE zero and stops if it encounters an error. Therefore, find the first LE in any logical volume that is stale and test this one. For example:

- Find the first stale LE:

```
# lvsdisplay -v /dev/vg01/lvol3 | more
```

```
.LV Status          available/stale
.
.
.
--- Logical extents ---
LE   PV1             PE1  Status 1  PV2             PE2  Status 2
0000 /dev/dsk/c0t3d0 0000 current  /dev/dsk/c1t3d0 0100 current
0001 /dev/dsk/c0t3d0 0001 current  /dev/dsk/c1t3d0 0101 current
0002 /dev/dsk/c0t3d0 0002 current  /dev/dsk/c1t3d0 0102 stale
0003 /dev/dsk/c0t3d0 0003 current  /dev/dsk/c1t3d0 0103 stale
```

In this case, LE number 2 is stale.

- Get the extent size for the VG:

```
# vgsdisplay /dev/vg01 | grep -I "PE Size"
PE size (Mbytes)      32
```

- Find the start of PE zero on each disk:

For a version 1.0 VG, enter:

```
xd -j 0x2048 -t uI -N 4 /dev/dsk/c0t3d0
```

For a version 2.x VG, enter:

```
xd -j 0x21a4 -t uI -N 4 /dev/dsk/c0t3d0
```

In this example, this is a version 1.0 VG.

```
# xd -j 0x2048 -t uI -N 4 /dev/dsk/c0t3d0  
0000000          1024  
0000004  
# xd -j 0x2048 -t uI -N 4 /dev/dsk/c1t3d0  
0000000          1024  
0000004
```

4. Calculate the location of the physical extent for each PV. Multiply the PE number by the PE size and then by 1024 to convert to Kb:

$$2 * 32 * 1024 = 65536$$

Add the offset to PE zero:

$$65536 + 1024 = 66560$$

5. Enter the following `dd` commands:

```
# dd bs=1k skip=66560 count=32768 if=/dev/rdisk/c0t3d0 of=/dev/null &  
# dd bs=1k skip=66560 count=32768 if=/dev/rdisk/c1t3d0 of=/dev/null &
```

Note the value calculated is used in the `skip` argument. The count is obtained by multiplying the PE size by 1024.

Note : The previous example recommends running the `dd` command in the background (by adding `&` at the end of the command) because you do not know if the `dd` command will hang when it does the read. If the `dd` command is run in the foreground, Ctrl+C stops the read on the disk.

4. Gathering Information About a Failing Disk

Once you know which disk is failing, you can decide how to deal with it. You can choose to remove the disk if your system does not need it, or you can choose to replace it. Before deciding on your course of action, you must gather some information to help guide you through the recovery process.

Is the questionable disk hot-swappable?

This determines whether you must power down your system to replace the disk. If you do not want to power down your system and the failing disk is not hot-swappable, the best you can do is disable LVM access to the disk.

Is it the root disk or part of the root volume group?

If the root disk is failing, the replacement process has a few extra steps to set up the boot area; in addition, you might have to boot from the mirror of the root disk if the primary root disk has failed. If a failing root disk is not mirrored, you must reinstall to the replacement disk, or recover it from an Ignite-UX backup.

To determine whether the disk is in the root volume group, enter the `lvslnboot` command with the `-v` option. It lists the disks in the root volume group, and any special volumes configured on them. For example:

```
# lvslnboot -v
Boot Definitions for Volume Group /dev/vg00:
Physical Volumes belonging in Root Volume Group:
    /dev/dsk/c0t5d0 (0/0/0/3/0.5.0) -- Boot Disk
Boot: lvol1      on:      /dev/dsk/c0t5d0
Root: lvol3      on:      /dev/dsk/c0t5d0
Swap: lvol2      on:      /dev/dsk/c0t5d0
Dump: lvol2      on:      /dev/dsk/c0t5d0, 0
```

What is the hardware path to the disk, LUN instance, LUN hardware path, and LUN hardware path to the disk?

For the HP-UX 11i v3 release (11.31) and later, when LVM is configured with persistent device files, run the `ioscan` command and note the hardware paths of the failed disk. For example:

```
# ioscan -m lun /dev/disk/disk62
Class      I  Lun H/W Path  Driver  S/W State  H/W Type  Health
Description
=====
disk      62  64000/0xfa00/0x2e  esdisk  CLAIMED   DEVICE    online
HP 73.4GST373405FC
           0/3/1/0/4/0.0x22000004cf247cb7.0x0
           0/3/1/0/4/1.0x21000004cf247cb7.0x0
           /dev/disk/disk62  /dev/rdisk/disk62
```

What recovery strategy do you have for the logical volumes on this disk?

Part of the disk removal or replacement process is based on what recovery strategy you have for the data on that disk. You can have different strategies (mirroring, restoring from backup, reinitializing from scratch) for each logical volume.

You can find the list of logical volumes using the disk with the `pvdiskdisplay` command. For example:

```
# pvdiskdisplay -v /dev/dsk/c0t5d0 | more
...
--- Distribution of physical volume ---
LV Name          LE of LV  PE for LV
/dev/vg00/lvol1  75        75
```

```

/dev/vg00/lvol2    512    512
/dev/vg00/lvol3    50     50
/dev/vg00/lvol4    50     50
/dev/vg00/lvol5    250    250
/dev/vg00/lvol6    450    450
/dev/vg00/lvol7    350    350
/dev/vg00/lvol8    1000   1000
/dev/vg00/lvol9    1000   1000
/dev/vg00/lvol10   3       3

```

...

If `pvdiskdisplay` fails, you have several options. You can refer to any configuration documentation you created in advance. Alternately, you can run `lvdisplay -v` on all the logical volumes in the volume group and see if any extents are mapped to an unavailable physical volume. The `lvdisplay` command shows '???' for the physical volume if it is unavailable.

The problem with this approach is that it is not precise if more than one disk is unavailable; to ensure that multiple simultaneous disk failures have not occurred, run `vgdisplay` to see if the active and current number of physical volumes differs by exactly one.

A third option for determining which logical volumes are on the disk is to use the `vgcfgdisplay` command. This command is available from your HP support representative.

If you have mirrored any logical volume onto a separate disk, confirm that the mirror copies are current. For each of the logical volumes affected, use `lvdisplay` to determine if the number of mirror copies is greater than zero. This verifies that the logical volume is mirrored. Then use `lvdisplay` again to determine which logical extents are mapped onto the suspect disk, and whether there is a current copy of that data on another disk. For example:

```

# lvdisplay -v /dev/vg00/lvol1
--- Logical volumes ---
LV Name           /dev/vg00/lvol1
VG Name           /dev/vg00
LV Permission     read/write
LV Status         available/syncd
Mirror copies     1
Consistency Recovery MWC
Schedule         parallel
LV Size (Mbytes)  300
Current LE       75
Allocated PE     150
Stripes          0
Stripe Size (Kbytes) 0
Bad block        off
Allocation       strict/contiguous
IO Timeout (Seconds) default

# lvdisplay -v /dev/vg00/lvol1 | grep -e /dev/dsk/c0t5d0 -e '???'
00000 /dev/dsk/c0t5d0 00000 current /dev/dsk/c2t6d0 00000 current
00001 /dev/dsk/c0t5d0 00001 current /dev/dsk/c2t6d0 00001 current
00002 /dev/dsk/c0t5d0 00002 current /dev/dsk/c2t6d0 00002 current
00003 /dev/dsk/c0t5d0 00003 current /dev/dsk/c2t6d0 00003 current
00004 /dev/dsk/c0t5d0 00004 current /dev/dsk/c2t6d0 00004 current
00005 /dev/dsk/c0t5d0 00005 current /dev/dsk/c2t6d0 00005 current

```

...

The first `lvdisplay` command output shows that `lvol1` is mirrored. In the second `lvdisplay` command output, you can see that all extents of the failing disk (in this case, `/dev/dsk/c0t5d0`) have a current copy elsewhere on the system, specifically on `/dev/dsk/c2t6d0`. If the disk `/dev/dsk/c0t5d0` is unavailable when the volume group is activated, its column contains a '???' instead of the disk name.

There might be an instance where you see that only the failed physical volume holds the current copy of a given extent (and all other mirror copies of the logical volume hold the stale data for that given extent), and LVM does not permit you to remove that physical volume from the volume group. In this case, use the `lvunstale` command (available from your HP support representative) to mark one of the mirror copies as “nonstale” for that given extent. HP recommends you use the `lvunstale` tool with caution.

With this information in hand, you can now decide how best to resolve the disk failure.

5. Removing the Disk

If you have a copy of the data on the failing disk, or you can move the data to another disk, you can choose to remove the disk from the system instead of replacing it.

Removing a Mirror Copy from a Disk

If you have a mirror copy of the data already, you can stop LVM from using the copy on the failing disk by reducing the number of mirrors. To remove the mirror copy from a specific disk, use `lvreduce`, and specify the disk from which to remove the mirror copy. For example:

```
# lvreduce -m 0 -A n /dev/vgname/lvname pvname (if you have a single mirror copy)
```

or:

```
# lvreduce -m 1 -A n /dev/vgname/lvname pvname (if you have two mirror copies)
```

The `-A n` option is used to prevent the `lvreduce` command from performing an automatic `vgcfgbackup` operation, which might hang while accessing a defective disk.

If you have only a single mirror copy and want to maintain redundancy, create a second mirror of the data on a different, functional disk, subject to the mirroring guidelines, described in [Preparing for Disk Recovery](#), before you run `lvreduce`.

You might encounter a situation where you have to remove from the volume group a failed physical volume or a physical volume that is not actually connected to the system but is still recorded in the LVM configuration file. Such a physical volume is sometimes called a *ghost disk* or *phantom disk*. You can get a ghost disk if the disk has failed before volume group activation, possibly because the system was rebooted after the failure.

A ghost disk is usually indicated by `vgdisplay` reporting more current physical volumes than active ones. Additionally, LVM commands might complain about the missing physical volumes as follows:

```
# vgdisplay vg01
vgdisplay: Warning: couldn't query physical volume "/dev/dsk/c5t5d5":
The specified path does not correspond to physical volume attached to
this volume group
vgdisplay: Couldn't query the list of physical volumes.
--- Volume groups ---
VG Name                /dev/vg01
VG Write Access        read/write
VG Status               available
Max LV                 255
Cur LV                 3
Open LV                3
Max PV                 16
Cur PV                 2 (#No. of PVs belonging to vg01)
Act PV                  1 (#No. of PVs recorded in the kernel)
Max PE per PV          4350
VGDA                    2
PE Size (Mbytes)       8
Total PE                4341
Alloc PE                4340
Free PE                 1
Total PVG               0
Total Spare PVs        0
Total Spare PVs in use 0
```

In these situations where the disk was not available at boot time, or the disk has failed before volume group activation (`pvdisplay` failed), the `lvreduce` command fails with an error that it could not query the physical volume. You can still remove the mirror copy, but you must specify the physical volume **key** rather than the name.

The physical volume key of a disk indicates its order in the volume group. The first physical volume has the key 0, the second has the key 1, and so on. This need not be the order of appearance in `/etc/lvmtab` file although it is usually like that, at least when a volume group is initially created. You can use the physical volume key to address a physical volume that is not attached to the volume group. This usually happens if it was not accessible during activation, for example, because of a hardware or configuration problem. You can obtain the key using `lvdisplay` with the `-k` option as follows:

```
# lvdisplay -v -k /dev/vg00/lvol1
...
--- Logical extents ---
  LE    PV1      PE1  Status 1 PV2      PE2  Status 2
  00000      0    00000  stale      1    00000  current
  00001      0    00001  stale      1    00001  current
  00002      0    00002  stale      1    00002  current
  00003      0    00003  stale      1    00003  current
  00004      0    00004  stale      1    00004  current
  00005      0    00005  stale      1    00005  current
...
```

Compare this output with the output of `lvdisplay` without `-k`, which you used to check the mirror status. The column that contained the failing disk (or '???') now holds the key. For this example, the key is 0. Use this key with `lvreduce` as follows:

```
# lvreduce -m 0 -A n -k /dev/vgname/lvname key (if you have a single mirror copy)
```

or:

```
# lvreduce -m 1 -A n -k /dev/vgname/lvname key (if you have two mirror copies)
```

Moving the Physical Extents to Another Disk

If the disk is marginal and you can still read from it, you can move the data onto another disk by moving the physical extents onto another disk.

The `pvmove` command moves logical volumes or certain extents of a logical volume from one physical volume to another. It is typically used to free up a disk; that is, to move all data from that physical volume so it can be removed from the volume group. In its simplest invocation, you specify the disk to free up, and LVM moves all the physical extents on that disk to any other disks in the volume group, subject to any mirroring allocation policies. For example:

```
# pvmove pvname
```

The `pvmove` command will fail if the logical volume is striped.

Note: In the September 2008 release of HP-UX 11i v3, the `pvmove` command is enhanced with several new features, including support for:

- Moving a range of physical extents
- Moving extents from the end of a physical volume
- Moving extents to a specific location on the destination physical volume
- Moving the physical extents from striped logical volumes and striped mirrored logical volumes
- A new option, `-p`, to preview physical extent movement details *without* performing the move

You can select a particular target disk or disks, if desired. For example, to move all the physical extents from `c0t5d0` to the physical volume `c0t2d0`, enter the following command:

```
# pvmove /dev/dsk/c0t5d0 /dev/dsk/c0t2d0
```

The `pvmove` command succeeds only if there is enough space on the destination physical volumes to hold all the allocated extents of the source physical volume. Before you move the extents with the `pvmove` command, check the “Total PE” field in the `pvdisplay source_pv_path` command output, and the “Free PE” field output in the `pvdisplay dest_pv_path` command output.

You can choose to move only the extents belonging to a particular logical volume. Use this option if only certain sectors on the disk are readable, or if you want to move only unmirrored logical volumes. For example, to move all physical extents of `lv014` that are located on physical volume `c0t5d0` to `c1t2d0`, enter the following command:

```
# pvmove -n /dev/vg01/lv014 /dev/dsk/c0t5d0 /dev/dsk/c1t2d0
```

Note that `pvmove` is not an atomic operation, and moves data extent by extent. If `pvmove` is abnormally terminated by a system crash or `kill -9`, the volume group can be left in an inconsistent configuration showing an additional pseudo mirror copy for the extents being moved. You can remove the extra mirror copy using the `lvreduce` command with the `-m` option on each of the affected logical volumes; there is no need to specify a disk.

