hp virtual partitions

september 2000

a white paper from Hewlett-Packard Company

table of contents

1.0 introduction

Changing Business Landscape Presents New IT Challenges

Implementing an effective infrastructure for e-services applications in today's dynamic business environment presents a number of new challenges for enterprises and service providers. For most organizations, it has become clear that the "new economy" is fuelled by well-designed infrastructures that match the objectives of the business, are flexible, and support fast time-to-market for new services.

Customers of service providers must not only have access to the precise capacity they need at any point in time, but the infrastructure must be flexible enough to keep that capacity separate from that utilized by other customers. The same holds true for major users of enterprise infrastructures. An effective way to meet this need is to "partition" the base infrastructure of the service provider or enterprise such that each customer's or user's unique capacity and performance demands can be met and measured on an individual basis. One of the major challenges facing these organizations is to manage these diverse workloads while, at the same time, achieve the most effective use of their computing assets to reduce total cost of ownership (TCO).

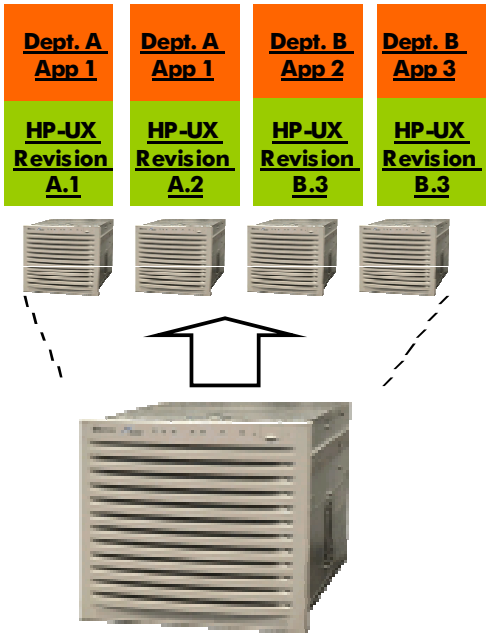
To address this challenge, Hewlett-Packard has created a family of flexible, powerful and far-reaching partitioning solutions—the HP Partitioning Continuum for Always-On infrastructure. The solutions provide hard partitions, virtual partitions, and resource partitions on an individual HP 9000 server node or within the HyperPlex—the HP 9000 Enterprise Server clustering implementation. The combination of HP Partitioning Continuum provides everything that the enterprise or service provider needs to implement a complete solution for partitioning, workload management and application isolation.

2.0 what are vPars?

For effective partitioning, it's necessary to isolate operating environments such that multiple customers' applications can coexist in the same server or cluster, while assuring complete privacy. In addition, it's often necessary to dynamically create, modify or even delete the isolated operating environments on a running server *without interrupting non-related partitions.* To meet this need, HP has developed HP Virtual Partitions—a unique technology that provides application (including name space) and operating systems isolation that runs on single server nodes or nPartitions (hard partitions on Superdome). Available for L-Class, N-Class, and Superdome server nodes, these virtual partitions can be dynamically created using software commands. Each partition runs its own image of the HP-UX 11i operating system (or later) and can host its own applications in a fully isolated environment. Within each virtual partition, up to 64 resource partitions can be created and utilized using solutions such as Process Resource Manager and HP-UX Workload Manager.

HP Virtual Partitions is a powerful, flexible tool that makes it possible to run multiple workloads—each with their unique OS configuration requirements—on the same server at the same time. It's also extremely well suited for making more effective use of underutilized server nodes. Additionally, HP Virtual Partitions are ideal for testing new or enhanced products in a production environment without the need to duplicate the entire environment. **Figure 1** is an example of a single N-Class server running four vPars.

Figure 1. Virtual Partitions within Single N-Class Server



3.0 why use vPars?

The value of vPars is realized as a reduction in the total cost of ownership for a server environment. This reduction in cost is attributable to an increased overall system utilization, increased flexibility in configurations, improved system availability, and improved scalability. The following sections will detail how vPars deliver on the above attributes.

