

HP-UX 11i v3 Crash Dump Improvements

January, 2007



Executive Summary	2
1 Overview of HP-UX 11i v3 Dump Improvements.....	2
2 Terms and Definitions	3
3 Performance Improvements.....	3
4 Configuration Improvements	14
5 Availability and Manageability Improvements.....	15
6 References.....	18
7 For more information.....	18

Executive Summary

Crash dump, the ability to write (dump) a copy of system memory onto disk in the event of a catastrophic system failure, is critical for system problem analysis and resolution. The Crash Dump utility has been enhanced in HP-UX 11i v3 to significantly increase performance and scalability, and to improve the availability and manageability of the dump configuration.

Dump performance is important because the amount of time it takes to perform a dump impacts directly the availability of a system. This is particularly true on large memory systems, because the more memory you have the longer it may take for the system to complete the dump and come back up. HP-UX 11i v3 provides new performance capabilities that can significantly reduce the dump time. HP testing has shown that dump times can be reduced to less than one-quarter of HP-UX 11i v2 dump times on equivalent configurations.

The dump configuration has been simplified in HP-UX 11i v3, unifying multiple diverse mechanisms (previously defined for different types of volume managers or raw devices) into a single mechanism, while maintaining backward compatibility.

New dump availability and manageability algorithms have also been added into HP-UX 11i v3 which provide run-time auto-detection and path failover/reconfiguration of dump device paths when an existing path fails. The dump configuration automatically adjusts when the size of a dump device expands or contracts, or when a dump device goes offline.

The dump format is unchanged in HP-UX 11i v3. Hence debuggers and utilities do not require any associated changes. Backward compatibility is maintained.

1 Overview of HP-UX 11i v3 Dump Improvements

The improvements to the crash dump facility in HP-UX 11i v3 fall into the following categories:

- Performance improvements
- Configuration improvements
- Availability and Manageability improvements

The performance improvements parallelize the dump process, creating multiple threads of execution to write to multiple devices in parallel and thus significantly reduce the overall dump time. Each thread of execution (known as a dump unit; see definition below) requires its own set of CPU¹s, dump devices, and other resources. As a result, the system configuration and dump device configuration determines the amount of parallelism that can be achieved and the resulting speed-up of the dump. See Section 3, "Performance Improvements", for details.

The configuration improvements centralize and simplify several different mechanisms for marking dump devices persistently across boot (`/stand/system` file definitions, and the `lvinboot`, and `vxvmbot` commands) into a single mechanism. For backwards compatibility, the old mechanisms can still be used but will be obsoleted in a future release. These improvements are discussed in Section 4, "Configuration Improvements".

The availability and manageability improvements provide auto-reconfiguration of the dump device through a new path when the currently selected path goes offline, as well as other features such as intelligent path selection to support the performance improvements, avoiding offline devices in the dump, and online expansion/contraction of dump devices. Section 5, "Availability and Manageability Improvements", discusses this area of improvements.

¹ The term "CPU" in this paper refers to a logical processor available to the HP-UX operating system. This is equivalent to a processor core if Itanium multi-threading is disabled. Note that online CPU addition or deletion will affect the number of CPUs available at dump.

2 Terms and Definitions

General Terms:

HBA	Host Bus Adapter. E.g., an I/O card with one or more ports on it for attachment to Fibre Channel, parallel SCSI, or other mass storage connectivity to a dump device.
Target	A storage device attachment to a mass storage interconnect such as Fibre Channel or SCSI.
LUN	An end device in a mass storage interconnect. A dump device in this paper.

Dump Related Terms:

Dump unit	A thread of execution during dump. A dump unit requires its own set of CPUs, dump devices, and other resources, which are non-overlapping with other dump units.
Reentrancy	Capability of a dump driver to issue multiple I/Os simultaneously, one I/O per HBA port, during dump.
Concurrency	Capability of a dump driver to issue multiple I/Os simultaneously per HBA port, during dump. In HP-UX 11i v3 this capability means that the driver can issue I/Os simultaneously to multiple devices under a given HBA port, one I/O per device.
Parallel Dump	The mode in the HP-UX 11i v3 dump infrastructure which enables the parallelism features.
Reentrant HBA port or device	An HBA port or device controlled by a reentrant driver.
Concurrent HBA port or device	An HBA port or device controlled by a concurrent driver.
td, mpt, c8xx, ciss, sasd	Driver names of various HP-provided dump drivers.

Other Terms:

Multi-pathing	The ability to find the various paths to a LUN, and failover to use an alternate path when a given path fails, and/or to load-balance across the various paths. HP-UX 11i v3 provides native multi-pathing which is built-in to the next generation mass storage stack.
----------------------	---

3 Performance Improvements

In earlier versions of HP-UX 11i there are two mechanisms available to an administrator to reduce system dump times: selection and compression. Selection reduces the size of the memory to be dumped. Compression reduces the size of the data that needs to be written to disk. In HP-UX 11i v3 a third mechanism, parallelism, has been added. Parallelism increases the rate at which the data can be written to disk.

The parallelism feature is called "parallel dump" or "concurrency mode" in the HP-UX 11i v3 dump infrastructure. In the initial release of HP-UX 11i v3 parallel dump support is dependent on the platform architecture (HP Integrity versus HP 9000). Parallel dump also has characteristics which are dependent on the type of dump driver, and on the number of CPUs and configured dump devices.

3.1 Requirements

3.1.1 Platform support

In the initial release of HP-UX 11i v3 only HP Integrity servers will support parallel dump. Enabling parallel dump on HP 9000 servers will not be allowed.