Removing the Disk from the Volume Group

After the disk no longer holds any physical extents, you can use the `vgreduce` command to remove the physical volume from the volume group so it is not inadvertently used again. Check for alternate links before removing the disk, since you must remove all the paths to a multipathed disk. Use the `pvdisplay` command as follows:

```
# pvdisplay /dev/dsk/c0t5d0 --- Physical volumes ---
PV Name                /dev/dsk/c0t5d0
PV Name                /dev/dsk/c1t6d0  Alternate Link
VG Name                /dev/vg01
PV Status              available
Allocatable            yes
VGDA                   2
Cur LV                0
PE Size (Mbytes)      4
Total PE              1023
Free PE               1023
Allocated PE          0
Stale PE              0
IO Timeout (Seconds)  default
Autoswitch             On
```

In this example, there are two entries for `PV Name`. Use the `vgreduce` command to reduce each path as follows:

```
# vgreduce vgname /dev/dsk/c0t5d0
# vgreduce vgname /dev/dsk/c1t6d0
```

If the disk is unavailable, the `vgreduce` command fails. You can still forcibly reduce it, but you must then rebuild the `lvmtab`, which has two side effects. First, any deactivated volume groups are left out of the `lvmtab`, so you must manually `vgimport` them later. Second, if any multipathed disks have their link order reset, and if you arranged your `pvl` links to implement load-balancing, you might have to arrange them again.

Starting with the HP-UX 11i v3 release, there is a new feature introduced in the mass storage subsystem that also supports multiple paths to a device and allows access to the multiple paths simultaneously. If the new multi-path behavior is enabled on the system, and the imported volume

groups were configured with only persistent device special files, there is no need to arrange them again.

On releases prior to HP-UX 11i v3, you must rebuild the `lvmtab` file as follows:

```
# vgreduce -f vgname <pvname>
# mv /etc/lvmtab /etc/lvmtab.save
# vgscan -v
```

Note : Starting with 11i v3, use the following steps to rebuild the LVM configuration files (`/etc/lvmtab` or `/etc/lvmtab_p`):

```
#vgreduce -f vgname <pvname>
#vgscan -f vgname
```

In cases where the physical volume is not readable (for example, when the physical volume is unattached either because the disk failed before volume group activation or because the system has been rebooted after the disk failure), running the `vgreduce` command with the `-f` option on those physical volumes removes them from the volume group, provided no logical volumes have extents mapped on that disk. Otherwise, if the unattached physical volume is not free, `vgreduce -f` reports an extent map to identify the associated logical volumes. You must free all physical extents using `lvreduce` or `lvremove` before you can remove the physical volume with the `vgreduce` command.

This completes the procedure for removing the disk from your LVM configuration. If the disk hardware allows it, you can remove it physically from the system. Otherwise, physically remove it at the next scheduled system reboot.

7. 6. Replacing the Disk (Releases Prior to 11i v3 or When LVM Volume Group is Configured with Only Legacy DSFs on 11i v3 or Later)

If you decide to replace the disk, you must perform a five-step procedure. How you perform each step depends on the information you gathered earlier (hot-swap information, logical volume names, and recovery strategy), so this procedure varies.

This section also includes several common scenarios for disk replacement, and a flowchart summarizing the disk replacement procedure. Restore any lost data onto the disk.

The five steps are:

1. Temporarily halt LVM attempts to access the disk.
2. Physically replace the faulty disk.
3. Configure LVM information on the disk.
4. Re-enable LVM access to the disk.
5. Restore any lost data onto the disk.

In the following steps, *pvname* is the character device special file for the physical volume. This name might be `/dev/rdisk/c2t15d0` or `/dev/rdisk/c2t1d0s2`.

Step1: Halting LVM Access to the Disk

This is known as *detaching* the disk. The actions you take to detach the disk depend on whether the data is mirrored, if the LVM Online Disk Replacement functionality is available, and what applications are using the disk. In some cases (for example, if an unmirrored file system cannot be unmounted), you must shut down the system. The following list describes how to halt LVM access to the disk:

- If the disk is not hot-swappable, you must power down the system to replace it. By shutting down the system, you halt LVM access to the disk, so you can skip this step.
- If the disk contains any unmirrored logical volumes or any mirrored logical volumes without an available and current mirror copy, halt any applications and unmount any file systems using these logical volumes. This prevents the applications or file systems from writing inconsistent data over the newly restored replacement disk. For each logical volume on the disk:
 - If the logical volume is mounted as a file system, try to unmount the file system.

```
# umount /dev/vgname/lvname
```

Attempting to unmount a file system that has open files (or that contains a user's current working directory) causes the command to fail with a `Device busy` message. You can use the following procedure to determine what users and applications are causing the unmount operation to fail:

1. Use the `fuser` command to find out what applications are using the file system as follows:

```
# fuser -u /dev/vgname/lvname
```

This command displays process IDs and users with open files mounted on that logical volume, and whether it is a user's working directory.

2. Use the `ps` command to map the list of process IDs to processes, and then determine whether you can halt those processes.
3. To kill processes using the logical volume, enter the following command:

```
# fuser -ku /dev/vgname/lvname
```

4. Then try to unmount the file system again as follows:

```
# umount /dev/vgname/lvname
```

- If the logical volume is being accessed as a raw device, you can use `fuser` to find out which applications are using it. Then you can halt those applications.

If for some reason you cannot disable access to the logical volume—for example, you cannot halt an application or you cannot unmount the file system—you must shut down the system.

- If you have LVM online replacement (OLR) functionality available, detach the device using the `-a` option of the `pvchange` command:

```
# pvchange -a N pvname
```

If `pvchange` fails with a message that the `-a` option is not recognized, the LVM OLR feature is not installed.

Note: Starting with HP-UX 11i v3, the LVM OLR feature is available as part of the base operating system. Because of the mass storage stack native multipath functionality on the HP-UX 11i v3 release, disabling specific paths to a device using the `pvchange -a n` command may not stop I/Os to that path as they did in earlier releases. Detaching an entire physical volume using `pvchange -a N` is still available in order to perform an Online Disk Replacement. Use the `scsimgr` command to disable physical volume paths using the `disable` option.

- If you do not have LVM OLR functionality, LVM continues to try to access the disk as long as it is in the volume group and has always been available. You can make LVM stop accessing the disk in the following ways:
 - Remove the disk from the volume group. This means reducing any logical volumes that have mirror copies on the faulty disk so that they no longer mirror onto that disk, and reducing the disk from the disk group, as described in [Removing the Disk](#). This maximizes access to the rest of the volume group, but requires more LVM commands to modify the configuration and then recreate it on a replacement disk.
 - Deactivate the volume group. You do not have to remove and recreate any mirrors, but all data in the volume group is inaccessible during the replacement procedure.
 - Shut down the system. This halts LVM access to the disk, but makes the entire system inaccessible. Use this option only if you do not want to remove the disk from the volume group, and you cannot deactivate it.

The following recommendations are intended to maximize system uptime and access to the volume group, but you can use a stronger approach if your data and system availability requirements allow.

- If `pvdisplay` shows PV status as available, halt LVM access to the disk by removing it from the volume group.
- If `pvdisplay` shows PV status as unavailable, or if `pvdisplay` fails to print the status, use `ioscan` to determine if the disk can be accessed at all. If `ioscan` reports the disk status as `NO_HW` on all its hardware paths, you can remove the disk. If `ioscan` shows any other status, halt LVM access to the disk by deactivating the volume group.

Note: Starting with the HP-UX 11i v3 release, if the affected volume group is configured with persistent device special files, use the `ioscan -N` command, which displays output using the agile view instead of the legacy view.

Step 2: Replacing the Faulty Disk

If the disk is hot-swappable, you can replace it without powering down the system. Otherwise, power down the system before replacing the disk. For the hardware details on how to replace the disk, see the hardware administrator's guide for the system or disk array.

If you powered down the system, reboot it normally. The only exception is if you replaced a disk in the root volume group.

- If you replaced the disk that you normally boot from, the replacement disk does not contain the information needed by the boot loader. If your root disk is mirrored, boot from it by using the alternate boot path. If the root disk was not mirrored, you must reinstall or recover your system.
- If there are only two disks in the root volume group, the system might fail its quorum check and might panic early in the boot process with the "panic: LVM: Configuration failure" message. In this situation, you must override quorum to successfully boot. To do this, interrupt the boot process and add the `-lq` option to the boot command normally used by the system. The boot process and options are discussed in Chapter 5 of [Managing Systems and Workgroups](#) (11i v1 and v2) and [System Administrator's Guide: Logical Volume Management](#) (11i v3).

Step 3: Initializing the Disk for LVM

This step copies LVM configuration information onto the disk, and marks it as owned by LVM so it can subsequently be attached to the volume group.

If you replaced a mirror of the root disk on an Integrity server, run the `idisk` command as described in step 1 of [Appendix D: Mirroring the Root Volume on Integrity Servers](#). For PA-RISC servers or non-root disks, this step is unnecessary.

For any replaced disk, restore LVM configuration information to the disk using the `vgcfgrestore` command as follows:

```
# vgcfgrestore -n vgname pvname
```

If you cannot use the `vgcfgrestore` command to write the original LVM header back to the new disk because a valid LVM configuration backup file (`/etc/lvmconf/vgXX.conf [.old]`) is missing or corrupted, you must remove the physical volume that is being restored from the volume group (by using the `vgreduce` command) to get a clean configuration.

Note: In these situations the `vgcfgrestore` command might fail to restore the LVM header, issuing a 'Mismatch between the backup file and the running kernel' message. If you are sure that your backup is valid, you can override this check by using the `-R` option. To remove a physical volume from a volume group, you must first free it by removing all of the logical extents. If the logical volumes on such a disk are not mirrored, the data is lost anyway. If it is mirrored, you must reduce the mirror before removing the physical volume.

Step 4: Re-enabling LVM Access to the Disk

The process in this step is known as *attaching* the disk. The action you take here depends on whether LVM OLR is available.

If you have LVM OLR on your system, attach the device by entering the `pvchange` command with the `-a` and `y` options as follows:

```
# pvchange -a y pvname
```

After LVM processes the `pvchange` command, it resumes using the device if possible.

If you do not have LVM OLR on your system, or you want to ensure that any alternate links are attached, enter the `vgchange` command with the `-a` and `y` options to activate the volume group and bring any detached devices online:

```
# vgchange -a y vgname
```

The `vgchange` command attaches all paths for all disks in the volume group, and automatically resumes recovering any unattached failed disks in the volume group. Therefore, only run `vgchange` after all work has been completed on all disks and paths in the volume group, and it is desirable to attach them all.

Step 5: Restoring Lost Data to the Disk

This final step can be a straightforward resynchronization for mirrored configurations, or a recovery of data from backup media.

- If a mirror of the root disk was replaced, initialize its boot information as follows:
 - For an Integrity server, follow steps 5, 6, and 8 in [Appendix D: Mirroring the Root Volume on Integrity Servers](#).
 - For a PA-RISC server, follow steps 4, 5, and 7 in [Appendix D: Mirroring the Root Volume on PA-RISC Servers](#).
- If all the data on the replaced disk was mirrored, you do not have to do anything; LVM automatically synchronizes the data on the disk with the other mirror copies of the data.
- If the disk contained any unmirrored logical volumes (or mirrored logical volumes that did not have a current copy on the system), restore the data from backup, mount the file systems, and restart any applications you halted in step 1.

Replacing a LVM Disk in an HP Serviceguard Cluster Volume Group

Replacing LVM disks in an HP Serviceguard cluster follows the same procedure described in steps 1-5, unless the volume group is shared. If the volume group is shared, make the following changes:

- When disabling LVM access to the disk, perform any online disk replacement steps individually on each cluster node sharing the volume group. If you do not have LVM OLR, and you detach the disk, you might need to make configuration changes that require you to deactivate the volume group on all cluster nodes. However, if you have Shared LVM Single Node Online Volume Reconfiguration (SNOR) installed, you can leave the volume group activated on one of the cluster nodes.
- When re-enabling LVM access, activate the physical volume on each cluster node sharing the volume group.

Special care is required when performing a Serviceguard rolling upgrade. For details, see the [LVM Online Disk Replacement \(LVM OLR\)](#) white paper.

Disk Replacement Scenarios

The following scenarios show several LVM disk replacement examples.

Scenario 1: Best Case

For this example, you have followed all the guidelines in [Section 1: Preparing for Disk Recovery](#): all disks are hot-swappable, all logical volumes are mirrored, and LVM OLR functionality is available on the system. In this case, you can detach the disk using the `pvchange` command, replace it, reattach it, and let LVM mirroring synchronize the logical volumes, all while the system remains up.

For this example, you assume that the bad disk is at hardware path `2/0/7.15.0` and has device special files named `/dev/rdisk/c2t15d0` and `/dev/dsk/c2t15d0`.

Check that the disk is not in the root volume group, and that all logical volumes on the bad disk are mirrored with a current copy available. Enter the following commands:

```
# lvlnboot -v
Boot Definitions for Volume Group /dev/vg00:
Physical Volumes belonging in Root Volume Group:
    /dev/dsk/c0t5d0 (0/0/0/3/0.5.0) -- Boot Disk
Boot: lv011      on:      /dev/dsk/c0t5d0
Root: lv013      on:      /dev/dsk/c0t5d0
```

```
Swap: lvol2      on:      /dev/dsk/c0t5d0
Dump: lvol2      on:      /dev/dsk/c0t5d0, 0
```

```
# pvdisplay -v /dev/dsk/c2t15d0 | more
```

```
...
--- Distribution of physical volume ---
LV Name          LE of LV  PE for LV
/dev/vg01/lvol1  4340     4340
...
```

```
# lvdisplay -v /dev/vg01/lvol1 | grep "Mirror copies"
Mirror copies          1
```

```
# lvdisplay -v /dev/vg01/lvol1 | grep -e /dev/dsk/c2t15d0 -e '???' | more
00000 /dev/dsk/c2t15d0 00000 current /dev/dsk/c5t15d0 00000 current
00001 /dev/dsk/c2t15d0 00001 current /dev/dsk/c5t15d0 00001 current
00002 /dev/dsk/c2t15d0 00002 current /dev/dsk/c5t15d0 00002 current
00003 /dev/dsk/c2t15d0 00003 current /dev/dsk/c5t15d0 00003 current
...
```

The `lvlnboot` command confirms that the disk is not in the root volume group. The `pvdisplay` command shows which logical volumes are on the disk. The `lvdisplay` command shows that all data in the logical volume has a current mirror copy on another disk. Enter the following commands to continue with the disk replacement:

```
# pvchange -a N /dev/dsk/c2t15d0
# <replace the hot-swappable disk>
# vgcfgrestore -n vg01 /dev/rdisk/c2t15d0
# vgchange -a y vg01
```

Scenario 2: No Mirroring and No LVM Online Replacement

In this example, the disk is still hot-swappable, but there are unmirrored logical volumes and the LVM OLR functionality is enabled on the system or not. Disabling LVM access to the logical volumes is more complicated, since you must find out what processes are using them.