3.1 increased system utilization

Many users of Enterprise class systems complain of utilization rates rarely above 50%. These characteristics are attributable to a variety of reasons but much is dependant upon servers being dedicated to a single application, which rarely fully utilizes the system.

vPars allow an administrator to allocate a subset of the system resources to each partition. Now each partition can run separate instances of the operating system with different OS versions, applications, or users. In general, a vPar will own a specified amount of memory or one or more ranges of physical memory, a specified pool of CPUs, and a set of I/O cards in the server.

Many users deploy multiple servers for reasons other than additional CPU capacity. For example, ISPs deploy small Web servers for customers that need to manage their own content. The importance of the deployment is to maintain customer data isolation in addition to performance. vPars can provide a means for isolating one set of users from another.

3.2 flexibility in configurations with isolation

vPars will allow greater flexibility in configuring servers. For example, many businesses have applications that are cyclic in nature such as, payroll, end of month billing, etc. Many times, the cycles for the applications are not aligned. Normally servers need to be configured to handle peak loads. This can lead to poor utilization as discussed above. With vPars, applications can be allocated to a larger percentage of the system resources during the peak usage times and utilize less resources during off-peak times. This frees up hardware resources that can be assigned to other vPars where the applications may be experiencing a high demand for these resources.

In addition to reconfiguration to cover peak loads, many times operating system adoption rates are slowed because all parts of all solutions that run on the server must be available and qualified before a server can be upgraded. vPars provide a way to do rolling OS and software upgrades on a given server. With vPars, the server can be partitioned into multiple operating system revisions. Applications available on the new OS release can take advantage of the new features. Those applications not yet available on the new OS can be executed on the same system, but run on an older version of the OS.

vPars can be used to set up isolated partitions as test environments. This could be for new revisions of current applications or for deployment of new applications. vPars allow testing on the exact deployment environment. This improves the quality of the test without replicating the cost of the deployment environment.

3.3 improved system availability and capacity

In traditional server environments, all CPUs within a server run the same OS instance and one or more applications. Application and OS failures may affect the entire system. For this reason, running fine-grained vPars can limit the impact of application or OS failures on overall application availability. For improved single system availability vPars allows you to run one application per partition. When a software failure does occur in one of the partitions, the application of that particular partition may be lost, but the rest of the applications on the other partitions continue to run. In fact, even if the OS panics in one of the partitions, applications running on the other vPars are not affected.

One of the inherent problems in a single system is the difficulty in expanding CPU resources when the demands of the application or multiple applications, exceed the servers configuration. Usually the system would need to be shut down and additional CPU's added. With vPars, a large server could have CPUs dynamically moved from one vPar to another without bringing the entire system down. This would allow resources to be moved to vPars with the greatest demands or removed from vPars where they are no longer required.

4.0 vPars features

- Core functionality is available for free with every HP-UX 11i release
- Support of HP 9000 L-Class, N-Class, and Superdome (including nPartitions)
- Support of multiple HP-UX instances (HP-UX 11i and later)
- Different virtual partitions can run different versions of HP-UX
- Single CPU granularity (virtual partition may contain single CPU)
  – L-Class—recommended up to 2 virtual partitions (max. 4)
  – N-Class—recommended up to 4 virtual partitions (max. 8)
  – Superdome—recommended up to 32 virtual partitions (max. 64)
- Dynamic reassignment of CPUs across virtual partitions
- Software fault isolation (application, including name space, and OS isolation)
- Individual reconfiguration and reboot, e.g., for rolling upgrades (virtual partitions don't affect each other)
- Command line interface (in future via GUI)
- Single toggle console
- Compatible with PRM, HP-UX WLM, ServiceControl Manager, and MC/ServiceGuard

5.0 vPars operational overview

To understand how vPars work it is best to compare it to a generic HP-UX server. **Figure 2** shows a 4-way HP-UX server. Without vPars, the entire server would be controlled by a single instance of HP-UX. All of the resources (CPU, Memory, and Disk) would be dedicated to the applications running in this single instance. The software stack for this server would look like the one in **Figure 3**.