3.1.2 Driver Capabilities

I/O support during dump is provided via dump drivers, and each configured dump driver reports its parallelism capabilities to the dump infrastructure. These capabilities are:

- Legacy: new parallelism feature is not supported
- Reentrant: supports parallelism per HBA port
- Concurrent: supports parallelism per dump device

The HP-provided dump drivers have the following capabilities in the initial HP-UX 11i v3 release:

Table 1: Driver Parallelism Capabilities

Dump Drivers	Parallelism Capability
fcd	Concurrent
td, mpt, c8xx, ciss, sasd	Reentrant

3.1.3 Dump Units

A Dump Unit is an independent sequential unit of execution within the dump process. Each dump unit is assigned an exclusive subset of the system resources needed to perform the dump, including CPUs, a portion of the physical memory to be dumped, and a subset of the configured dump devices. The dump infrastructure in HP-UX 11i v3 automatically partitions system resources at dump time into dump units.

Each dump unit operates sequentially. Parallelism is achieved by multiple dump units executing in parallel.

3.1.3.1 Resource Requirements

The following requirements must be met to achieve multiple dump units, and hence parallelism:

- Multiple CPUs:
 - One CPU per dump unit for an uncompressed dump. E.g. to achieve 4-way parallelism (4 dump units) in an uncompressed dump, the system must have at least 4 CPUs.
 - Five CPUs per dump unit for a compressed dump (4 CPUs compressing data and one CPU writing the data to the disks).
- Multiple dump devices:
 - A dump device cannot be shared across multiple dump units. Therefore, to achieve N dump units at least N dump devices must be configured, subject to driver constraints listed below:
 - Devices controlled by legacy dump drivers: multiple “legacy devices” cannot be accessed in parallel, so all such devices will be assigned to a single dump unit.
 - Devices controlled by reentrant dump drivers: multiple “reentrant devices” can be accessed in parallel only if the devices are configured through separate HBA ports. Thus all “reentrant devices” on the same HBA port will be assigned to a single dump unit. To achieve, for example, 4 dump units using reentrant dump devices requires 4 devices each accessible through a separate HBA port.

- Devices controlled by concurrent dump drivers: each “concurrent device” can be accessed in parallel. Each can therefore be assigned to a separate dump unit, even if configured through a single HBA port.
 - Logical volumes configured as dump devices (e.g., in an LVM environment): all logical volumes which reside on the same physical device (LUN) are assigned to the same dump unit.²
 - Shared swap/dump devices: while there is no restriction on the creation of dump units with shared swap/dump devices or volumes, see section 3.4.4 recommending against their use with parallel dump.
- Platform support:
 - Must be Integrity platforms in the initial HP-UX 11i v3 release.

These requirements can be distilled into the following formulas for calculating the number of dump units that can be achieved:

CPU Parallelism = (number of CPUs available at dump time) / (1 or 5, depending on whether or not compression is enabled)

Device Parallelism = (number of reentrant dump HBA ports) + (number of concurrent dump devices) + (1 if there are any legacy dump devices)

Number of Dump Units = Minimum (CPU Parallelism, Device Parallelism)

The term “reentrant dump HBA port” in the Device Parallelism formula is defined in Section 2. The number of reentrant dump HBA ports is the number of HBA ports through which reentrant dump devices are configured.

Examples illustrating parallelism and the use of these formulas are given in the next section.

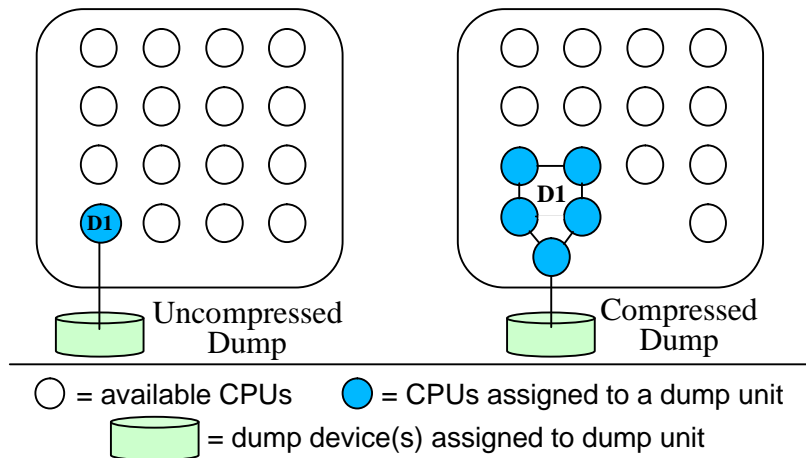
3.2 Parallelism Examples

In the diagrams below each box with 16 circles represents a 16 CPU system. Uncompressed dump units are designated by a single CPU, and compressed dump units by a group of 5 CPUs, using the labels D1, D2, ... to denote the various dump units.

3.2.1 CPUs and Dump Units

Figure 1 illustrates the relationship between CPUs and dump units. In uncompressed dump each dump unit is comprised of one CPU. In compressed dump each dump unit is comprised of five CPUs.

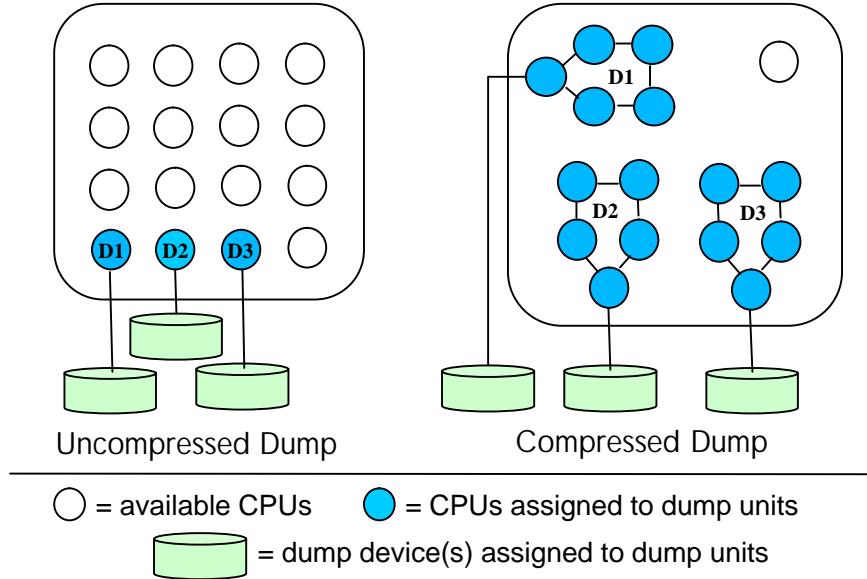
Figure 1 – CPUs per Dump Unit



² It is important to emphasize that configuring multiple dump volumes on a single physical volume will not allow for parallelism. Parallelism at dump time can only be achieved across multiple physical devices (LUNs).