The bad disk is represented by device special file `/dev/dsk/c2t2d0`. Enter the following commands:

```
# lvlnboot -v
Boot Definitions for Volume Group /dev/vg00:
Physical Volumes belonging in Root Volume Group:
    /dev/dsk/c0t5d0 (0/0/0/3/0.5.0) -- Boot Disk
Boot: lvol1      on:      /dev/dsk/c0t5d0
Root: lvol3      on:      /dev/dsk/c0t5d0
Swap: lvol2      on:      /dev/dsk/c0t5d0
Dump: lvol2      on:      /dev/dsk/c0t5d0, 0
```

```
# pvdisplay -v /dev/dsk/c2t2d0 | more
```

```
...
--- Distribution of physical volume ---
LV Name          LE of LV  PE for LV
/dev/vg01/lvol1  4340     4340
...
```

```
# lvdisplay -v /dev/vg01/lvol1 | grep "Mirror copies"
Mirror copies          0
```

This confirms that the logical volume is not mirrored, and it is not in the root volume group. As system administrator, you know that the logical volume is a mounted file system. To disable access to the logical volume, try to unmount it. Use the `fuser` command to isolate and terminate processes using the file system, if necessary. Enter the following commands:

```

# umount /dev/vg01/lvol1
umount: cannot unmount /dump : Device busy
# fuser -u /dev/vg01/lvol1
/dev/vg01/lvol1: 27815c(root) 27184c(root)
# ps -fp27815 -p27184
    UID    PID  PPID  C   STIME TTY      TIME COMMAND
    root 27815 27184  0 09:04:05 pts/0    0:00 vi test.c
    root 27184 27182  0 08:26:24 pts/0    0:00 -sh
# fuser -ku /dev/vg01/lvol1
/dev/vg01/lvol1: 27815c(root) 27184c(root)
# umount /dev/vg01/lvol1

```

For this example, it is assumed that you are permitted to halt access to the entire volume group while you recover the disk. Use `vgchange` to deactivate the volume group and stop LVM from accessing the disk:

```
# vgchange -a n vg01
```

Proceed with the disk replacement and recover data from backup:

```

# <replace the hot-swappable disk>
# vgcfgrestore -n vg01 /dev/rdisk/c2t2d0
# vgchange -a y vg01
# newfs [options] /dev/vg01/rlvol1
# mount /dev/vg01/lvol1 /dump
# <restore the file system from backup>

```

Scenario 3: No Hot-Swappable Disk

In this example, the disk is not hot-swappable, so you must reboot the system to replace it. Once again, the bad disk is represented by device special file `/dev/dsk/c2t2d0`. Enter the following commands:

```

# lvinboot -v
Boot Definitions for Volume Group /dev/vg00:
Physical Volumes belonging in Root Volume Group:
    /dev/dsk/c0t5d0 (0/0/0/3/0.5.0) -- Boot Disk
Boot: lvol1    on:    /dev/dsk/c0t5d0
Root: lvol3    on:    /dev/dsk/c0t5d0
Swap: lvol2    on:    /dev/dsk/c0t5d0
Dump: lvol2    on:    /dev/dsk/c0t5d0, 0

# pvdisplay -v /dev/dsk/c2t2d0 | more
...
--- Distribution of physical volume ---
LV Name      LE of LV  PE for LV
/dev/vg01/lvol1  4340     4340
...

# lvdisplay -v /dev/vg01/lvol1 | grep "Mirror copies"
Mirror copies          0

```

This confirms that the logical volume is not mirrored, and it is not in the root volume group. Shutting down the system disables access to the disk, so you do not need to determine who is using the logical volume.

```

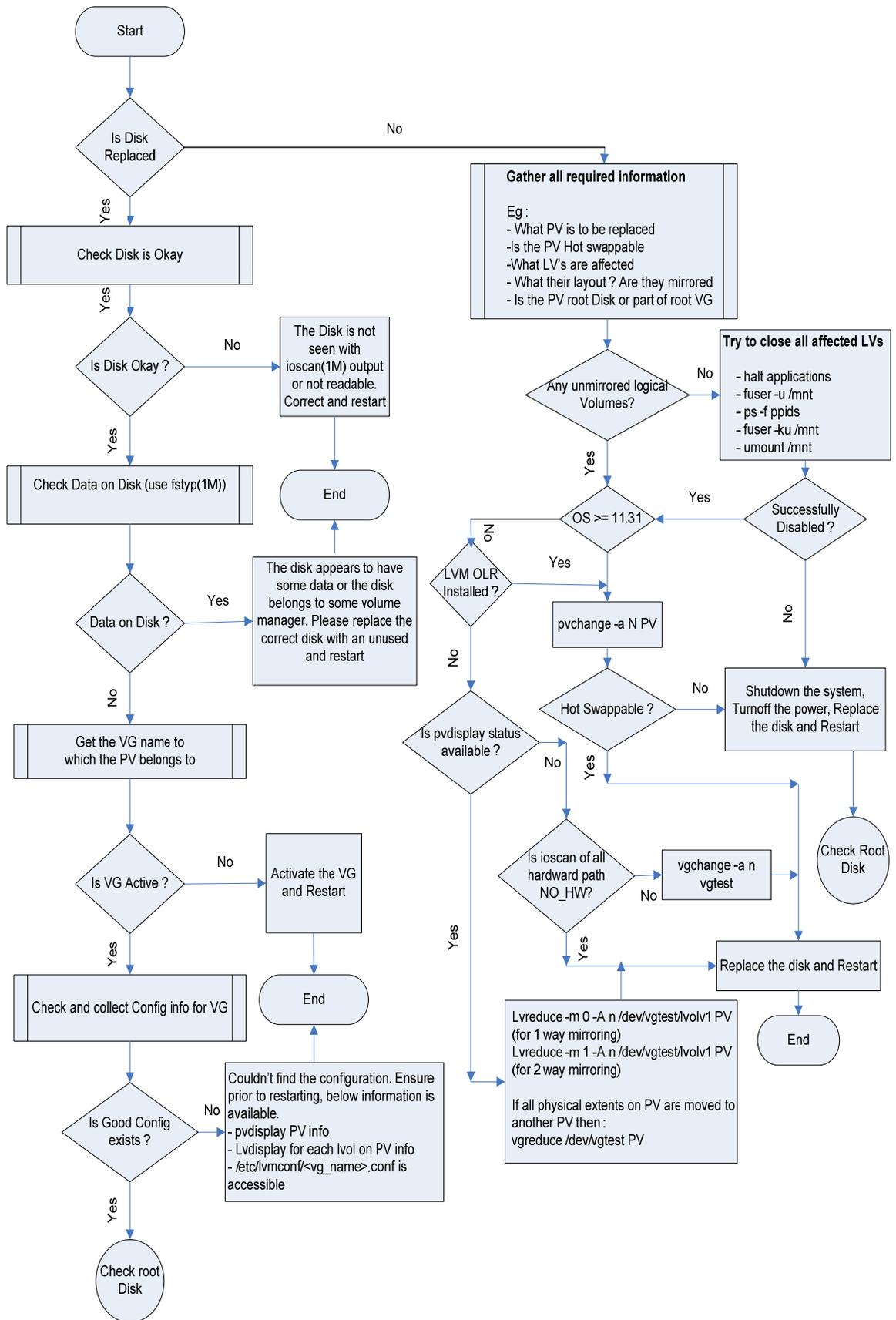
# shutdown -h
# <replace the disk>
# <reboot normally>
# vgcfgrestore -n vg01 /dev/rdisk/c2t2d0
# vgchange -a y vg01

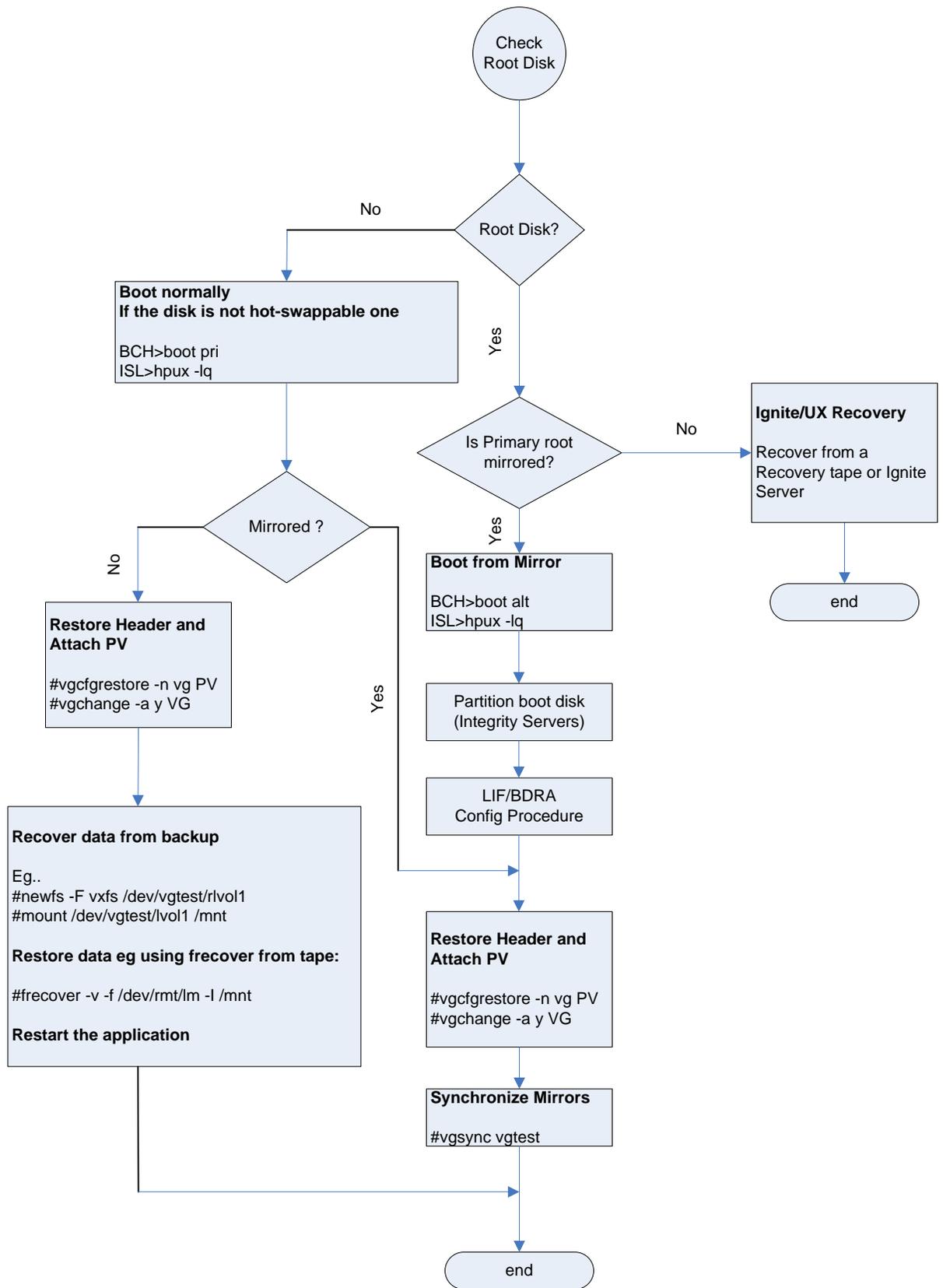
```

```
# newfs [options] /dev/vg01/rlvol1  
# mount /dev/vg01/lvol1 /app  
# <restore the file system from backup>
```

Disk Replacement Process Flowchart

The following flowchart summarizes the disk replacement process.





8. 7. Replacing the Disk (11i v3 release Onwards when the LVM Volume Group is Configured with Persistent DSFs)

After you isolate a failed disk, the replacement process depends on answers to the following questions:

- Is the disk hot-swappable?
- Is the disk the root disk or part of the root volume group?
- What logical volumes are on the disk, and are they mirrored?

Based on the gathered information, choose the appropriate procedure.

Replacing a Mirrored Nonboot Disk

Use this procedure if *all* the physical extents on the disk have copies on another disk, and the disk is *not* a boot disk. If the disk contains any unmirrored logical volumes or any mirrored logical volumes without an available and current mirror copy, see [Replacing an Unmirrored Nonboot Disk](#).

For this example, the disk to be replaced is at LUN hardware path 0/1/1/1.0x3.0x0, with device special files named `/dev/disk/disk14` and `/dev/rdisk/disk14`. Follow these steps:

1. Save the hardware paths to the disk.

Run the `ioscan` command and note the hardware paths of the failed disk.

```
# ioscan -m lun /dev/disk/disk14
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x0  esdisk CLAIMED  DEVICE  offline HP MSA Vol
          0/1/1/1.0x3.0x0
/dev/disk/disk14      /dev/rdisk/disk14
```

In this example, the LUN instance number is 14, the lunpath hardware path is 64000/0xfa00/0x0, and the lunpath hardware path is 0/1/1/1.0x3.0x0.

When the failed disk is replaced, a new LUN instance and LUN hardware path are created. To identify the disk after it is replaced, you must use the lunpath hardware path (0/1/1/1.0x3.0x0).

2. Halt LVM access to the disk.

If the disk is not hot-swappable, power off the system to replace it. By shutting down the system, you halt LVM access to the disk, so you can skip this step.

If the disk is hot-swappable, detach it using the `-a` option of the `pvchange` command:

```
# pvchange -a N /dev/disk/disk14
```

3. Replace the disk.

For the hardware details on how to replace the disk, see the hardware administrator's guide for the system or disk array.

If the disk is hot-swappable, replace it. If the disk is not hot-swappable, shut down the system, turn off the power, and replace the disk. Reboot the system.

4. Notify the mass storage subsystem that the disk has been replaced.

If the system was not rebooted to replace the failed disk, run `scsimgr` before using the new disk as a replacement for the old disk. For example:

```
# scsimgr replace_wwid -D /dev/rdisk/disk14
```

This command lets the storage subsystem replace the old disk's LUN World-Wide-Identifier (WWID) with the new disk's LUN WWID. The storage subsystem creates a new LUN instance and new device special files for the replacement disk.

5. Determine the new LUN instance number for the replacement disk. For example:

```
# ioscan -m lun
```

```
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x0    esdisk  NO_HW     DEVICE   offline HP MSA Vol
                /dev/disk/disk14      /dev/rdisk/disk14
...
disk 28 64000/0xfa00/0x1c    esdisk  CLAIMED   DEVICE   online  HP MSA Vol
                0/1/1/1.0x3.0x0
                /dev/disk/disk28      /dev/rdisk/disk28
```

In this example, LUN instance 28 was created for the new disk, with LUN hardware path 64000/0xfa00/0x1c, device special files /dev/disk/disk28 and /dev/rdisk/disk28, at the same lunpath hardware path as the old disk, 0/1/1/1.0x3.0x0. The old LUN instance 14 for the old disk now has no lunpath associated with it.

Note: If the system was rebooted to replace the failed disk, running `ioscan -m lun` does not display the old disk.