Figure 2. Generic HP-UX Server Block Diagram

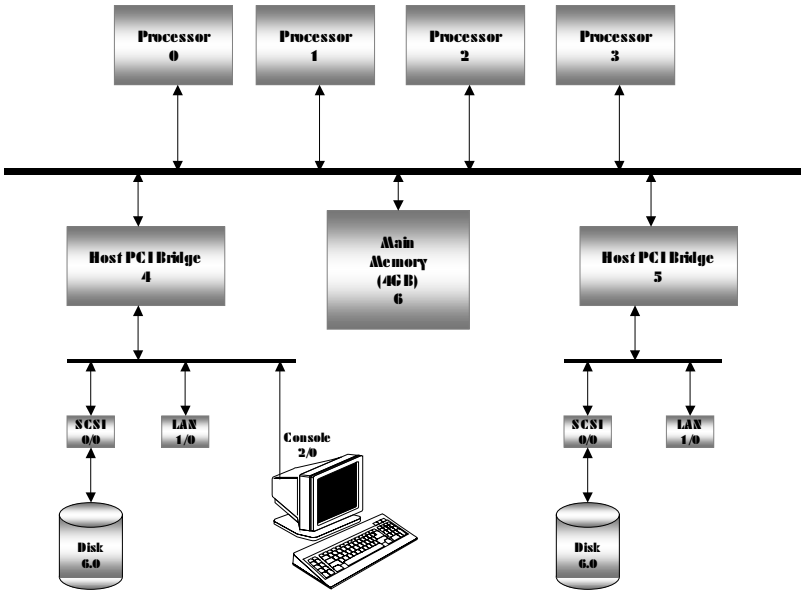
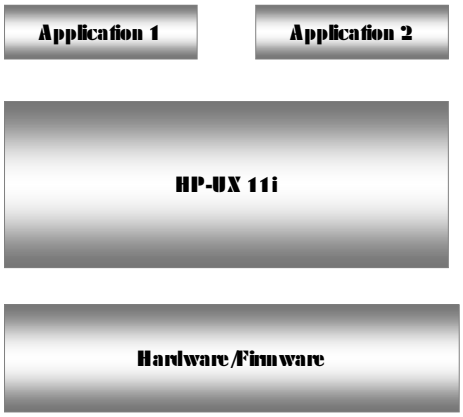


Figure 3. Generic HP-UX Server Software Stack



Using vPars, the server in **Figure 3** can be broken into two partitions, each with a subset of the hardware (**Figure 4**). Each vPar has its own boot disk, at least one CPU, one LAN connection, and enough memory to run HP-UX and applications that are intended to be hosted on this vPar. Since each vPar can run its own copy of HP-UX (potentially at different release versions or patch levels), each is completely isolated from software errors, system panics, etc. A software stack for two vPars is shown in **Figure 5**.