Figure 2 shows the CPUs used in three dump units. Uncompressed dump uses 3 of the 16 CPUs, while compressed dump uses 15 of the 16 available CPUs.

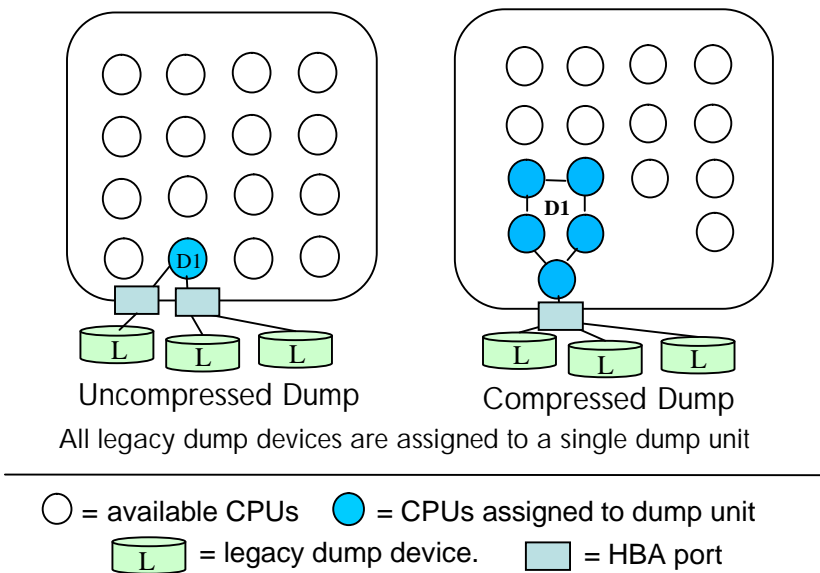
Figure 2 – CPU Usage in Three Dump Units



3.2.2 Devices and Dump Units

Figure 3 illustrates the relationship between devices controlled by legacy dump drivers (“legacy dump devices”) and dump units. Multiple legacy devices cannot be accessed in parallel, so all legacy devices get assigned to a single dump unit.

Figure 3 – Legacy Devices and Dump Units



Figures 4 and 5 illustrate the relationship between devices controlled by reentrant dump drivers (“reentrant dump devices”) and dump units. Multiple reentrant devices can be accessed in parallel if they are on separate HBA ports. In Figure 4 the three reentrant devices are only accessible via a single HBA port and thus get assigned to a single dump unit. In Figure 5 the reentrant devices are accessible via three different HBA ports, allowing three dump units to be created.

Reentrant Devices and Dump Units

Figure 4: reentrant devices only accessible via a single HBA port

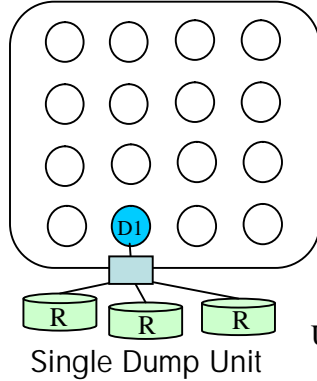
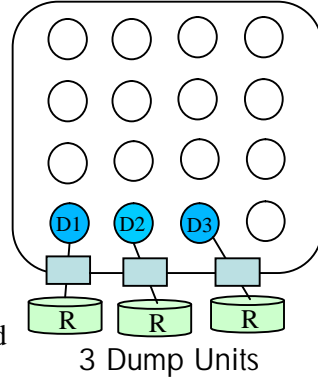


Figure 5: reentrant devices accessible via 3 HBA ports



○ = available CPUs ● = CPUs assigned to a dump unit
 [R] = Reentrant dump device [] = HBA port

Figure 6 and 7 illustrate the relationship between devices controlled by concurrent dump drivers (“concurrent dump devices”) and dump units. Multiple concurrent devices can be accessed in parallel irrespective of which controller ports they are configured under. Each concurrent device can therefore be in a separate dump unit, even if the devices are only accessible through a single HBA port.

Concurrent Devices and Dump Units

Figure 6: concurrent devices, one HBA port

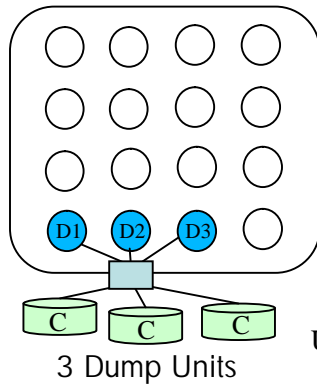
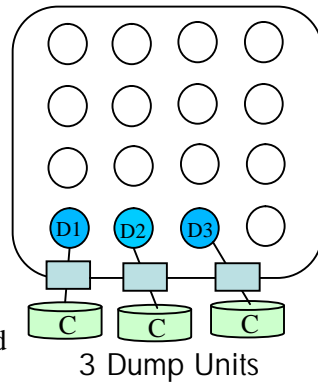