6. Assign the old instance number to the replacement disk. For example:

```
# io_redirect_dsf -d /dev/disk/disk14 -n /dev/disk/disk28
```

This assigns the old LUN instance number (14) to the replacement disk. In addition, the device special files for the new disk are renamed to be consistent with the old LUN instance number. The following `ioscan -m lun` output shows the result:

```
# ioscan -m lun /dev/disk/disk14
```

```
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x1c    esdisk  CLAIMED   DEVICE   online  HP MSA Vol
                0/1/1/1.0x3.0x0
                /dev/disk/disk14      /dev/rdisk/disk14
```

The LUN representation of the old disk with LUN hardware path 64000/0xfa00/0x0 was removed. The LUN representation of the new disk with LUN hardware path 64000/0xfa00/0x1c was reassigned from LUN instance 28 to LUN instance 14 and its device special files were renamed as /dev/disk/disk14 and /dev/rdisk/disk14.

7. Restore LVM configuration information to the disk. For example:

```
# vgcfgrestore -n /dev/vgmn /dev/rdisk/disk14
```

8. Restore LVM access to the disk.

If you did *not* reboot the system in step 2, reattach the disk as follows:

```
# pvchange -a y /dev/disk/disk14
```

If you did reboot the system, reattach the disk by reactivating the volume group as follows:

```
# vgchange -a y /dev/vgmn
```

Note: The `vgchange` command with the `-a y` option can be run on a volume group that is deactivated or already activated. It attaches all paths for all disks in the volume group and resumes automatically recovering any disks in the volume group that had been offline or any disks in the volume group that were replaced. Therefore, run `vgchange` only after all work has been completed on all disks and paths in the volume group, and it is necessary to attach them all.

Because all the data on the replaced disk was mirrored, you do not need to do anything else; LVM automatically synchronizes the data on the disk with the other mirror copies of the data.

Replacing an Unmirrored Nonboot Disk

Use this procedure if *any* of the physical extents on the disk do not have mirror copies elsewhere, and your disk is not a boot disk.

In this example, the disk to be replaced is at lunpath hardware path `0/1/1/1.0x3.0x0`, with device special files named `/dev/disk/disk14` and `/dev/rdisk/disk14`. Follow these steps:

1. Save the hardware paths to the disk.

Enter the `ioscan` command and note the hardware paths of the failed disk:

```
# ioscan -m lun /dev/disk/disk14
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x0  esdisk CLAIMED  DEVICE  offline HP MSA Vol
        0/1/1/1.0x3.0x0
                /dev/disk/disk14      /dev/rdisk/disk14
```

In this example, the LUN instance number is 14, the LUN hardware path is `64000/0xfa00/0x0`, and the lunpath hardware path is `0/1/1/1.0x3.0x0`.

When the failed disk is replaced, a new LUN instance and LUN hardware path are created. To identify the disk after it is replaced, you must use the lunpath hardware path (`0/1/1/1.0x3.0x0`).

2. Halt LVM access to the disk.

If the disk is not hot-swappable, power off the system to replace it. By shutting down the system, you halt LVM access to the disk, so you can skip this step. If the disk is hot-swappable, disable user and LVM access to all unmirrored logical volumes.

First, disable *user* access to all unmirrored logical volumes. Halt any applications and unmount any file systems using these logical volumes. This prevents the applications or file systems from writing inconsistent data over the newly restored replacement disk.

For each unmirrored logical volume using the disk:

- a. Use the `fuser` command to make sure no one is accessing the logical volume, either as a raw device or as a file system. If users have files open in the file system or it is their current working directory, `fuser` reports their process IDs.

For example, if the logical volume was `/dev/vg01/lvol1`, enter:

```
# fuser -cu dev/vg01/lvol1
/dev/vg01/lvol1: 27815c (root) 27184c (root)
```

- b. If `fuser` reports process IDs using the logical volume, use the `ps` command to map the list of process IDs to processes, and determine whether you can halt those processes. For example, look up processes 27815 and 27184 as follows:

```
# ps -fp27815 -p27184
UID  PID  PPID  C   STIME TTY      TIME COMMAND
root 27815 27184  0   09:04:05 pts/0    0:00 vi test.c
root 27184 27182  0   08:26:24 pts/0    0:00 -sh
```

- c. If so, use `fuser` with the `-k` option to kill all processes accessing the logical volume.

The example processes are noncritical, so kill them as follows:

```
# fuser -ku dev/vg01/lvol1
/dev/vg01/lvol1: 27815c(root) 27184c(root)
```

- d. If the logical volume is being used as a file system, unmount it as follows:

```
# umount /dev/vg01/lvol1
```

Note: If you cannot stop the applications using the logical volume, or you cannot unmount the file system, you must shut down the system.

After disabling user access to the unmirrored logical volumes, disable LVM access to the disk:

```
# pvchange -a N /dev/disk/disk14
```

3. Replace the disk.

For the hardware details on how to replace the disk, see the hardware administrator's guide for the system or disk array.

If the disk is hot-swappable, replace it.

If the disk is not hot-swappable, shut down the system, turn off the power, and replace the disk. Reboot the system.

Notify the mass storage subsystem that the disk has been replaced.

If the system was not rebooted to replace the failed disk, run `scsimgr` before using the new disk as a replacement for the old disk. For example:

```
# scsimgr replace_wwid -D /dev/rdisk/disk14
```

This command lets the storage subsystem replace the old disk's LUN World-Wide-Identifier (WWID) with the new disk's LUN WWID. The storage subsystem creates a new LUN instance and new device special files for the replacement disk.

4. Determine the new LUN instance number for the replacement disk. For example:

```
# ioscan -m lun
Class I Lun H/W Path      Driver  S/W State H/W Type Health Description
=====
disk 14 64000/0xfa00/0x0  esdisk NO_HW      DEVICE  offline HP MSA Vol
                /dev/disk/disk14      /dev/rdisk/disk14
...
disk 28 64000/0xfa00/0x1c esdisk CLAIMED   DEVICE  online  HP MSA Vol
                0/1/1/1.0x3.0x0
                /dev/disk/disk28      /dev/rdisk/disk28
```

In this example, LUN instance 28 was created for the new disk, with LUN hardware path 64000/0xfa00/0x1c, device special files /dev/disk/disk28 and /dev/rdisk/disk28, at the same lunpath hardware path as the old disk, 0/1/1/1.0x3.0x0. The old LUN instance 14 for the old disk now has no lunpath associated with it.

Note: If the system was rebooted to replace the failed disk, `ioscan -m lun` does not display the old disk.

5. Assign the old instance number to the replacement disk. For example:

```
# io_redirect_dsf -d /dev/disk/disk14 -n /dev/disk/disk28
```

This assigns the old LUN instance number (14) to the replacement disk. In addition, the device special files for the new disk are renamed to be consistent with the old LUN instance number.

The following `ioscan -m lun` output shows the result:

```
# ioscan -m lun /dev/disk/disk14
```

```
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x1c esdisk CLAIMED  DEVICE  online  HP MSA Vol
          0/1/1/1.0x3.0x0
          /dev/disk/disk14      /dev/rdisk/disk14
```

The LUN representation of the old disk with LUN hardware path 64000/0xfa00/0x0 was removed. The LUN representation of the new disk with LUN hardware path 64000/0xfa00/0x1c was reassigned from LUN instance 28 to LUN instance 14 and its device special files were renamed as /dev/disk/disk14 and /dev/rdisk/disk14.

6. Restore LVM configuration information to the new disk. For example:

```
# vgcfgrestore -n /dev/vgmn /dev/rdisk/disk14
```

7. Restore LVM access to the disk.

If you did *not* reboot the system in [step 2](#), reattach the disk as follows:

```
# pvchange -a y /dev/disk/disk14
```

If you did reboot the system, reattach the disk by reactivating the volume group as follows:

```
# vgchange -a y /dev/vgmn
```

NOTE: You can run the `vgchange` command with the `-a y` option on a volume group that is deactivated or already activated. It attaches all paths for all disks in the volume group and resumes automatically recovering any disks in the volume group that were offline or any disks in the volume group that were replaced. Therefore, run `vgchange` only after all work has been completed on all disks and paths in the volume group, and it is necessary to attach them all.

8. Recover any lost data.

LVM recovers all the mirrored logical volumes on the disk, and starts that recovery when the volume group is activated.

For all the unmirrored logical volumes that you identified in [step 2](#), restore the data from backup and re-enable user access as follows:

- For raw volumes, restore the full raw volume using the utility that was used to create your backup. Then restart the application.
- For file systems, you must recreate the file systems first. For example:

```
# newfs -F fstype /dev/vgmn/rlvolnn
```

Use the logical volume's character device file for the `newfs` command. For file systems that had nondefault configurations, see [newfs\(1M\)](#) for the correct options.

After creating the file system, mount it under the mount point that it previously occupied. Then restore the data for that file system from your full backups.

Tip: To make the file system recreation step easier, record how they were originally created. You can change other file system parameters, such as those used to tune file system performance. The file system must be at least as large as before the disk failure.

Replacing a Mirrored Boot Disk

There are two additional operations you must perform when replacing a mirrored boot disk:

1. You must initialize boot information on the replacement disk.
2. If the replacement requires rebooting the system, and the primary boot disk is being replaced, you must boot from the alternate boot disk.

In this example, the disk to be replaced is at lunpath hardware path `0/1/1/1.0x3.0x0`, with device special files named `/dev/disk/disk14` and `/dev/rdisk/disk14`. The system is an HP Integrity server, so the physical volume names must specify the HP-UX partition on the boot disk (`/dev/disk/disk14_p2` and `/dev/disk/disk14_p2`).

1. Save the hardware paths to the disk.

Run the `ioscan` command and note the hardware paths of the failed disk as follows:

```
# ioscan -m lun /dev/disk/disk14
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x0    esdisk CLAIMED  DEVICE  offline HP MSA Vol
          0/1/1/1.0x3.0x0
                /dev/disk/disk14      /dev/rdisk/disk14
                /dev/disk/disk14_p1 /dev/rdisk/disk14_p1
                /dev/disk/disk14_p2 /dev/rdisk/disk14_p2
                /dev/disk/disk14_p3 /dev/rdisk/disk14_p3
```

In this example, the LUN instance number is 14, the LUN hardware path is `64000/0xfa00/0x0`, and the lunpath hardware path is `0/1/1/1.0x3.0x0`.

When the failed disk is replaced, a new LUN instance and LUN hardware path are created. To identify the disk after it is replaced, you must use the lunpath hardware path (`0/1/1/1.0x3.0x0`).

2. Halt LVM access to the disk.

If the disk is not hot-swappable, power off the system to replace it. By shutting down the system, you halt LVM access to the disk, so you can skip this step.

If the disk is hot-swappable, detach the device using the `-a` option of the `pvchange` command:

```
# pvchange -a N /dev/disk/disk14_p2
```

Note: On an HP 9000 server, the boot disk is not partitioned so the physical volume refers to the entire disk, not the HP-UX partition. Enter the following command:

```
# pvchange -a N /dev/disk/disk14
```

3. Replace the disk.

For the hardware details on how to replace the disk, see the hardware administrator's guide for the system or disk array.

If the disk is hot-swappable, replace it.

If the disk is not hot-swappable, shut down the system, turn off the power, and replace the disk. Reboot the system. Two problems can occur:

- If you replaced the disk that you normally boot from, the replacement disk does not contain the information needed by the boot loader. In this case, interrupt the boot process and boot from the mirror boot disk, which is configured as the alternate boot path.
- If there are only two disks in the root volume group, the system probably fails its quorum check as described in [Volume Group Activation Failures](#). It can panic early in the boot process with the message:

```
panic: LVM: Configuration failure
```

In this situation, you must override quorum to boot successfully. Do this by interrupting the boot process and adding the `-lq` option to the boot command.

For information on the boot process and how to select boot options, see the *HP-UX System Administrator's Guide: Configuration Management*.

4. Notify the mass storage subsystem that the disk has been replaced.

If the system was not rebooted to replace the failed disk, run `scsimgr` before using the new disk as a replacement for the old disk. For example:

```
# scsimgr replace_wwid -D /dev/rdisk/disk14
```

This command allows the storage subsystem to replace the old disk's LUN World-Wide-Identifier (WWID) with the new disk's LUN WWID. The storage subsystem creates a new LUN instance and new device special files for the replacement disk.

5. Determine the new LUN instance number for the replacement disk. For example:

```
# ioscan -m lun
```

```
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x0  esdisk NO_HW   DEVICE  offline HP MSA Vol
          /dev/disk/disk14      /dev/rdisk/disk14
          /dev/disk/disk14_p1  /dev/rdisk/disk14_p1
          /dev/disk/disk14_p2  /dev/rdisk/disk14_p2
          /dev/disk/disk14_p3  /dev/rdisk/disk14_p3
...
disk 28 64000/0xfa00/0x1c  esdisk CLAIMED DEVICE  online  HP MSA Vol
          0/1/1/1.0x3.0x0
          /dev/disk/disk28      /dev/rdisk/disk28
```

In this example, LUN instance 28 was created for the new disk, with LUN hardware path 64000/0xfa00/0x1c, device special files /dev/disk/disk28 and /dev/rdisk/disk28, at the same lunpath hardware path as the old disk, 0/1/1/1.0x3.0x0. The old LUN instance 14 for the old disk now has no lunpath associated with it.

Note: If the system was rebooted to replace the failed disk, then `ioscan -m lun` does not display the old disk.

6. (HP Integrity servers only) Partition the replacement disk.

Partition the disk using the `idisk` command and a partition description file, and create the partition device files using `insf`, as described in [Mirroring the Root Volume on Integrity Servers](#).

7. Assign the old instance number to the replacement disk. For example:

```
# io_redirect_dsf -d /dev/disk/disk14 -n /dev/disk/disk28
```

This assigns the old LUN instance number (14) to the replacement disk. In addition, the device special files for the new disk are renamed to be consistent with the old LUN instance number.

The following `ioscan -m lun` output shows the result:

```
# ioscan -m lun /dev/disk/disk14
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x1c esdisk  CLAIMED  DEVICE   online  HP MSA Vol
          0/1/1/1.0x3.0x0
          /dev/disk/disk14      /dev/rdisk/disk14
          /dev/disk/disk14_p1  /dev/rdisk/disk14_p1
          /dev/disk/disk14_p2  /dev/rdisk/disk14_p2
          /dev/disk/disk14_p3  /dev/rdisk/disk14_p3
```

The LUN representation of the old disk with LUN hardware path 64000/0xfa00/0x0 was removed. The LUN representation of the new disk with LUN hardware path 64000/0xfa00/0x1c was reassigned from LUN instance 28 to LUN instance 14 and its device special files were renamed as /dev/disk/disk14 and /dev/rdisk/disk14.