Figure 4. Generic HP-UX Server Block Diagram with Two vPars

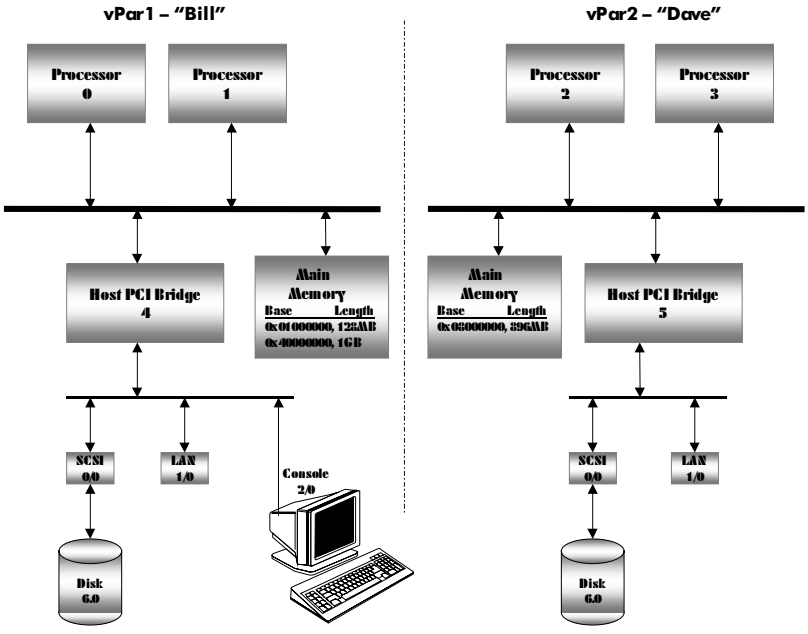
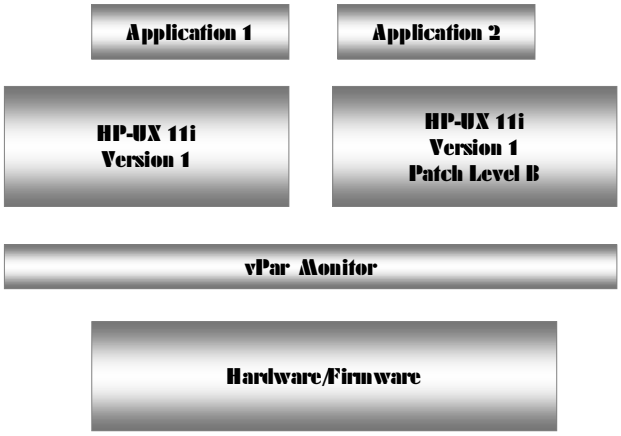


Figure 5. Software Stack with Two vPars



Notice that there is an additional layer of software in the vPar software stack, the Virtual Partition Monitor (vPar Monitor). The monitor manages the partitioning of the resources and creates the illusion for each instance of HP-UX that it is on a standalone system with only the resources that have been dedicated to that vPar. Each instance of HP-UX is completely unaware of the additional hardware in the system. The individual instances of HP-UX have complete ownership of the hardware resources they've been assigned to. The monitor is not involved in accessing I/O hardware or physical memory once it has transferred ownership of the hardware to a vPar.

5.1 virtual partition monitor

The Virtual Partition Monitor, or vPar Monitor, manages the resources, loads kernels, and emulates global platform resources to create the illusion that each individual vPar is a complete HP-UX system. At the heart of the monitor is the partition database that tracks what resources are associated with which vPar. When the vPar Monitor is running, the master copy of this database is kept in the monitor. All changes to the partition database are preserved across system reboots.

In a system running without vPars, the HP-UX kernel is booted directly by the secondary loader from the Initial System Loader (ISL) prompt using a command like:

```
ISL>hpux/stand/vmunix
```

If vPars are used, the vPar Monitor is booted from ISL instead of HP-UX. The monitor will then load the individual vPars. In a system with vPars, this initial load command would look like:

```
ISL>hpux/stand/vpmon
```

The monitor code is loaded from the file /stand/vpmon on the system boot device in the same way as a normal HP-UX kernel would be loaded from the file /stand/vmunix. The monitor loads the partition database from /stand/vpdb and internally creates each vPar according to the resources allocated to each in the database. If there are no command-line options, the monitor is booted in interactive mode with a command-line interface. Once the vPars are up and running, the vPar Monitor is infrequently invoked. The monitor is invoked only when HP-UX makes calls to firmware, when the OS is shutting down, or when vPars management commands are executed.

Commands to create and manage the virtual partitions are described in Managing vPars (**Section 6.0**).

5.2 hardware and software support

The first release of vPars will be supported on L-Class and N-Class Servers running HP-UX 11i. Superdome partitions will be supported shortly after first release. Plans to support other operating systems on an IA-64 platform such as Windows® or Linux are under investigation.