Figure 7: concurrent devices, three HBA ports

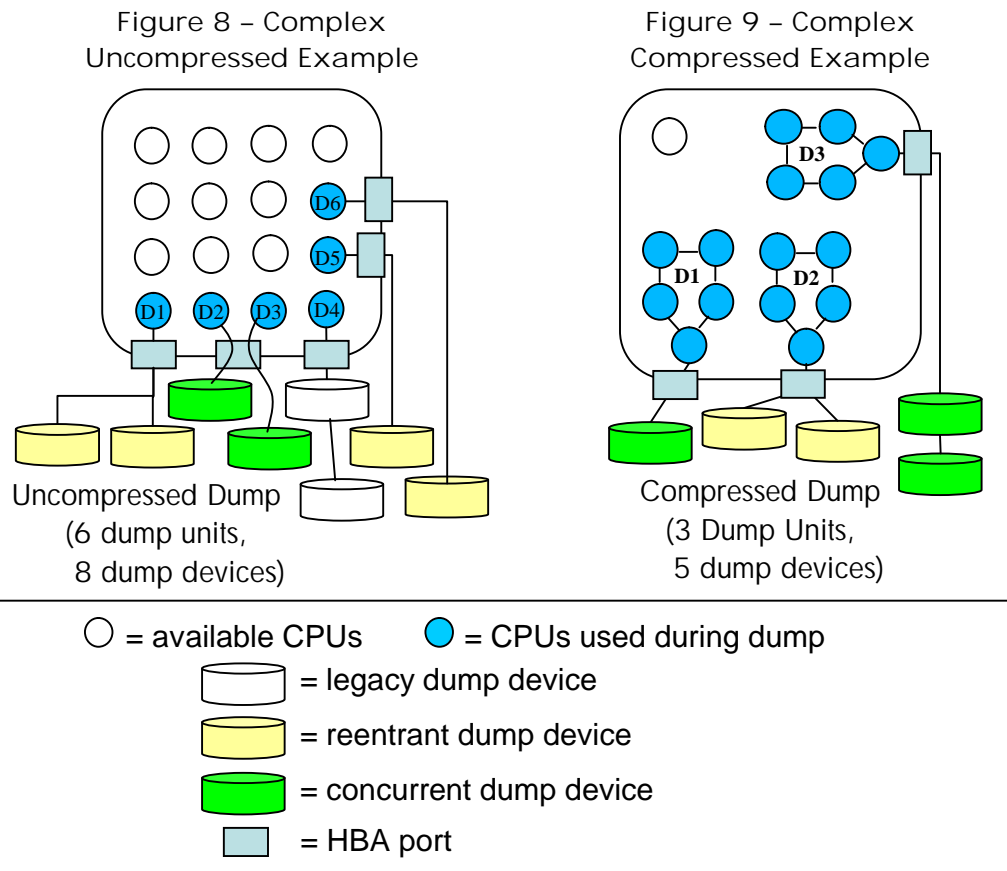


○ = available CPUs ● = CPUs assigned to a dump unit
 [C] = Concurrent dump device [] = HBA port

Figures 8 and 9 combine elements of each of the above examples to illustrate more complex sets of devices and dump units.

Figure 8 shows an uncompressed example with 2 legacy devices, 4 reentrant devices through 3 HBA ports, and 2 concurrent devices. The two legacy devices are assigned to one dump unit; the two concurrent devices each get assigned to an additional dump unit; and the four reentrant devices are assigned to three additional dump units (one for each of their 3 HBA ports). This results in a total of 6 dump units. The Device Parallelism formula in section 3.1.3 can be usefully applied here.

Figure 9 shows a compressed dump with 2 reentrant devices through 1 HBA port, and 3 concurrent devices. In this case the Device Parallelism supports 4 dump units (3 for the concurrent devices + 1 for the reentrant devices), but the CPU Parallelism in a 16-CPU system will only support 3 compressed dump units.



3.3 Automatic HBA Selection

When a dump device is configured a path to the device is automatically selected by the dump infrastructure in a manner which balances the configured dump devices across the available HBA ports. The purpose is to make the HBA assignments in a manner which maximizes parallelism. This occurs at run-time, when a dump device is configured, not at dump time.

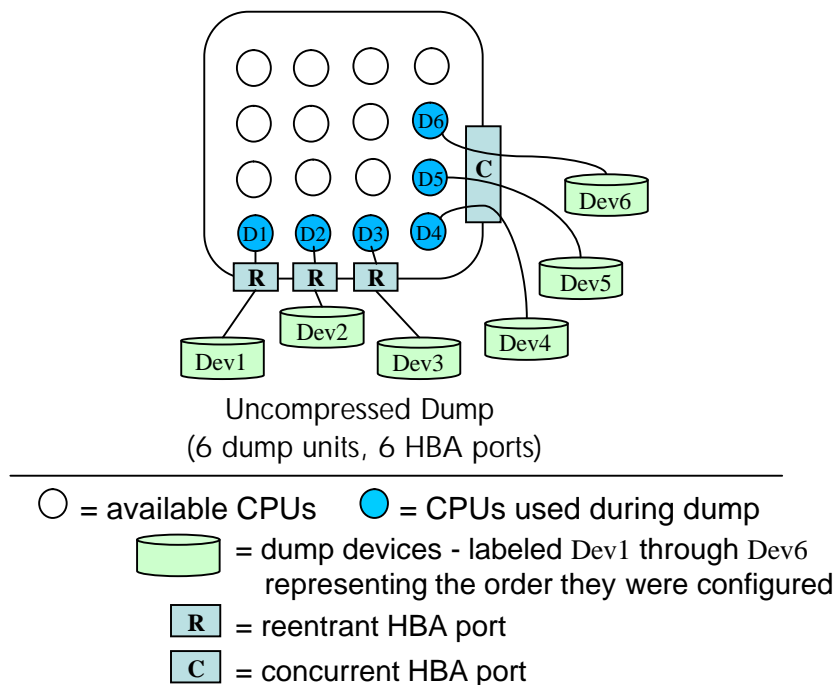
During dump device configuration a path to the device is chosen through an HBA port which has not been used by previously configured dump devices, where possible. If all available HBA ports have been used then an HBA port which has the least number of devices configured on it will be

chosen, with the following exception: concurrent HBAs will always be selected over reentrant HBAs once the available ports have all been used. This exception is due to the fact that having more than one reentrant device on the same HBA port will not increase parallelism.