8. Restore LVM configuration information to the new disk. For example:

```
# vgcfgrestore -n /dev/vg00 /dev/rdisk/disk14_p2
```

Note: On an HP 9000 server, the boot disk is not partitioned, so the physical volume refers to the entire disk, not the HP-UX partition. Enter the following command:

```
# vgcfgrestore -n /dev/vg00 /dev/rdisk/disk14
```

9. Restore LVM access to the disk.

If you did *not* reboot the system in [step 2](#), reattach the disk as follows:

```
# pvchange -a y /dev/disk/disk14_p2
```

On an HP 9000 server, use this command:

```
# pvchange -a y /dev/disk/disk14
```

If you did reboot the system, reattach the disk by reactivating the volume group as follows:

```
# vgchange -a y /dev/vg00
```

Note: The `vgchange` command with the `-a y` option can be run on a volume group that is deactivated or already activated. It attaches all paths for all disks in the volume group and resumes automatically recovering any disks in the volume group that were offline or any disks in

the volume group that were replaced. Therefore, run `vgchange` only after all work has been completed on all disks and paths in the volume group, and it is necessary to attach them all.

10. Initialize boot information on the disk.

For an HP Integrity server, set up the boot area and update the autoboot file in the disk's EFI partition as described in step 5 and step 6 of [Mirroring the Root Volume on Integrity Servers](#) listed in Appendix D.

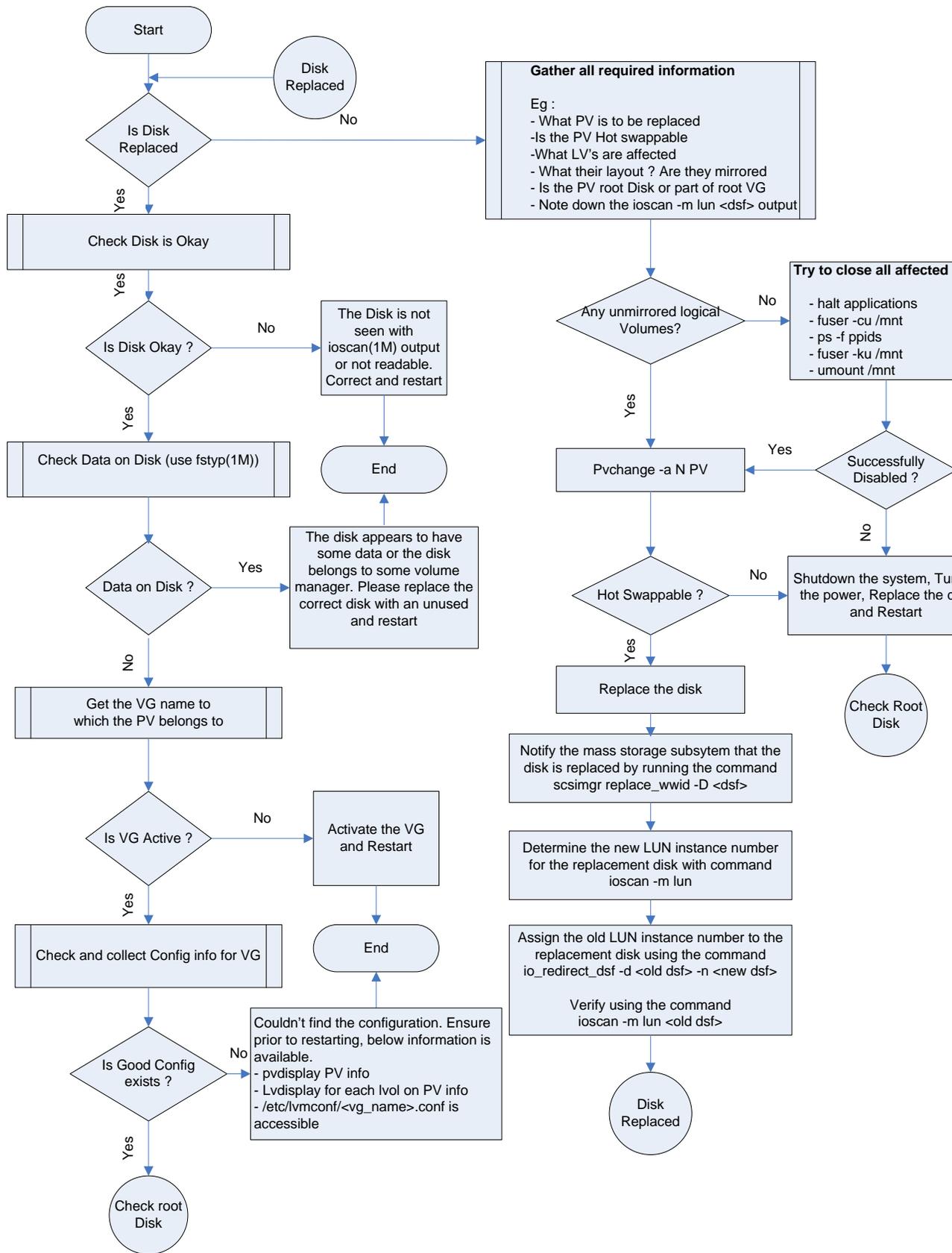
For an HP 9000 server, set up the boot area and update the autoboot file as described in step 4 and step 5 of [Mirroring the Root Volume Boot Disk on PA-RISC Servers](#) in Appendix D.

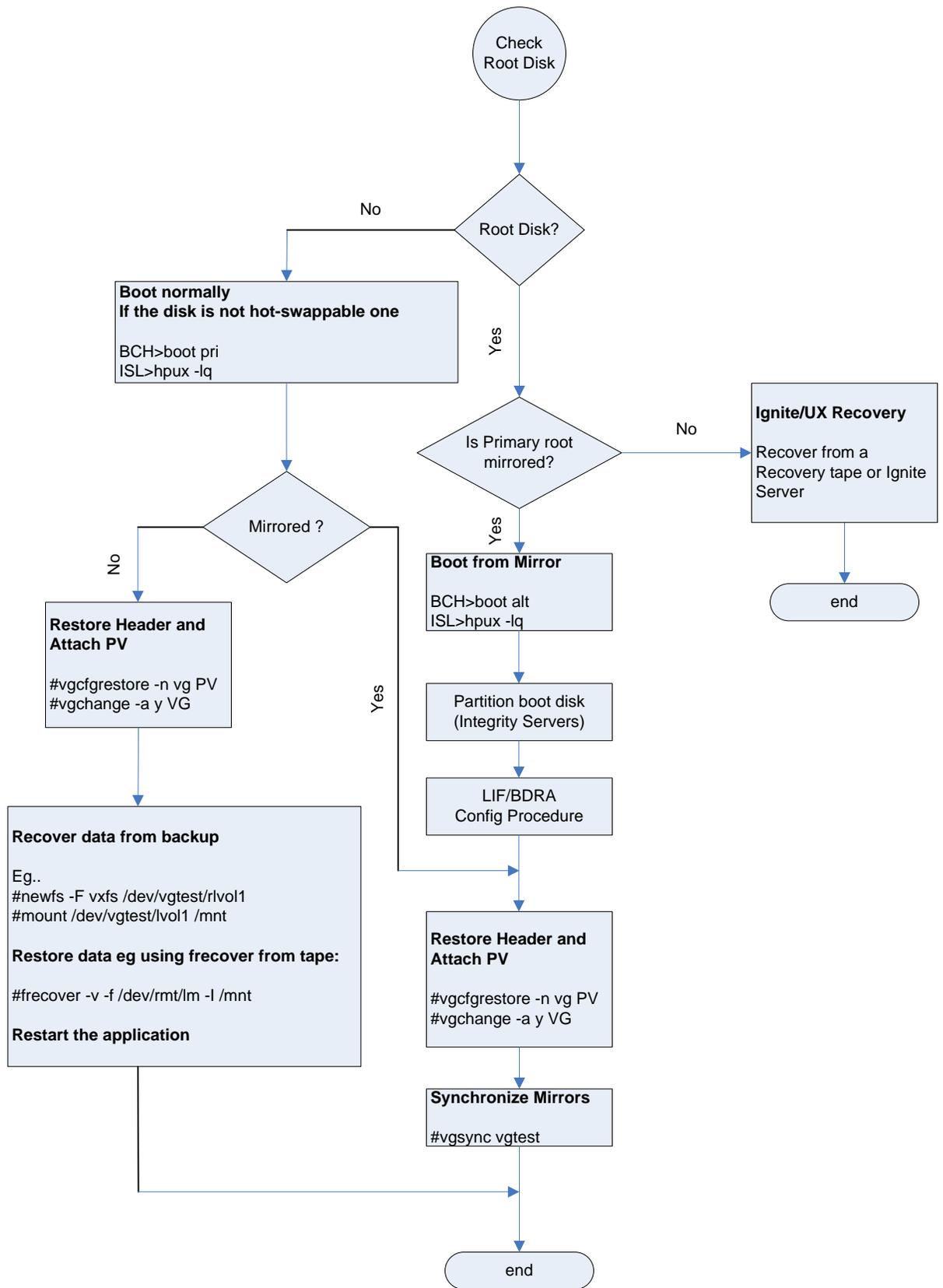
Replacing an Unmirrored Boot Disk

With the failure of an unmirrored boot disk, you have lost the only copy of information that is required to boot the system. You must reinstall to the replacement disk, or recover it from an Ignite-UX backup.

Disk Replacement Flowchart

See the disk replacement flowchart documented on the following page.





Conclusion

In your role as system manager, you will encounter disk failures. LVM can lessen the impact of those disk failures, enabling you to configure your data storage to make a disk failure transparent to users, and to keep your system and data available during the recovery process. By making use of hardware features, such as hot-swappable disks, and software features, such as mirroring and online disk replacement, you can maximize your system availability and minimize data loss due to disk failure.

Appendix A: Using Device File Types

Prior to the HP-UX 11i v3 release, there were only legacy device special files. Starting with the HP-UX 11i v3 release, mass storage devices, such as disk devices and tape devices, have two types of device files, persistent device special files and legacy device special files. You can use both to access the mass storage device independently, and can co-exist on the same system.

A legacy device special file is locked to a particular physical hardware path, and does not support agile addressing. This device special file contains hardware path information such as SCSI bus, target, and LUN (Logical Unit Number) in the device file name and minor number. Specifically, the class and instance portions of the device special file name indicate hardware path information and are in the format `c#t#d#` as follows:

c#	The instance number assigned by the operating system to the interface card. It is a decimal number with a range of 0 to 255. There is no direct correlation between instance number and physical slot number.
t#	The target address on a remote bus (for example, SCSI address). It is a decimal number with a typical range of 0 to 15.
d#	The device unit number at the target address (for example, the LUN in a SCSI device). It is a decimal number with a typical range of 0 to 7.

A persistent device special file is associated with a LUN hardware path, transparently supporting agile addressing and multipathing. In other words, a persistent device special file is unchanged if the LUN is moved from one HBA to another, moved from one switch or hub port to another, presented via a different target port to the host, or configured with multiple hardware paths. Like the LUN hardware path, the binding of device special file to device persists across reboots, but is not guaranteed to persist across installations.

Persistent and Legacy device special files are represented as follows:

- Persistent device special files, (`/dev/disk/disk3`), or
- Legacy device special file names, (`/dev/dsk/c0t6d6`)

To display the mapping between the legacy and persistent device special files, use the `ioscan` command. Either a legacy or persistent special file can be specified as an argument. Multiple legacy special files mapping to the persistent special file are separated by a white space, as follows:

```
# ioscan -m dsf /dev/disk/disk240
Persistent DSF                Legacy DSF(s)
=====
/dev/disk/disk240            /dev/dsk/c17t1d2
                             /dev/dsk/c19t1d2
# ioscan -m dsf /dev/dsk/c17t1d2
Persistent DSF                Legacy DSF(s)
=====
/dev/disk/disk240            /dev/dsk/c17t1d2
# ioscan -m dsf /dev/dsk/c19t1d2
Persistent DSF                Legacy DSF(s)
=====
/dev/disk/disk240            /dev/dsk/c19t1d2
```

You can derive hardware path information, class names, and instance numbers from `ioscan` output; see [ioscan\(1M\)](#) for more information. There are three different types of paths to a device: legacy hardware path, lunpath hardware path, and LUN hardware path. For more information, see [intro\(7\)](#).

Appendix B: Device Special File Naming Model

HP-UX 11i v3 introduces a new representation of mass storage devices called the **agile view**. In this representation, the device special file (DSF) name for each disk no longer contains path (or link) information. There are two DFS types:

- The multipathed disk has a single **persistent DSF** regardless of the number of physical paths to it.
- The legacy view, represented by the **legacy DSF**, continues to exist.

You can use both DSF types to access a given mass storage device independently, and both can coexist on a system.

While LVM supports the use of both DSF naming models within the same volume group, you can configure the volume group to use both styles of DSFs: on a given system, some volume groups can use legacy DSFs, others persistent DSFs, and the remaining can use a mix of persistent and legacy DSFs. Supporting a mixed mode configuration allows a volume group that currently uses one DSF naming model to be extended to include a different DSF naming model (see [vgextend\(1M\)](#) for more information). Any such configuration results in a mixed mode volume group. Mixed mode facilitates a phased migration from legacy to persistent naming model.

HP recommends using persistent DSFs for LVM configurations and encourages configuring new volume groups using persistent DSFs. To fully utilize all the capabilities of the new mass storage stack, HP recommends migrating volume groups from legacy DSFs to persistent DSFs. HP provides `/usr/contrib/bin/vgdsf` to facilitate this migration. This script works for both root and non-root volume groups. See the [For More Information](#) section in the [LVM Migration from legacy to persistent naming model](#) white paper for more details.

Most of the LVM commands are backward compatible, and work transparently with both legacy and persistent DSFs. New options have been added to the [vgimport\(1M\)](#) and [vgscan\(1M\)](#) commands, to appropriately choose the required DSF naming model.

Starting with the HP-UX 11i v3 release, HP recommends using persistent DSFs for LVM configurations and advises you to configure all new volume groups using persistent DSFs. To verify if the legacy naming model is enabled on a system, use the [insf\(1M\)](#) command. To disable the legacy naming model, use [rmsf\(1M\)](#). Note that to configure new volumes groups using persistent DSFs, you do not need to disable the legacy naming model (which removes all legacy I/O nodes and their device special files on the system). The `rmsf` command does not complete successfully if any legacy I/O nodes are in the open state. If this is the case, the command fails and returns information, such as process name (PID) and the device special file, about the processes that have opened the legacy I/O nodes.

For more information on the benefits of migration, see [The Next Generation Mass Storage Stack](#) white paper.

This appendix lists the new options that were added to some LVM commands to specify the preference of DSF naming model. Also, it lists the behavioral differences of commands for each message, the cause is listed, and an administrator action is recommended. The appendix is divided into two sections, one for the new options provided to specify the DSF naming model and one for the behavioral differences of commands after disabling the legacy naming model.