5.3 vPars and CPUs

The recommended number of vPars for HP 9000 servers is up to 2 vPars on an L-Class, up to 4 vPars on an N-Class, and up to 32 vPars on a Superdome (maximum up to 4, 8, and 64, respectively). vPars are also supported within nPartitions of Superdome consisting of multiple cells. Each cell can support up to two vPars. CPUs are not shared or time-sliced between vPars. A vPar can be configured to have just one CPU.

HP Virtual Partitions also provides operational flexibility by allowing you to reassign CPUs from one partition to another—without having to reboot the affected partitions!

5.4 vPars and memory

Memory allocation for vPars is done by specifying either the amount of memory or one or more physical memory ranges. Under no circumstances is sharing of physical memory allowed between multiple vPars. Each vPar will require a minimal amount of memory for booting HP-UX and running the applications.

Dynamic re-assignment of physical pages among vPars without rebooting is being investigated but will not be supported at first release.

5.5 vPars and I/O

The first release of vPars requires that each Local PCI Bus Adapter (top-level PCI buses) is assigned to, at most, one vPar. Therefore, the Local Bus Adapters (LBAs) and the interface cards attached to those LBAs may not be shared among vPars. Sharing of LBAs and interface cards will be investigated for future releases of vPars.

5.6 vPars and console

The system console on a standalone (non-partitioned) server serves two distinct and unique purposes. First, the system console is used to monitor and interact with the Guardian Service Processor (GSP) and Initial System Loader (ISL). Additionally, the system console is used to monitor and interact with HP-UX, particularly when the system is in single user mode or when networking or terminal services are unavailable.

With vPars, the system console continues to be used for interacting with GSP and ISL. However each vPar may designate a separate serial port to be used for its vPar console for monitoring and interacting with HP-UX. Alternatively, a vPar may elect to use a “virtual console.” When configured to use a virtual console, the vPar uses the system console in a multiplexed fashion. When the system console is multiplexed among several vPars, a system administrator with access to the system console may open a console session on any of those vPars.

6.0 managing vPars

There are a set of commands for creating and managing vPars. The definitions of the vPars are stored as a binary file on one or more of the vPar boot disks. The default location is /stand/vpdb on the disk that the monitor image was loaded from—usually the system boot device. Changes made to the partition database are synchronized with the monitor and preserved across reboots. The commands can be issued from any of the active vPars, provided appropriate user permission is granted.

6.1 vPar commands

There are a number of vPar commands available to the system administrator. These commands can be executed from any active vPar. A summary is as follows:

- **vpcreate**—create a vPar, with or without resources
- **vpdestroy**—destroy a vPar, where the partition definition is removed and all resources associated with the given vPar are reclaimed by monitor.
- **vpadd**—add resources to an existing vPar
- **vpremove**—remove resources from an existing vPar
- **vpmodify**—modifies attributes of an existing vPar (such as changing the boot device)
- **vpboot**—start a vPar
- **vpreset**—stop a vPar
- **vpdisplay**—display one or more vPar definitions, including resources and attributes associated with each defined vPar
- **vpstatus**—check status (up or down) of one or more vPars

6.2 vPars with hp-ux management tools

At first release, all vPar commands will need to be performed via a command line interface from an active vPar. The next release of vPars will have support for both SAM (System Administrator Manager) and SCM (ServiceControl Manager). Both these tools will help system administrators adopt vPars and ease both management, and monitoring, of vPar environments.

7.0 vPars and security

Each virtual partition functions like a standalone server and thus is administered via the root user like any other HP-UX server. vPar commands to modify vPars are performed by root. A root user, on any vPar, can run vPar commands for its vPar or any other vPar on the system. With this in mind, certain security features have been incorporated into the vPars implementation to alleviate potential security problems. A summary of these features is as follows:

- Protection against applications in one vPar from intentionally or accidentally stomping on or reading another vPar's memory.
- Resources (such as CPU or Memory) cannot be accessed simultaneously by two vPars. Resources must first be removed from one vPar then added to another.
- Non-root users cannot create, modify, or destroy vPars.
- A resource must be available before it can be added to a vPar.
- Two vPar commands can run simultaneously from two different vPars, but all access to the monitor will be serialized.
- The monitor ensures that the partition database is synchronized with the monitor's internal state.