Offline or disabled paths are not included in the automatic HBA selection. See Section 5, "Availability and Manageability Improvements", for information regarding automatic reconfiguration across path offline events.

Figure 10 illustrates how this would work in an example configuration. In this example there are 6 dump devices and 4 HBA ports, each of which has paths to all 6 of the devices. Three of the HBA ports are reentrant, and one is concurrent. The first 4 dump devices configured will each be assigned a path to a different HBA port. The 5th and 6th devices will be assigned paths through the concurrent HBA, resulting in the 6 dump units shown below.

Figure 10 – Automatic HBA Selection



Figures 11 and 12 illustrate the impact that configuration ordering can have on path selection in some configurations. In this example there are two devices and two HBA ports. The HBA port named `hba1` has paths to both devices, while `hba2` only has a path to `Dev2`. If, as shown in Figure 11, `Dev2` is configured first and `hba1`'s path is selected then when `Dev1` is configured it would also be configured through `hba1` (its only path). The red lines in Figure 11 indicate the configured dump paths, which go through `hba1` for both devices. To maximize parallelism the two devices should be configured on separate HBA ports.³

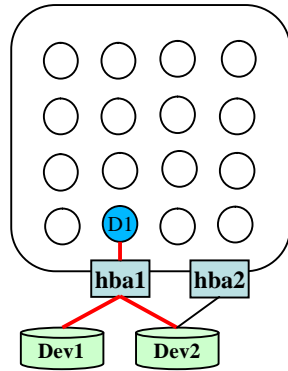
The system administrator can deal with such situations by careful ordering of dump device configurations (configuring `Dev1` before `Dev2`) or by disabling of lun paths⁴ (i.e., disabling the path from `hba1` to `Dev2` in this case) to better balance the dump configuration across the available

³ In the example in Figure 11 only one dump unit would be created if the HBAs were reentrant. If the HBAs were concurrent then two dump units would be created regardless of the configuration order. However, even in the concurrent case it is recommended that the devices be spread across the available HBAs to maximize performance.

HBA ports. This situation is illustrated in Figure 12, in which reordering or disabling produces a dump configuration in which both HBA ports are used.

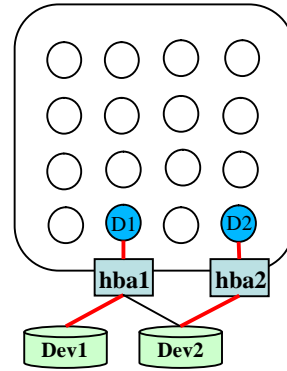
Configuration Ordering in HBA Selection

Figure 11 – Dev2 then Dev1



Configuration of Dev2 before Dev 1 can result in only 1 of the 2 HBA ports being used.

Figure 12 – Dev1 then Dev2



Use of both HBA ports can be accomplished by configuring Dev1 before Dev2, or by disabling the path from hba1 to Dev2.

— (black line) = Unused path
 — (red line) = Configured dump path

Note: The results in Figure 12 can be accomplished by disabling the lun path after the dump configuration is complete, without worrying about ordering or disabling until then. This will cause auto-reconfiguration and selection of a non-disabled path. See section 5.1 for details.

3.4 Performance Guidelines

When creating a dump configuration to optimize for performance, remember that there are three mechanisms for reducing system dump time: selection, compression, and parallelism. Selection can be enabled independently of the other two, and can significantly reduce the size of memory to be dumped and thus the overall dump time. Compression and parallelism can also be independently enabled, but there are interactions and tradeoffs between them since compression requires more CPU resources and can thus limit the available parallelism.

Given a particular level of available parallelism, the actual dump time reduction as the number of dump units increases will depend on a number of factors, including:

- How much hardware overlap there is between paths to devices in different dump units.
- Whether or not compression is enabled.
- Varying device or HBA or link speeds across the different dump units.

Each of these factors is discussed below.

⁴ The `scsimgr(1m)` command can be used to disable a lun path. The specific command to disable a lun path is `scsimgr disable -H <lun_path hw_path>`

Note that this disables the lun path for normal access at run-time, which should be fine for a dedicated dump device but may not be for a device shared with swap or root for example.

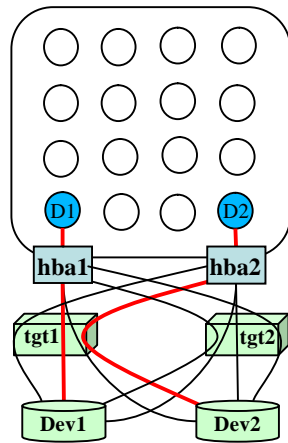
3.4.1 Hardware overlap across dump units

Performance tends to scale better when the hardware overlap (sharing of components such as HBAs, links, targets) between paths to devices in different dump units is minimized.

For example, performance will generally be better when dump devices for separate dump units are configured through separate target ports. Dumps generate large sequential writes which will compete for bandwidth on the link and in the target controller. This issue is similar to the impact of configuration ordering on HBA selection discussed in section 3.3, and can be resolved in a similar manner using lun path disabling. Figures 13 and 14 illustrate how this would work.