New Options to Specify the DSF Naming Model

Most LVM commands are transparent to the DSF naming model used, but new options were added to the [vgimport\(1M\)](#) (option `-N`) and [vgscan\(1M\)](#) (options `-N` and `-B`) commands to specify the preference of DSF naming model. These options are detailed in their respective manpages, but are summarized in the following table:

<code>vgimport -N</code>	Configure the volume group using persistent DSFs. You can only use this option together with the scan option, <code>-s</code> . In the absence of the <code>-N</code> option, legacy DSFs are used.
<code>vgscan</code>	Recover the volume group information by using LVM data structures in kernel memory, and by probing all LVM disks on the system. In the case of volume groups activated at least once since the last system boot, the <code>/etc/lvmtab</code> file is recovered with legacy and persistent DSFs, as was the configuration during activation. The <code>-N</code> and <code>-B</code> options allow the user to override this default behavior.
<code>vgscan -N</code>	Recover the <code>/etc/lvmtab</code> file using persistent DSFs, with the following exception: If there are volume groups activated that are using legacy DSFs, <code>vgscan</code> populates <code>/etc/lvmtab</code> using legacy DSFs for those physical volumes.
<code>vgscan -B</code>	Recover the <code>/etc/lvmtab</code> file using both persistent and legacy DSFs. Persistent DSFs are added before legacy DSFs.

Behavioral Differences of Commands After Disabling the Legacy Naming Model

By default, and to maintain backward compatibility, the `vgscan`, `vgscan -p` and `vgimport -s` commands execute in the legacy naming model. You need to invoke these commands together with the `-N` option, if the legacy naming model has been disabled using the `rmsf -l` command. In the absence of the `-N` option, these commands fail with the following error message:

```
"The legacy naming model has been disabled on the system.
  Try with the -N option."
```

Appendix C: Volume Group Versions and LVM Configuration Files

Volume Group Version

With the March 2008 release of HP-UX 11i v3, LVM supports a new version of volume group- Version 2.0. Version 1.0 is the version supported on all current and previous versions of HP-UX 11i.

A Version 2.0 volume group is a volume group whose metadata layout is different from the one used for Version 1.0 volume groups. Version 2.0 enables the configuration of larger volume groups (up to 512), logical volumes (up to 511), physical volumes (up to 511) and several other volume group parameters. A Version 2.0 volume group can be managed the same way as a Version 1.0 volume group, using the same user interface. But features such as sparing, bad block relocation, cluster lock, boot, dump, and primary swap are not supported on Version 2.0 volume groups. Also, the `vgmodify`, `pvck`, `lvlnboot` and `lvrmboot` commands are not supported on Version 2.0 volume groups.

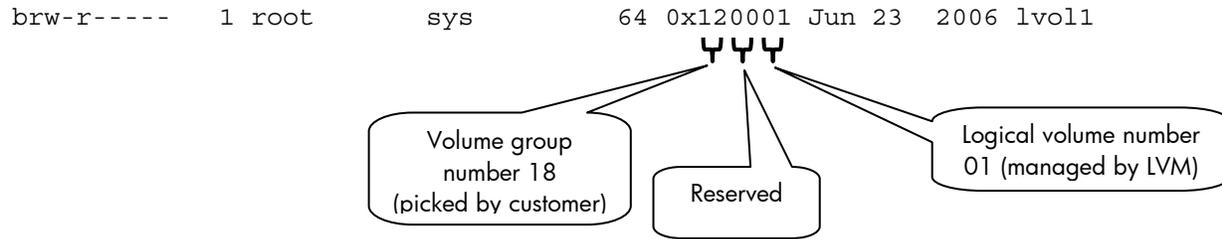
With the September 2008 release of HP-UX 11i v3, LVM supports another version of volume group- version 2.1. You can manage a Version 2.1 volume group the same way as a Version 1.0 volume group, using the same user interface. Version 2.1 is similar to Version 2.0, but allows configuration of much greater number of volume groups (up to 2048), physical volumes (up to 2048) and logical volumes (up to 2047). With this release, cluster lock is supported on Version 2.x volume groups.

Device Special Files

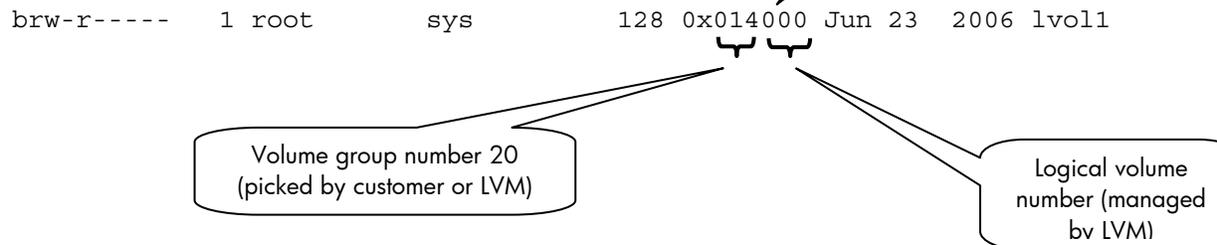
With the new volume group version, LVM device special files have changed. For Version 1.0 volume groups, procedures to create the device files will not need to change. For Version 2.x volume groups, there are a few differences illustrated in the following figure:

- Device special files for Version 1.0 volume groups are unchanged
- Device special files for Version 2.x volume groups have a new major number (128)
- Device special files for Version 2.x volume groups have a different minor number encoding scheme

1.0 Minor Number Encoding:



2.0 Minor Number Encoding:



lvmtab, lvmtab_p

LVM has a well known configuration file `/etc/lvmtab`. With the introduction of Version 2.x volume groups, LVM uses a new configuration file, `/etc/lvmtab_p`. The differences between the two versions are as follows:

- `/etc/lvmtab` is a private binary file. It contains information only related to Version 1.0 volume groups. It is compatible with all supported HP-UX releases.
- `/etc/lvmtab_p` is a new private binary file. It contains information only related to Version 2.x volume groups. It has a different internal structure from `lvmtab` and contains additional information.

For example:

```
# strings /etc/lvmtab_p
/dev/vgtest12
A00000000000000004Sun Mar 16 03:02:19 2007241fa9de-dba3-11da-9cd2-
23efa80dc3e7
/dev/disk/disk79
/dev/disk/disk80
/dev/disk/disk81
/dev/disk/disk82
/dev/disk/disk83
/dev/disk/disk84
/dev/disk/disk85
```

For more details see the following white papers available from the docs.hp.com website:

- HP-UX Logical Volume Manager and MirrorDisk/UX Release Notes
- HP-UX 11i v3 March 2008 (B.11.31)
- HP-UX LVM Supported Limits
- LVM Version 2.0 Volume Groups in HP-UX 11i v3

Appendix D: Procedures

This section contains details on some of the procedures described in earlier sections of this document.

Mirroring the Root Volume on PA-RISC Servers

To set up a mirrored root configuration, you must add a disk to the root volume group, mirror all the root logical volumes onto it, and make it bootable. For this example, the disk is at path `2/0/7.15.0` and has device special files named `/dev/rdisk/c2t15d0` and `/dev/dsk/c2t15d0`.

1. Use the `insf` command with the `-e` option to make sure the device files are in place. For example:

```
# insf -e -H 2/0/7.15.0
```

You now have the following legacy device files for this disk:

<code>/dev/dsk/c2t15d0</code>	The entire disk (block access)
<code>/dev/rdisk/c2t15d0</code>	The entire disk (character access)

Starting with HP-UX 11i v3, you must have the persistent device special files as well. To achieve this, use the `insf` command with the `-e` option or `-eC disk` option. If you are running `insf -e -H <hw_path>`, you must run the `insf` command for both the legacy and new style lun hardware path.

```
# insf -e -H 0/5/1/0.5.14.0.0.0.1
```

You now have the following persistent device files for this disk:

<code>/dev/disk/disk96</code>	The entire disk (block access)
<code>/dev/rdisk/disk96</code>	The entire disk (character access)

2. Enter the `pvcreate` command with the `-B` option to create a physical volume:

```
# pvcreate -B /dev/rdisk/c2t15d0
```

3. Enter `vgextend` to add the physical volume to your existing root volume group:

```
# vgextend /dev/vg00 /dev/dsk/c2t15d0
```

4. Enter the `mkboot` command to set up the boot area:

```
# mkboot -l /dev/rdisk/c2t15d0
```

To check, use the `lifls` command to list the contents of LIF directory on standard output:

```
# lifls -l /dev/rdisk/c2t15d0
```

5. Enter the `mkboot` command to add an autoboot file to the disk boot area. If you expect to boot from this disk only when you lose quorum, you can use the alternate string `"hpux -lq"` to disable quorum checking (This not recommended; use this with caution to prevent data loss):

```
# mkboot -a "hpux" /dev/rdisk/c2t15d0
```

Note: By default, LVM enforces the presence of less than 50% of a volume group's physical volumes being available at volume group activation time. For example, if the root volume group

contains two physical volumes, and one of them is not accessible, the system rejects to boot unless you disable the quorum check using the `-lq` option.

6. Enter `lvextend` command to mirror each logical volume in `vg00` (the root volume group) onto the specified physical volume. You must extend the logical volumes in the same order that they are configured on the original boot disk. Use the `pvdisplay` command with the `-v` option to determine the list of logical volumes and their order. For example:

```
# pvdisplay -v /dev/dsk/c0t6d0 | grep 'current.*0000 $'
00000 current /dev/vg00/lvol1 00000
00038 current /dev/vg00/lvol2 00000
00550 current /dev/vg00/lvol3 00000
00583 current /dev/vg00/lvol4 00000
00608 current /dev/vg00/lvol5 00000
00611 current /dev/vg00/lvol6 00000
00923 current /dev/vg00/lvol7 00000
01252 current /dev/vg00/lvol8 00000
```

In this example, mirror the logical volumes as follows:

```
# lvextend -m 1 /dev/vg00/lvol1 /dev/dsk/c2t15d0
# lvextend -m 1 /dev/vg00/lvol2 /dev/dsk/c2t15d0
# lvextend -m 1 /dev/vg00/lvol3 /dev/dsk/c2t15d0
# lvextend -m 1 /dev/vg00/lvol4 /dev/dsk/c2t15d0
# lvextend -m 1 /dev/vg00/lvol5 /dev/dsk/c2t15d0
# lvextend -m 1 /dev/vg00/lvol6 /dev/dsk/c2t15d0
# lvextend -m 1 /dev/vg00/lvol7 /dev/dsk/c2t15d0
# lvextend -m 1 /dev/vg00/lvol8 /dev/dsk/c2t15d0
```

The `lvextend` operation can take several minutes to copy all the data from the original copy of the data to the mirrored extents. The logical volumes are still accessible to user applications during this command.

To check the progress of the synchronization, enter:

```
# lvsdisplay -v $(find /dev/vg00 -type b) | grep stale | wc -l
```

You can use a shell loop to extend a bunch of `lvols` automatically as follows:

```
# for lvol in lvol1 lvol2 lvol3 ... (specify any LV you need to mirror)
> do
> lvextend -m 1 /dev/vgXX/$lvol /dev/dsk/cXtXdX
> done
```

7. To update the root volume group information, enter:

```
# lvlnboot -R /dev/vg00
```

8. To verify that the mirrored disk is displayed as a boot disk and that the boot, root, and swap logical volumes appear to be on both disks, enter:

```
# lvlnboot -v
```

9. To specify the mirror disk as the alternate boot path in nonvolatile memory, enter:

```
# setboot -a 2/0/7.15.0
```

10. To add a line to `/stand/bootconf` for the new boot disk, use `vi` or another text editor as follows:

```
# vi /stand/bootconf
l /dev/dsk/c2t15d0
```

In this example, `l` denotes LVM.

Mirroring the Root Volume on Integrity Servers

The procedure to mirror the root disk on Integrity servers is similar to the procedure for PA-RISC servers. The difference is that Integrity server boot disks are partitioned; you must set up the partitions, copy utilities to the EFI partition, and use the HP-UX partition device files for LVM commands.

For this example, the disk is at hardware path 0/1/1/0.1.0, with a device special file named /dev/rdisk/c2t1d0.

1. Partition the disk by using the `idisk` command and a partition description file.

First, clear the previous partition configuration on the disk:

```
# idisk -Rw /dev/rdisk/c2t1d0
```

- a. Create a partition description file. For example:

```
# vi /tmp/pdf
```

In this example, the partition description file contains:

```
3
EFI 500MB
HPUX 100%
HPSP 400MB
```

- b. Partition the disk using `idisk` and the partition description file created in step 1a:

```
# idisk -f /tmp/pdf -w /dev/rdisk/c2t1d0
```

To verify that your partitions are correctly laid out, enter the following command:

```
# idisk /dev/rdisk/c2t1d0
```

2. Enter the `insf` command with the `-e` option to create the legacy device files for all the partitions:

```
# insf -e -H 0/1/1/0.1.0
```

You now have the following device files for this disk:

/dev/dsk/c2t1d0	The entire disk (block access)
/dev/rdisk/c2t1d0	The entire disk (character access)
/dev/dsk/c2t1d0s1	The EFI partition (block access)
/dev/rdisk/c2t1d0s1	The EFI partition (character access)
/dev/dsk/c2t1d0s2	The HP-UX partition (block access)
/dev/rdisk/c2t1d0s2	The HP-UX partition (character access)
/dev/dsk/c2t1d0s3	The Service partition (block access)
/dev/rdisk/c2t1d0s3	The Service partition (character access)

Starting with the HP-UX 11i v3 release, you must have the persistent device special files. To achieve this, use the `insf` command with the `-e` option or `-eC disk` option. If you are running `insf -e -H <hw_path>`, you must run the `insf` command for both the legacy and new style lun hardware path. For example:

```
# insf -e -H 0/5/1/0.5.14.0.0.0.1
```

You now have the following device files for this disk:

/dev/disk/disk75	The entire disk (block access)
/dev/rdisk/disk75	The entire disk (character access)
/dev/disk/disk75_p1	The EFI partition (block access)
/dev/rdisk/disk75_p1	The EFI partition (character access)
/dev/disk/disk75_p2	The HP-UX partition (block access)
/dev/rdisk/disk75_p2	The HP-UX partition (character access)
/dev/disk/disk75_p3	The Service partition (block access)
/dev/rdisk/disk75_p3	The Service partition (character access)

Use `efi_fsinit` to initialize the FAT filesystem on the EFI partition:

```
# efi_fsinit -d /dev/rdisk/c2t1d0s1
```

3. Create a physical volume using `pvcreate` with the `-B` option. Be sure to use the device file denoting the HP-UX partition.

```
# pvcreate -B /dev/rdisk/c2t1d0s2
```

4. Add the physical volume to your existing root volume group using `vgextend`:

```
# vgextend /dev/vg00 /dev/dsk/c2t1d0s2
```

5. Use the `mkboot` command to set up the boot area. Specify the `-e` and `-l` options to copy EFI utilities to the EFI partition, and use the device special file for the entire disk:

```
# mkboot -e -l /dev/rdisk/c2t1d0
```