8.0 vPars and system isolation

8.1 vPars and software isolation

In traditional server environments, all CPUs within a server run a single OS instance and one or more applications. Application and OS failures may affect the entire system.

With HP Virtual Partitions, however, the software stack contains the vPar Monitor between the hardware/firmware level and the OS, allowing each partition to support a different instance of HP-UX 11i (or later) with each instance capable of being at a different version and patch level.

Consequently, if any virtual partition crashes, only that particular virtual partition is affected—providing complete software isolation. This isolation of OS at the version and patch level also enables rolling OS and application upgrades.

8.2 vPars and hardware isolation

There are shared hardware components on a system or Superdome hardware partition, whose failure could bring down the entire server or hard partition, including all the vPars. Virtual Partitions are not necessarily isolated from the failure of a hardware resource that is bound to another vPar. Increased isolation of some hardware failures to a single vPar is being considered for future release.

A high-availability system configuration requires at least two paths for each vPar to any critical device. Special care needs to be taken when designing the highest availability within a vPar environment. MC/ServiceGuard can be used to improve application availability.

9.0 compatibility with different solutions

9.1 vPars and instant capacity on demand (iCOD)

Instant Capacity on Demand (iCOD) is an option available on many of the HP 9000 servers. This allows the customer to purchase a fully populated CPU configuration for their server, but only activate a subset of CPUs initially. Additional CPUs can be "turned on" via a command line interface.

Together iCOD and vPars offer unique configuration opportunities for system administrators on HP 9000 servers. At first release, coexistence of both iCOD and vPars on a single server will not be supported; however, support is planned for a future release.

9.2 vPars and mc/serviceguard

MC/ServiceGuard will be supported on vPars. However, because of the various vPar configurations on HP 9000 servers and Superdome hardware partitions, careful design criteria will need to be followed. By first release of vPars, the MC/ServiceGuard Support Organization will supply recommended guidelines when using vPars in mission critical implementations.

9.3 vPars and PRM and hp-ux WLM

HP's resource partitioning solutions—PRM and HP-UX WLM—enables dynamic and goal-based resource management respectively. These solutions allow one to distribute system resources among different workloads on a single system or a single OS instance. HP PRM and HP-UX WLM are compatible with vPars, providing an additional degree of flexibility and control within a virtual partition.

10.0 vPar example (commands are subject to change)

The following example shows an abbreviated procedure for creating and modifying a three-vPar configuration on an 8-way N-Class server.

The sample procedures will assume that we are creating three vPars on an 8-way N-Class server as depicted by the block diagram shown in **Figure 6** and that we wish to partition the resources as shown in **Table 1**.