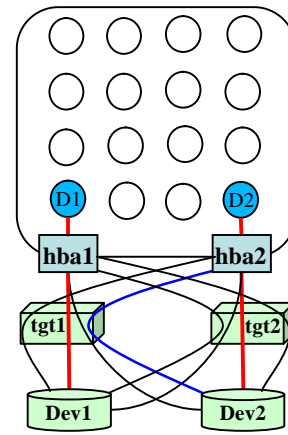
In the configuration example in Figures 13 and 14 there are two devices, two target ports, and two HBA ports. Each HBA port has 2 paths to each device, one through each target. In Figure 13, Dev1 is configured first and a path is automatically selected through hba1, followed by the configuration of Dev2 for which a path through hba2 is selected. The choice of paths through separate HBA ports occurs automatically, as required by the auto-HBA-selection algorithm described in section 3.3. If both auto-selected paths are through the same target (tgt1 in Figure 13) the administrator can disable one of the selected tgt1 paths to balance the dump units across the targets. The result of disabling the hba2-tgt1-Dev2 path is shown in Figure 14. After disabling this path, Dev2 will be auto-reconfigured via the only other hba2 path available to Dev2, which is through tgt2, thus balancing the dump units across the available targets.

Figure 13 - Redundant target configuration



Both dump units are configured with paths through the same target (tgt1).

Figure 14 - Balancing paths across targets



After disabling the hba2-tgt1-Dev2 path each dump unit is configured on a separate target.

-
- (black line) = Unused path
 - (red line) = Configured dump path
 - (blue line) = Disabled path
-

3.4.2 Compression/parallelism tradeoffs

Compressed dump reduces the size of the data in memory by compressing it before writing it to disk. The compression ratio (ratio of the size of data pre-compression to post-compression) depends on the data pattern in memory (e.g., whether the data is fairly random or not). As a result the compression ratio can vary for different memory areas that are dumped, and therefore in parallel dump one dump unit may have a different compression ratio from another. This leads to imbalances in the size of the post-compression data of various dump units, even though the actual memory area to be dumped has been evenly distributed. Since the overall dump time is

gated by the dump unit which takes the longest time to complete the dump of its memory area, the imbalances inherent in compressed dump can reduce the scalability of the parallelism.

For example, on a system with 3 dump units with equal sizes of memory to be dumped and equal I/O rates to the disk, the uncompressed dump time for each dump unit would be equal. I.e., without compression if one dump unit takes 30 minutes to complete the overall dump time will be 30 minutes plus dump startup and completion time. However, with compression the data sizes to be dumped across the three dump units may now be unequal due to varying compression ratios across the different memory areas. This can result in unequal dump times across the dump units, producing dump times, for example, of 3 minutes, 5 minutes, and 7 minutes, respectively, for the three dump units. The overall dump has to wait for the 7 minute dump unit to complete, thus reducing somewhat the scalability of the 3 dump unit parallelism.

In some configurations one may need to consider the tradeoffs between compression and higher levels of parallelism. Given typical compression ratios (and corresponding reduction in dump times) of 5 to 10, this will often be greater than the performance improvement that can be achieved with additional parallelism⁵ without compression. Combining compression, for example, with 2 to 4 dump units would generally be better than uncompressed dump with more dump units. Not to mention the increased complexity of dealing with the large numbers of dump devices that would be needed in the latter case. Gray areas would include systems which only have enough CPUs for one compressed dump unit, e.g. an 8-CPU system. Even in this case compression would generally be recommended because it would give 5 to 10 times the performance of non-compression versus a maximum of 8, assuming perfect scaling, with just parallelism.

3.4.3 Varying device/HBA/link speeds

As mentioned above, the overall dump time in parallel dump is gated by the dump unit of longest duration. Therefore, varying device or HBA or link speeds across the various dump units can result in a reduction of the overall scalability of the parallelism. For example, consider a system with two dump units, each with one-half of system memory to dump, in which one of the dump units takes twice as long to dump its portion of memory as the other. In this case the dump time with parallelism would be 2/3 of the dump time without parallelism (rather than the 1/2 time that can potentially be achieved if both dump units operate at equal speeds).

Note: Using identical sizes and types of dump devices and HBAs in the dump configuration is one way to avoid inequalities in dump speeds or times across the dump units. This will tend to produce more predictable results and better overall parallelism.

3.4.4 Shared swap/dump devices

It is recommended that shared swap and dump devices or volumes not be used with parallel dump. Using a shared swap/dump device can significantly increase the subsequent reboot time because such devices result in swap being disabled while saving the corresponding dump data (eg. in /var/adm/crash). In the case of parallel dump (multiple dump units), if any of the dump devices were shared with swap then swapping will be disabled across the saving of the whole dump (not just the saving of the shared dump/swap devices), which can significantly increase the reboot time.

⁵ Typical compression ratios of 5 to 10 are based on internal HP testing and the results seen on individual systems may vary.

3.4.5 Recommended procedure to configure parallel dump for performance

The following outlines a recommended procedure to configure parallel dump for performance:

1. Identify the available devices and HBA ports for dump.
2. Calculate the maximum CPU parallelism for both compressed and uncompressed dump.
3. Calculate the maximum Device parallelism.
4. Calculate the overall dump parallelism based on steps 2 and 3 for compressed and uncompressed dump. See the formulas at the end of section 3.1.
5. Choose the dump format (compressed or uncompressed) that maximizes the parallelism. Keep in mind the discussions in sections 3.4.1 through 3.4.3.
6. Set up the configuration, if satisfied, or
Go back to step 1 and look at changing the dump configuration or available resources.

3.5 Viewing, Enabling, Disabling of Parallel Dump

3.5.1 Viewing current setting

The current setting of the parallel dump feature can be viewed by running "crashconf -v" (see crashconf(1m)). The output contains the text "Dump Parallel: ON" if the feature is enabled. If the feature is disabled "Dump Parallel: OFF" is displayed in the "crashconf -v" command output. See section 5.2 for example crashconf -v output.