To list the contents of the EFI directory, enter the `efi_ls` command:

```
# efi_ls -d /dev/rdisk/c2t1d0s1
FileName                               Last Modified      Size
EFI/                                     11/ 9/2006         0
STARTUP.NSH                             11/ 9/2006        296
```

total space 523251712 bytes, free space 519856128 bytes

To list the contents of the LIF directory, enter the `lifls` command:

```
# lifls -l /dev/rdisk/c2t1d0s2
volume ISL10 data size 7984 directory size 8 06/04/19 14:02:34
filename type start size implement created
=====
ISL          -12800    584   242   0           06/04/19 14:02:34
AUTO        -12289     832    1    0           06/04/19 14:02:34
HPUX        -12928     840   1024  0           06/04/19 14:02:34
PAD         -12290    1864  1468  0           06/04/19 14:02:34
LABEL      BIN    3336   8     0           06/11/09 06:22:46
```

Enter the following to check the contents of the `AUTO` file on EFI partition on primary disk:

```
# efi_cp -d /dev/rdisk/c1t0d1s1 -u /EFI/HPUX/AUTO /tmp/prim
# cat /tmp/prim
```

6. To update the autoboot file in the EFI partition, follow these steps:
 - a. Create an `AUTO` file in the current directory. If you expect to boot from this disk only when you have lost quorum, you can use the alternate string `"boot vmunix -lq"` to disable quorum checking (This not recommended; use this with caution to prevent data loss):

```
# echo "boot vmunix" > ./AUTO
```

- b. Copy the file from the current directory into the new disk EFI partition. Be sure to use the device file with the s1 suffix:

```
# efi_cp -d /dev/rdisk/c2t1d0s1 ./AUTO /efi/hpux/auto
```

- c. To check the contents of the AUTO file on EFI partition on mirror disk:

```
# efi_cp -d /dev/rdisk/c2t0d0s1 -u /EFI/HPUX/AUTO /tmp/mir
# cat /tmp/mir
boot vmunix
```

7. Use the `lvextend` command to mirror each logical volume in `vg00` (the root volume group) onto the specified physical volume. You must extend the logical volumes in the same order that they are configured on the original boot disk. Use the `pvdisplay` command with the `-v` option to determine the list of logical volumes and their order. For example:

```
# pvdisplay -v /dev/dsk/c0t0d0s2 | grep 'current.*0000 $'
00000 current /dev/vg00/lvol1 00000
00010 current /dev/vg00/lvol2 00000
00138 current /dev/vg00/lvol3 00000
00151 current /dev/vg00/lvol4 00000
00158 current /dev/vg00/lvol5 00000
00159 current /dev/vg00/lvol6 00000
00271 current /dev/vg00/lvol7 00000
00408 current /dev/vg00/lvol8 00000
```

In this example, mirror the logical volumes as follows:

```
# lvextend -m 1 /dev/vg00/lvol1 /dev/dsk/c2t1d0s2
# lvextend -m 1 /dev/vg00/lvol2 /dev/dsk/c2t1d0s2
# lvextend -m 1 /dev/vg00/lvol3 /dev/dsk/c2t1d0s2
# lvextend -m 1 /dev/vg00/lvol4 /dev/dsk/c2t1d0s2
# lvextend -m 1 /dev/vg00/lvol5 /dev/dsk/c2t1d0s2
# lvextend -m 1 /dev/vg00/lvol6 /dev/dsk/c2t1d0s2
# lvextend -m 1 /dev/vg00/lvol7 /dev/dsk/c2t1d0s2
# lvextend -m 1 /dev/vg00/lvol8 /dev/dsk/c2t1d0s2
```

The `lvextend` operation can take several minutes to copy all the data from the original copy of the data to the mirrored extents. The logical volumes are still accessible to users applications during this command.

To check the progress of the synchronization, you can use the following:

```
# lvdisplay -v $(find /dev/vg00 -type b) | grep stale | wc -l
```

You can use a shell loop to extend a bunch of `lvols` automatically as follows:

```
# for lvol in lvol1 lvol2 lvol3 ... (specify any LV you need to mirror)
> do
> lvextend -m 1 /dev/vgXX/$lvol /dev/dsk/cXtXdX
> done
```

8. Update the root volume group information:

```
# lvolnboot -R /dev/vg00
```

9. Verify that the mirrored disk is displayed as a boot disk and that the boot, root, and swap logical volumes appear on both disks:

```
# lvolnboot -v
```

10. Specify the mirror disk as the alternate boot path in nonvolatile memory:

```
# setboot -a 0/1/1/0.1.0
```

11. To add a line to `/stand/bootconf` for the new boot disk, use `vi` or another text editor as follows:

```
# vi /stand/bootconf  
1 /dev/dsk/c2t1d0s2
```

where 1 denotes LVM.

12. Copy the HP service partition(skip this, if you do not have a service partition):

```
# dd if=/dev/rdisk/c1t0d1s3 of=/dev/rdisk/c2t1d0s3 bs=1024k &
```

Note : The previous example recommends running the `dd` command in the background (by adding `&` at the end of the command) if you are not sure the `dd` command will hang when it does the read. If the `dd` command is run in the foreground, `Ctrl+C` stops the read on the disk.

Appendix E: LVM Error Messages

This appendix lists some of the warning and error messages reported by LVM. For each message, the cause is listed, and an administrator action is recommended. The appendix is divided into two sections, one for LVM command errors, and one for the system log file `/var/adm/syslog/syslog.log` error messages.

LVM Command Error Messages

All LVM commands

```
vgcfgbackup: /etc/lvmtab is out of date with the running kernel:  
Kernel indicates # disks for "/dev/vgname"; /etc/lvmtab has # disks.  
Cannot proceed with backup.
```

Cause:

The number of current and active physical volumes, printed by `vgdisplay` as `Cur PV` and `Act PV`, are not the same. `Cur PV` and `Act PV` must always agree for the volume group. This error also indicates that the `/etc/lvmtab` file, which is used to match physical volumes to a volume group, is out of date with the LVM data structures in memory and on disk.

Recommended Action:

Try to locate any missing disks. For each of the disk in the volume group, use `ioscan` and `diskinfo` to confirm that the disk is functioning properly.

lvchange

```
"m": Illegal option.
```

Cause:

The system does not have HP MirrorDisk/UX installed.

Recommended Action:

Install HP MirrorDisk/UX.

lvextend

```
lvextend: Not enough physical extents available.  
Logical volume "/dev/vgname/lvname" could not be extended.  
Failure possibly caused by strict allocation policy
```

Cause:

There is not enough space in the volume group to extend the logical volume to the requested size. This is typically caused by one of three situations:

- a. There are not enough free physical extents in the volume group. Run `vgdisplay` to confirm the number of available physical extents, and multiply that number by the extent size to determine the free space in the volume group. For example:

```
# vgdisplay vg00  
--- Volume groups ---  
VG Name                /dev/vg00  
VG Write Access        read/write  
VG Status               available  
Max LV                 255  
Cur LV                 10  
Open LV                 10
```

Max PV	16
Cur PV	1
Act PV	1
Max PE per PV	4350
VGDA	2
PE Size (Mbytes)	4
Total PE	4340
Alloc PE	3740
Free PE	600
Total PVG	0
Total Spare PVs	0
Total Spare PVs in use	0

In this example, the total free space is 600 * 4 MB, or 2400 MB.

- b. The logical volume is mirrored with a strict allocation policy, and there are not enough extents on a separate disk to comply with the allocation policy. To confirm this, run `lvdisplay` to determine which disks the logical volume occupies, and then check whether there is sufficient space on the other disks in the volume group.
- c. In a SAN environment, one of the disks was dynamically increased in size. LVM did not detect the asynchronous change in size.

Recommended Action:

- a. Choose a smaller size for the logical volume, or add more disk space to the volume group.
- b. Choose a smaller size for the logical volume, or add more disk space to the volume group. Alternatively, free up space on an available disk using `pvmove`.
- c. None. LVM does not support dynamic resizing of disks.

Note: When an LVM volume group is created, several configuration parameters are set (such as `max_pe`, `max_pv`, and `max_lv`). The new `vgmodify` command enables you to change these configuration parameters on an existing volume group, which avoids having to migrate user data. For more information, see [vgmodify\(1M\)](#) and the [LVM Volume Group Dynamic LUN expansion \(DLE\)/vgmodify](#) whitepaper.

"m": Illegal option.

Cause:

The system does not have HP MirrorDisk/UX installed.

Recommended Action:

Install HP MirrorDisk/UX.

`lvlnboot`

lvlnboot: Unable to configure swap logical volume. Swap logical volume size beyond the IODC max address.

Cause:

The boot disk firmware cannot access the entire range of the swap logical volume. This happens with older host bus adapters when primary swap is configured past 4 GB on the disk.

Recommended Action:

Upgrade the system firmware or use a newer host bus adapter that supports block addressing. If neither of these actions is successful, reduce the size of the primary swap logical volume so that it does not exceed 4 GB.

pvchange

"a": Illegal option.

Cause:

LVM OLR is not installed.

Recommended Action:

Install the patches enabling LVM OLR, or use an alternate replacement procedure.

The HP-UX kernel running on this system does not provide this feature. Install the appropriate kernel patch to enable it.

Cause:

LVM OLR is not completely installed. Both the LVM command and kernel components are required to enable LVM OLR. In this case, the command patch is installed and the kernel patch is not.

Recommended Action:

Install the appropriate kernel patch to enable LVM OLR, or use an alternate replacement procedure.

Unable to detach the path or physical volume via the pathname provided. Either use pvchange(1M) -a N to detach the PV using an attached path or detach each path to the PV individually using pvchange(1M) -a n

Cause:

The specified path is not part of any volume group because the path has not been successfully attached to the otherwise active volume group to which it belongs.

Recommended Action:

Check the specified path name to make sure it is correct. If the error occurred while detaching a physical volume, specify a different path that it was attached to before. If it is not clear whether any path was attached before, individually detach each path to the physical volume using pvchange with the -a n option.

Warning: Detaching a physical volume reduces the availability of data within the logical volumes residing on that disk. Prior to detaching a physical volume or the last available path to it, verify that there are alternate copies of the data available on other disks in the volume group. If necessary, use pvchange(1M) to reverse this operation.

Cause:

This warning is advisory only and generated whenever a path or physical volume is detached.

Recommended Action:

None.

vgcfgbackup

Invalid LVMREC on Physical Volume.

Cause:

The LVM header on the disk is incorrect. This can happen when an existing LVM disk is

overwritten with a command like `dd` or `pvcreate`. If the disk is shared between two systems, it is likely that one of the systems was not aware that the disk was already in a volume group. The corruption can also be caused by running `vgchgid` incorrectly when using BCV split volumes.

Recommended Action:

Restore a known good configuration to the disk using `vgcfgrestore`. Be sure to use a valid copy dated before the first occurrence of the problem.

```
# vgcfgrestore -n vgname pvname
```

`vgcfgrestore`

```
Cannot restore Physical Volume pvname Detach the PV or deactivate the VG, before restoring the PV.
```

Cause:

The `vgcfgrestore` command was used to initialize a disk that already belongs to an active volume group.

Recommended Action:

Detach the physical volume or deactivate the volume group before attempting to restore the physical volume. If there is reason to believe that the data on the disk is corrupted, the disk can be detached and marked using `vgcfgrestore` then attached again without replacing the disk. This causes LVM to reinitialize the disk and synchronize any mirrored user data mapped there.

`vgchange`

```
Warning: couldn't query physical volume "pvname":  
The specified path does not correspond to physical volume attached to  
this volume group  
Warning: couldn't query all of the physical volumes.
```

Cause:

This error has several possible causes:

- a. The disk was missing when the volume group was activated, but was later restored. This typically occurs when a system is rebooted or the volume group is activated with a disk missing, uncabled, or powered down.
- b. The disk LVM header was overwritten with the wrong volume group information. If the disk is shared between two systems, it is likely that one of the systems was not aware that the disk was already in a volume group. To confirm, check the volume group information using the `dump_lvmtab` command, available from your HP support representative, and look for inconsistencies. For example:

```
# dump_lvmtab -s | more  
SYSTEM : 0x35c8cf58  
TIME    : 0x3f9acc69 : Sat Oct 25 15:18:01 2003  
FILE    : /etc/lvmtab  
HEADER  : version:0x03e8   vgnum:7  
VG[00]  VGID:35c8cf58 3dd13164 (@0x00040c) pvnum:2 state:0  
/dev/vg00  
  (00)  VGID:35c8cf58 3dd13164 PVID:35c8cf58 3dd13164  
/dev/dsk/c0t6d0  
  (01)  VGID:35c8cf58 3dd13164 PVID:35c8cf58 3dda4694  
/dev/dsk/c4t6d0  
VG[01]  VGID:065f303f 3e63f01a (@0x001032) pvnum:92 state:0  
/dev/vg01  
  (00)  !VGID:35c8cf58 3f8df316 PVID:065f303f 3e63effa  
/dev/dsk/c40t0d0  
  (01)  !VGID:35c8cf58 3f8df316 PVID:065f303f 3e63effe
```

```

/dev/dsk/c40t0d4
  (02) !VGID:35c8cf58 3f8df316 PVID:065f303f 3e63f003
/dev/dsk/c40t1d0
...

```

In this example, note that the volume group ids (VGID) for the disks in /dev/vg01 are not consistent; inconsistencies are marked !VGID.

Recommended Action:

- a. Use `ioscan` and `diskinfo` to confirm that the disk is functioning properly. Re-activate the volume group using the following command:


```
# vgchange -a y vgname
```
- b. There are several methods of recovery from this error. If you are not familiar with the commands outlined in the following procedures, contact your HP support representative for assistance.
 - 1) Restore a known good configuration to the disks using `vgcfgrestore`. Be sure to use a valid copy dated before the first occurrence of the problem.


```
# vgcfgrestore -n vgname pvname
```
 - 2) Recreate the volume group and its logical volumes, restoring the data from the most current backup.
 - 3) Export and re-import the volume group using the following commands:


```
# vgexport -m vgname.map -v -f vgname.file /dev/vgname
# mkdir /dev/vgname
# mknod /dev/vgname/group c 64 unique_minor_number
# vgimport -m vgname.map -v -f vgname.file /dev/vgname
```

```
vgchange: Couldn't set the unique id for volume group "/dev/vgname"
```

Cause:

There are multiple LVM group files with the same minor number.

Recommended Action:

List the LVM group files. If there are any duplicate minor numbers, export one of the affected volume groups, create a new group file with a unique minor number, and re-import the volume group. If you are not familiar with the procedure shown here, contact your HP support representative for assistance:

```

# ll /dev/*/group
# vgexport -m vgname.map -v -f vgname.file /dev/vgname
# mkdir /dev/vgname
# mknod /dev/vgname/group c 64 unique_minor_number
# vgimport -m vgname.map -v -f vgname.file /dev/vgname

```

vgcreate

```

vgcreate: Volume group "/dev/vgname" could not be created:
VGRA for the disk is too big for the specified parameters. Increase
the extent size or decrease max_PVs/max_LVs and try again.