Figure 6. Block Diagram of an N-Class Server

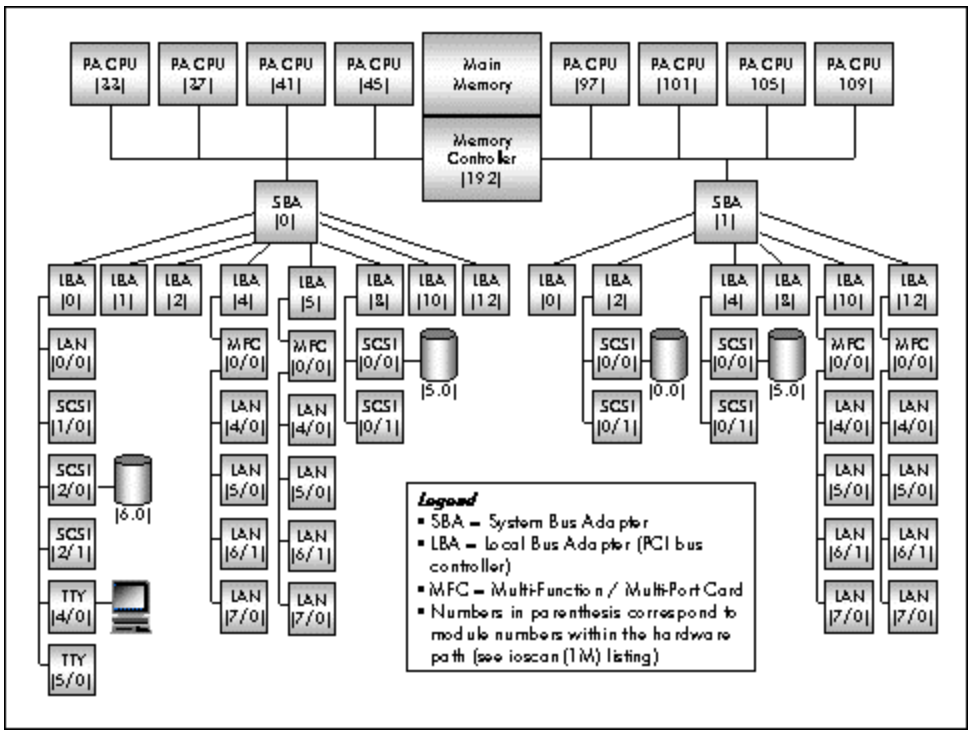
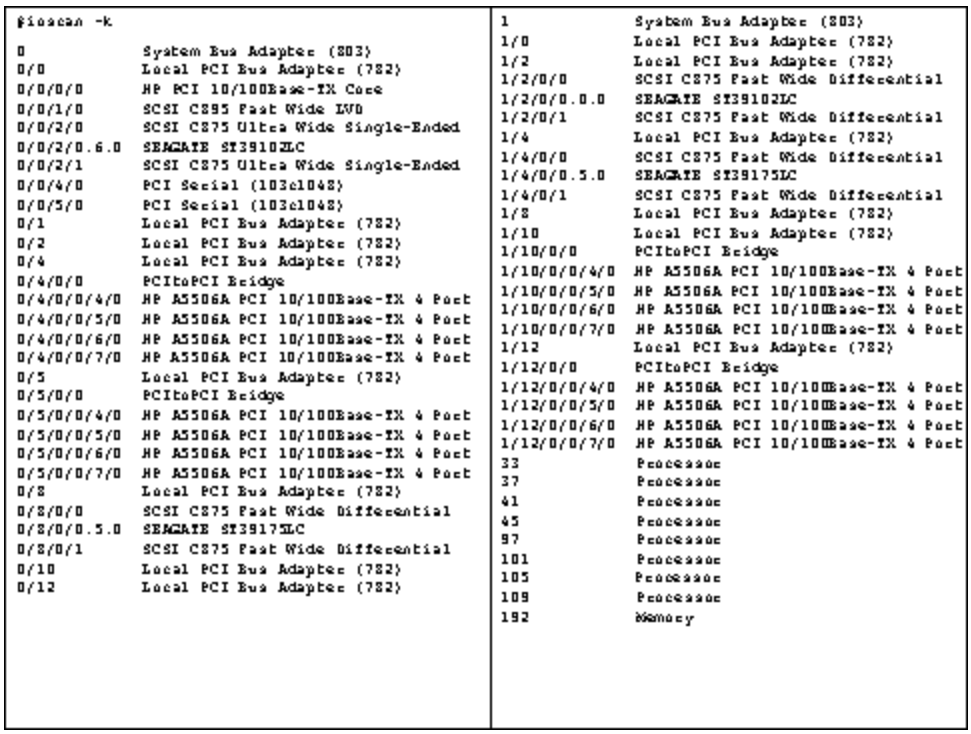