Parallel dump is "on" by default on HP Integrity servers, and is "off" by default on HP 9000 servers. Parallel dump cannot be enabled on HP 9000 servers.

3.5.2 Enabling and Disabling Parallel Dump

Parallel dump can be enabled or disabled at run time by running the crashconf(1m) command with the new "-p" option. When combined with the -t option, the -p setting can be made persistent across reboots.

Alternatively, parallel dump can be enabled or disabled on each boot by setting CRASHCONF_CONCURRENT_DUMP to 1 in the /etc/rc.config.d/crashconf init script. Setting CRASHCONF_CONCURRENT_DUMP to 0 disables the feature. These settings take effect when the script is executed during boot.

3.5.3 Determining the driver capability of a dump device

The dump driver corresponding to a configured dump device can be determined by doing the following:

- (1) `crashconf -l` to obtain the lun path hw path for each configured dump device
- (2) Extract the HBA port's hw path (the portion left of the 1st '.' in the lun path hw path)
- (3) `ioscan -kf | grep '<hba_hwpath> '` (with a space following the hba port hw path)

The driver name will be displayed in the 4th column. See Table 1: Driver Parallelism Capabilities for the capability of each of the HP-provided dump drivers.

4 Configuration Improvements

4.1 Persistent Marking of Dump Devices

In HP-UX 11i v2 there are several mechanisms available for persistent marking of dump devices (causing the dump configuration to persist across reboots). The mechanism that can be used depends on whether the dump device is an LVM or VxVM volume, or a raw device, and includes the following:

- `/stand/system`: dump devices can be specified in the `/stand/system` file as described in the `system(4)` man page.
- `lvinboot`: persistently marks LVM logical volumes as dump devices
- `vxvmbboot`: persistently marks VxVM logical volumes as dump devices

In HP-UX 11i v3, while preserving these mechanisms for backwards compatibility, a new centralized mechanism is provided:

- `crashconf -s`: a new `crashconf` option, `-s`, provides a unified mechanism for persistent marking of dump devices

This new option to `crashconf(1m)` centralizes the dump configuration functionality into a single mechanism, and deprecates the need for the legacy mechanisms.

Note: Using `crashconf -s` puts the dump configuration into “`config_crashconf_mode`” in which only dump devices marked persistent via `crashconf` will be persistently configured. In this case markings via the legacy mechanisms (`/stand/system`, `lvinboot`, `vxvmbboot`) will be ignored. The legacy mode, which is the default, can be re-enabled via the `-o` option. See `crashconf(1m)` for details.

4.2 New/Enhanced command options and tunables

In addition to the new `-s` and `-o` options to `crashconf(1m)`, the following new `crashconf` options are provided in HP-UX 11i v3:

- `-l` new option to display the lun path for each dump device
- `-d` new option to delete dump devices
- `-p` new option to enable/disable parallel dump

The `-t` and `-v` options have been enhanced to control/display the new parallel dump option.

Dump tunables, `dump_compress_on(5)` and `dump_concurrent_on(5)`, can also be used to persistently enable/disable compression or parallel dump. The value of these tunables can be changed or displayed using the `kctune(1m)` command.

By default, parallel dump is enabled on HP Integrity servers, and disabled on HP 9000 servers.

4.3 Device File Naming

In HP-UX 11i v3 a device has several names associated with it: the legacy device file names that correspond to the various paths to the device (as used in versions prior to HP-UX 11i v3),

and a new device file name (known as a “persistent” device special file) introduced in HP-UX 11i v3 that corresponds to the LUN itself. If a dump device is configured using a legacy device file name the dump infrastructure will convert it to the new LUN device file and then choose a lun path as discussed in Section 3.3. See the man page `intro(7)` and the HP-UX 11i v3 Mass Storage Device Naming White Paper for general information and details regarding the HP-UX 11i v3 device file formats.

4.4 HBA Dump Capability

HP-UX 11i v3 provides a new command, `scsimgr(1m)`, which has an option to display whether or not an HBA port is dump-capable. The following example shows the capability of the HBA port addressed by the device file `/dev/c8xx0`:

```
# scsimgr get_attr -a capability -D /dev/c8xx0

          SCSI ATTRIBUTES FOR CONTROLLER : /dev/c8xx0

name = capability
current = Boot Dump
default =
saved =
```

The following example shows the capability of the HBA port at hardware path `2/0/12/1/1`:

```
# scsimgr get_attr -a capability -H 2/0/12/1/1

          SCSI ATTRIBUTES FOR CONTROLLER : 2/0/12/1/1

name = capability
current = Boot Dump
default =
saved =
```

Both of the HBA ports in the above examples are capable of dump. Some HBAs/drivers may not support dump, including 3rd party HBAs for example, so the `scsimgr` command can be used to quickly discover if dump is supported through a given HBA.

4.5 Ignite-UX Recovery

Dump devices which have been marked persistent using the `-s` option of `crashconf(1m)` may need to be reconfigured for purposes of dump after an Ignite-UX recovery operation. After completing the recovery run `crashconf(1m)` to check the dump device configuration and correct it as needed.

5 Availability and Manageability Improvements

Many of the availability and manageability improvements make use of the native multi-pathing provided in HP-UX 11i v3. The native multi-pathing automatically correlates the paths to a LUN and notifies the dump subsystem of path offline and device offline events and other hardware events so that the dump configuration can be automatically adjusted as needed. The events and reconfiguration are available at run time, not at dump time.⁶

5.1 Path Failover and Auto-reconfiguration

At run time, HP-UX 11i v3 supports native multi-pathing for normal I/O operations. At dump time, I/O will go through only one of the selected paths to the device. The path used is normally the one chosen by the infrastructure during dump configuration as discussed in section 3.3. However, if the selected path goes offline or is disabled during run time operation, the dump subsystem will be notified and a different path will be automatically selected (using the HBA selection rules in section 3.3 with respect to the remaining available paths) and the dump device reconfigured. As noted in section 4.2, the `-l` option to `crashconf(1m)` can be used to display the currently configured lun path for each dump device.