```

Cause:

The Volume Group Reserved Area at the front of each LVM disk cannot hold all the information about the disks in this volume group. This error typically occurs if you use disks larger than 100 GB.

Recommended Action:

Adjust the volume group creation parameters. Use the `-s` option of the `vgextend` command to select an extent size larger than 4 MB, or use the `-p` option to select a smaller number of physical volumes. See [vgcreate\(1M\)](#) for information on these options.

vgdisplay

```
vgdisplay: Couldn't query volume group "/dev/vg00".  
Possible error in the Volume Group minor number;  
Please check and make sure the group minor number is unique.  
vgdisplay: Cannot display volume group "/dev/vg00".
```

Cause:

This error has several possible causes:

- a. There are multiple LVM group files with the same minor number.
- b. Serviceguard was previously installed on the system, and the `/dev/slvmvg` device file still exists.

Recommended Action:

- a. List the LVM group files. If there are any duplicate minor numbers, export one of the affected volume groups, create a new group file with a unique minor number, and re-import the volume group. If you are not familiar with the following process, contact your HP support representative for assistance:

```
# ll /dev/*/group  
# vgexport -m vgroupname.map -v -f vgroupname.file /dev/vgroupname  
# mkdir /dev/vgroupname  
# mknod /dev/vgroupname/group c 64 unique_minor_number  
# vgimport -m vgroupname.map -v -f vgroupname.file /dev/vgroupname
```

- b. Remove the `/dev/slvmvg` device file and recreate the `/etc/lvmtab` file with the following commands:

```
# rm /dev/slvmvg  
# mv /etc/lvmtab /etc/lvmtab.old  
# vgscan -v
```

```
Warning: couldn't query physical volume "pvname":  
The specified path does not correspond to physical volume attached to  
this volume group  
Warning: couldn't query all of the physical volumes.
```

Cause:

This error has several possible causes. They are described earlier under the `vgchange` error messages.

Recommended Action:

See the recommended actions under the `vgchange` error messages.

vgextend

```
vgextend: Not enough physical extents per physical volume.  
Need: #, Have: #.
```

Cause:

The disk size exceeds the volume group maximum disk size. This limitation is defined when the volume group is created, as a product of the extent size specified with the `-s` option of `vgcreate` and the maximum number of physical extents per disk specified with the `-e` option. Typically, the disk is successfully added to the volume group, but all of it is not accessible.

Recommended Action:

The volume group extent size and number of physical extents per disk are not dynamic. The only way to use the entire disk is to re-create the volume group with new values for the `-s` and `-e` options. Alternatively, you can work with an HP support representative to adjust the volume group characteristics using the `vgmodify` command; note that this utility is currently unsupported and available only from your HP support representative.

`vgimport`

Verification of unique LVM disk id on each disk in the volume group /dev/vgname failed.

Cause:

There are two possible causes for this message:

- a. The `vgimport` command used the `-s` option, and two or more disks on the system have the same LVM identifier; this can happen when disks are created with BCV copy or cloned with `dd`.
- b. LVM was unable to read the disk header; this can happen when you create new logical units on a SAN array.

Recommended Action:

- a. Do not use the `-s` option to `vgimport`. Alternatively, use `vgchgid` to change the LVM identifiers on copied or cloned disks.
- b. Retry the `vgimport` command. For a long term solution, install patch PHKL_30510 or one of its superseding patches.

Syslog Error Messages

LVM: VG 64 0x010000: Data in one or more logical volumes on PV 188 0x072000 was lost when the disk was replaced. This occurred because the disk contained the only copy of the data. Prior to using these logical volumes, restore the data from backup.

Cause:

LVM cannot synchronize the data on a replaced disk automatically, as when LVM discovers an unmirrored logical volume residing on a disk that was just replaced. When all data on a disk is mirrored elsewhere and a copy is available, LVM automatically synchronizes the data on the replaced disk from the mirrors of the data on other disks.

Recommended Action:

Restore the contents of the logical volume from backup.

LVM: VG 64 0x010000: PVLink 188 0x072000 Detached.

Cause:

This message is advisory and generated whenever a disk path is detached.

Recommended Action:

None.

Appendix F: Moving a Root Disk to a New Disk or Another Disk

Follow these steps to move root disk `/dev/dsk/c1t1d1` (source disk) to disk `/dev/dsk/c2t2d2` (destination disk) staying within the same volume group:

1. To make the destination disk a bootable LVM disk, enter:
`# pvcreate -f -B /dev/rdisk/c2t2d2`
2. To make the disk bootable. Enter:
`# mkboot /dev/rdisk/c2t2d2`
`# mkboot -a "hpux -a (;0)/stand/vmunix" /dev/rdisk/c2t2d2`
3. To extend your root volume group with the destination disk, enter:
`# vgextend /dev/vg00 /dev/dsk/c2t2d2`
4. To move all physical extents from the source disk to the destination disk, enter:
`# pvmove /dev/dsk/c1t1d1 /dev/dsk/c2t2d2`
5. To reduce the source disk from the volume group, enter:
`# vgreduce /dev/vg00 /dev/dsk/c1t1d1`
6. To shut down and reboot from the new root disk in maintenance mode, enter:
`ISL> hpux -lm (;0)/stand/vmunix`
7. To update the `BDRA` and the `LABEL` file, enter:
`# vgchange -a y /dev/vg00`
`# lvinboot -b /dev/vg00/lvol1`
`# lvinboot -s /dev/vg00/lvol2`
`# lvinboot -r /dev/vg00/lvol3`
`# lvinboot -Rv`
`# vgchange -a n /dev/vg00`
8. Reboot the system.

Appendix G: Recreating Volume Group Information

There might be situations when the volume group directory, for example `vgtest` under `/dev`, is accidentally removed. In such a situation, use the following steps to re-create the `vgtest` volume group:

1. To manually create the directory, enter:

```
# mkdir /dev/vgtest
```
2. To create the character special file `group` under `/dev/vgtest` directory, enter:

```
# mknod /dev/vgtest/group c 64 0xXX0000
```

(`XX` = the minor number for the group file. This should be unique among all the volume groups on the system. `XX` ranges from `00` to `ff`.)
3. To move the `/etc/lvmtab` file, enter:

```
# mv /etc/lvmtab /etc/lvmtab.old
```
4. To create the new `/etc/lvmtab` file, enter:

```
# vgscan -v
```
5. To import the information from the disk that was removed by accident, enter:
Note: If the system has not been rebooted since the accident, you must do it now.

```
# vgimport -v /dev/vgtest /dev/dsk/c1t1d1 <and any other disks>
```
6. To activate the imported volume group, enter:

```
# vgchange -a y /dev/vgXX
```
7. Mount the file systems or just reboot the system.

If you have naming conflicts, remember that you can rename logical volumes, but you must export and import volume groups for renaming.

Appendix H: Disk Relocation and Recovery Using `vgexport` and `vgimport`

Follow these steps to move a volume group `/disks` from one system (for example, A) to another system (for example, B):

1. Unmount all of the mounted `lvols` within the volume group, `vgtest`, that you are exporting, and close all of the logical volumes:

```
# umount /dev/vgtest/lvolX /mount_point_name (repeat for all lvols)
```
2. To deactivate the volume group `vgtest`, enter:

```
# vgchange -a n /dev/vgtest
```
3. To check the group file minor number and all `lvols`, enter:

```
# cd /dev/vgtest  
# ll
```
4. To export the volume group LVM info to the map file, enter:

```
# vgexport -v -m /tmp/vgtest_map /dev/vgtest
```
5. Enter the following commands on system B to create the volume group director `/dev/vgtest` and the character special file group:

```
# mkdir /dev/vgtest  
# mknod /dev/vgtest/group c 64 0xXX0000  
      (XX = volume group minor number, should be unique on System B)
```
6. Import the LVM volume group `vgtest` onto the system:

```
# vgimport -v -m /tmp/vgtest_map /dev/vgtest <import_pv1>  
<import_pv2> ...
```

Notes:

1. During the `vgimport` action, the order of physical volumes listed in the `/etc/lvmtab` file depends on the order in which the physical volumes are presented on the `vgimport` command line. Whenever a bootable volume group is imported, the boot information present in the boot disks might be incorrect due to the change in the order of imported disks in the `/etc/lvmtab` file. This is because the boot information on the boot disks assumes a certain order of disks in `/etc/lvmtab` and requires a resynchronization of this information after the first activation of a newly imported bootable volume group. To resynchronize the information on the boot disk after the first activation of a newly imported bootable volume group, enter the `lvlnboot` command in recovery mode (`-R` option).
2. Starting with HP-UX 11i v3, while importing a volume group for a given physical volume belonging to the importing volume group, there exists a persistent device special file (and the legacy device special files provided the legacy mode is not disabled using the `rmsf -L` command).

If you choose to use the new feature introduced by the mass storage subsystem (that supports multiple paths to a device and allows multiple paths simultaneously), follow these steps:

- a. Ensure that the Mass Storage Subsystem multipath behavior is enabled using the `scsimgr` command.
- b. On the `vgimport` command line, use the persistent device special file of the physical volumes to configure the LVM. If you need to configure legacy device files as well for LVM configuration, you must ensure that the persistent device special file precedes the corresponding legacy device special files on the `vgimport` command line. In such cases, another alternative is to first configure the LVM with persistent device special files

using `vgimport` command, followed by configuring the LVM with legacy device special files as well using `vgextend` command.

If you choose to use the traditional LVM behavior on HP-UX 11i v3 as well, follow these steps:

- a. Use the `scsimgr` command to disable the Mass Storage Subsystem multipath behavior.
 - b. Only the legacy device special files are used in the `vgimport` command line while configuring the LVM volume group.
3. Activate the volume group and enter `vgdisplay -v /dev/vgtest` to see the result:
- ```
vgchange -a y /dev/vgtest
```

## Appendix I: Splitting Mirrors to Perform Backups

Make sure the database is not active before you split it. The following example uses `/dev/vg02/lvol1` to show this process. Use `lvdisplay -v /dev/vg02/lvol1` to verify that `/dev/vg02/lvol1` is mirrored and current.

1. To split the logical volume, enter:

```
sync
lvsplit -s backup /dev/vg02/lvol1
```

The system console displays the following message:

```
Logical volume "/dev/vg02/lvol1backup" has been successfully created
with character device "/dev/vg02/rlvol1backup".
Logical volume "/dev/vg02/lvol1" has been successfully split.
```

The original logical volume, `/dev/vg02/lvol1`, continues to be available for use.

The new logical volume, `/dev/vg02/lvol1backup`, split off from the original, now exists.

2. Enter `fsck` on the split logical volume:

```
fsck /dev/vg02/rlvol1backup
```

or:

```
fsck -F vxfs /dev/vg02/rlvol1backup (for JFS file systems)
```

3. Perform the backup of the file system:

```
"/dev/vg02/lvol1backup". (preferred backup utility)
```

4. When the backup is done, unmount the file system (if you had mounted it) in the split-off logical volume when you are ready to merge the split pairs.

5. To merge the split-of the logical volume and the original logical volume back into one mirrored logical volume, enter:

```
lvmerge /dev/vg02/lvol1backup /dev/vg02/lvol1
```

6. To verify the merged mirror, enter:

```
lvdisplay -v /dev/vg02/lvol1
```

## Appendix J: Moving an Existing Root Disk to a New Hardware Path

Before you shut down the system, note the `/etc/lvmtab` contents, and note which disks are in `vg00`. Shut down the system and connect the existing root drive to the new path. Boot up the system and escape boot sequence, and then boot off the root drive at the new hardware path to ISL.

1. To boot into LVM maintenance mode, enter:  
`ISL> hpux -lm (;0)/stand/vmunix`
2. To find the new hardware path of the root disk, enter:  
`# ioscan -fnC disk`
3. To install the special file for the root disk at the new hardware path, enter:  
`# insf -e -H <hardware_path>`
4. To find the special device file for the root disk, enter:  
`# ioscan -fnC disk` (example of device file name: `/dev/dsk/c2t6d0`)
5. Note the minor number of the `_group` file in `/dev/vg00`.
6. To export the root file system `vg00` to a mapfile, enter:  
`# vgexport -v -m /tmp/vg00_map /dev/vg00`
7. To create the volume group and the group file to be used for the `vgimport` command, enter:  
`# mkdir /dev/vg00`  
`# mknod /dev/vg00/group c 64 0x000000`
8. To import the root disk and any other disks that might be in the `vg00` volume group, enter:  
`# vgimport -v -m /tmp/vg00_map /dev/vg00 /dev/dsk/cXtXd0`  
<and any other disk in `vg00`>
9. To activate the volume group, enter:  
`# vgchange -a y /dev/vg00`
10. To update the LIF's Label file with the information contained in the BDRA, enter:  
`# lvlboot -Rv`
11. To reboot the system, enter:  
`# shutdown -r 0`

**Note:** Do not forget to update the Stable Storage new boot path from the ISL prompt!

## For more information

To learn more about some of the LVM features, see the following documents on the HP documentation website:

<http://docs.hp.com> (Use search with the given name of the whitepaper)

<http://www.docs.hp.com/en/oshpux11iv3#LVM%20Volume%20Manager>

- LVM Version 2.0 Volume Groups in HP-UX 11i v3
- SLVM Single-Node Online Reconfiguration (SLVM SNOR)
- LVM Online Disk Replacement (LVM OLR)
- When Good Disks Go Bad: Dealing with Disk Failures under LVM
- LVM Volume Group Dynamic LUN expansion (DLE)/vgmodify
- LVM Volume Group Quiesce/Resume
- LVM New Features in HP-UX 11i v3

To learn more about configuring LVM and migration of LVM VG configuration from legacy to agile naming model, see the following documents on HP documentation website:

<http://docs.hp.com> (Use search with the given name of the whitepaper)

<http://www.docs.hp.com/en/oshpux11iv3#LVM%20Volume%20Manager>

- LVM Migration from Legacy to Agile Naming Model
- HP-UX System Administrator's Guide: Logical Volume Management
- HP-UX LVM Performance Assessment ( The whitepaper will be available soon)

To learn more about the agile view and the new mass storage stack, see the following document on the HP documentation website:

<http://docs.hp.com/en/netsys.html#Storage%20Area%20Management>

- The Next Generation Mass Storage Stack

To learn more about supported node and host name sizes on HP-UX, see the following document on HP documentation website:

<http://www.docs.hp.com/en/oshpux11iv3#White%20Papers>

## Call to Action

HP welcomes your input. Please give us comments about this white paper, or suggestions for LVM or related documentation, through our technical documentation feedback website:

<http://docs.hp.com/en/feedback.html>

© 2008 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

5991-1236, 10/2008



invent