This auto-reconfiguration gets invoked only when a currently selected path goes offline or is disabled. Thus, if the currently selected path goes offline or is disabled and then later comes back online or is re-enabled it will not automatically be re-selected.

Note: Dump functionality may be affected if the system administrator removes the system definitions associated with device special files of configured dump devices (e.g., using the `-a` or `-H` options to the `rmsf(1m)` command) instead of disabling the corresponding LUN or LUN path. Prior to removing, the administrator should verify that the device special file is not configured for crash dump.

5.2 Avoiding off-line devices in the dump

When a dump device goes offline at run-time, the dump subsystem is notified. A device can go offline if all the paths available to the LUN go offline or the LUN itself goes offline. The dump subsystem will mark the device as offline and it will not be used while dumping, and dump unit allocation and other operations at dump time will take this into account. If all the dump devices configured have gone off-line, dump will be aborted. The `crashconf(1m)` command's `-v` option has been enhanced to display offline device information to the user. When a dump device goes offline the dump subsystem will also log a message to the syslog file.

The following example shows `crashconf -v` output with an offline dump device:

⁶ If a lun or lun path goes offline at any time after the system crashes, the offline device or path will not be avoided during dump and the dump will fail. However, this is a significant improvement over previous releases of HP-UX in which the failure window was not only after the system crashed, but anytime after the dump device was last manually configured.

```
# crashconf -v
Crash dump configuration has been changed since boot.
```

CLASS	PAGES	INCLUDED IN DUMP	DESCRIPTION
UNUSED	2659456	no, by default	unused pages
USERPG	61331	no, by default	user process pages
BCACHE	19807	no, by default	buffer cache pages
KCODE	4425	no, by default	kernel code pages
USTACK	1074	yes, by default	user process stacks
FSDATA	90	yes, by default	file system metadata
KDDATA	108403	yes, by default	kernel dynamic data
KSDATA	248157	yes, by default	kernel static data
SUPERPG	42473	no, by default	unused kernel super pages

```
Total pages on system:      3145216
Total pages included in dump: 357724
```

```
Dump compressed:    ON
```

```
Dump Parallel:     ON
```

DEVICE	OFFSET(kB)	SIZE (kB)	LOGICAL VOL.	NAME
1:0x000001	1051488	4194304	64:0x000002	/dev/vg00/lvol2
1:0x000000	0	143374740		/dev/disk/disk4 (offline)

```
-----
147569044
```

```
Dump device configuration mode is config_deprecated_mode.
Use crashconf -s option to change the mode.
```

5.3 Device special file redirection

If a dump device goes offline or fails and needs to be replaced, the HP-UX 11i v3 mass storage subsystem will put the new lun in an "Authentication Failure" state and disallow access until the lun has been authenticated via the 'scsimgr replace_wwid' command. A message is logged to syslog informing the user that the replace_wwid operation needs to be performed. At this point the user can also redirect the original device special file to the new disk via either an option on the 'scsimgr replace_wwid' command or via io_redirect_dsf(1m). The dump subsystem will get an event notification of the device file redirection and will then reconfigure dump to work with the new device. If the re-configuration fails, dump will mark the device off-line.

5.4 Disabling of legacy device special files

By default, as discussed in section 4.3, HP-UX 11i v3 has two modes of operation with respect to device special file creation and formats: legacy and a new "persistent" format. If desired the administrator can disable legacy device files via the 'rmsf -L' command (see rmsf(1m) for details). This removes all legacy device files and I/O tree nodes, and disables further creation of legacy nodes and device files.

Note: On a system with multiple boot disks containing different OS versions, if a legacy-disabled HP-UX 11i v3 system crashes and the next boot is on previous release of HP-UX (e.g., HP-UX 11i v2), savecrash will not be able to save the dump. Likewise, if a legacy-enabled HP-UX 11i v3 or pre-11i v3 kernel crashes and the next boot of the system is on a legacy-disabled kernel, savecrash will fail.

In these cases the boot will otherwise succeed, and the dump can still be saved by rebooting an appropriate kernel.

See also the WARNINGS section in the savecrash(1m) man page.

5.5 Expansion/contraction of dump device

The HP-UX I/O subsystem in HP-UX 11i v3 supports online expansion and contraction of the size of luns. The dump subsystem will get event notification when the size of a dump device (full raw disk) expands or contracts. Dump will then dynamically update the internal data structures and at dump time the expanded/contracted device size will be used.

5.6 Additional features

5.6.1 Supporting character device special file

The `crashconf(1m)/crashconf(2)` commands have been enhanced in HP-UX 11i v3 to accept both character as well as block device special files for configuration.

5.6.2 Auto re-configuration of lvm logical volumes

The dump subsystem will get an event notification when attributes of a logical volume change, and will automatically reconfigure the affected dump volumes. For example, when the `lvextend` or `lvreduce` commands are used to extend/reduce the size of an already configured dump volume, dump will be notified and will reconfigure the volume with the modified size.

6 References

http://docs.hp.com/en/8651/HP-UX_Compressed_Dump_For_11iV1.pdf
(Installing and Configuring the Compressed Dump Utility on HP-UX 11i v1)

http://docs.hp.com/en/5434/Cdump_WP.book.pdf
(Compressed Dump White Paper)

<http://docs.hp.com/en/netsys.html#Storage%20Area%20Management>
(HP-UX 11i v3 Mass Storage Device Naming White Paper)

<http://docs.hp.com/en/netsys.html#Storage%20Area%20Management>
(The Next Generation Mass Storage Stack - HP-UX 11i v3 White Paper)

7 For more information

<http://www.hp.com/go/hpux11i>