Figure 7. ioscan (1M) Listing of an N-Class Server



```
# ioscan -k
0               System Bus Adapter (803)
0/0             Local PCI Bus Adapter (782)
0/0/0/0         HP PCI 10/100Base-TX Core
0/0/1/0         SCSI C895 Fast Wide LVD
0/0/2/0         SCSI C875 Ultra Wide Single-Ended
0/0/2/0.6.0     SEAGATE ST3910LC
0/0/2/1         SCSI C875 Ultra Wide Single-Ended
0/0/4/0         PCI Serial (103c1048)
0/0/5/0         PCI Serial (103c1048)
0/1             Local PCI Bus Adapter (782)
0/2             Local PCI Bus Adapter (782)
0/4             Local PCI Bus Adapter (782)
0/4/0/0         PCItoPCI Bridge
0/4/0/0/4/0     HP A5506A PCI 10/100Base-TX 4 Port
0/4/0/0/5/0     HP A5506A PCI 10/100Base-TX 4 Port
0/4/0/0/6/0     HP A5506A PCI 10/100Base-TX 4 Port
0/4/0/0/7/0     HP A5506A PCI 10/100Base-TX 4 Port
0/5             Local PCI Bus Adapter (782)
0/5/0/0         PCItoPCI Bridge
0/5/0/0/4/0     HP A5506A PCI 10/100Base-TX 4 Port
0/5/0/0/5/0     HP A5506A PCI 10/100Base-TX 4 Port
0/5/0/0/6/0     HP A5506A PCI 10/100Base-TX 4 Port
0/5/0/0/7/0     HP A5506A PCI 10/100Base-TX 4 Port
0/8             Local PCI Bus Adapter (782)
0/8/0/0         SCSI C875 Fast Wide Differential
0/8/0/0.5.0     SEAGATE ST39175LC
0/8/0/1         SCSI C875 Fast Wide Differential
0/10            Local PCI Bus Adapter (782)
0/12            Local PCI Bus Adapter (782)
1               System Bus Adapter (803)
1/0             Local PCI Bus Adapter (782)
1/2             Local PCI Bus Adapter (782)
1/2/0/0         SCSI C875 Fast Wide Differential
1/2/0/0.8.0     SEAGATE ST39102LC
1/2/0/1         SCSI C875 Fast Wide Differential
1/4             Local PCI Bus Adapter (782)
1/4/0/0         SCSI C875 Fast Wide Differential
1/4/0/0.5.0     SEAGATE ST39175LC
1/4/0/1         SCSI C875 Fast Wide Differential
1/8             Local PCI Bus Adapter (782)
1/10            Local PCI Bus Adapter (782)
1/10/0/0        PCItoPCI Bridge
1/10/0/0/4/0    HP A5506A PCI 10/100Base-TX 4 Port
1/10/0/0/5/0    HP A5506A PCI 10/100Base-TX 4 Port
1/10/0/0/6/0    HP A5506A PCI 10/100Base-TX 4 Port
1/10/0/0/7/0    HP A5506A PCI 10/100Base-TX 4 Port
1/12            Local PCI Bus Adapter (782)
1/12/0/0        PCItoPCI Bridge
1/12/0/0/4/0    HP A5506A PCI 10/100Base-TX 4 Port
1/12/0/0/5/0    HP A5506A PCI 10/100Base-TX 4 Port
1/12/0/0/6/0    HP A5506A PCI 10/100Base-TX 4 Port
1/12/0/0/7/0    HP A5506A PCI 10/100Base-TX 4 Port
33              Processor
37              Processor
41              Processor
45              Processor
97              Processor
101             Processor
105             Processor
109             Processor
192             Memory
```

Table 1. vPar Definition for 8-Way N-Class Examples

vPar Number	**0**	**1**	**2**
vPar Name	Bergen	Oslo	Trondheim
Bound CPU Hardware Paths	33, 37	41, 45	105, 109
Physical Memory Ranges	0x01000000 (112MB) 0x40000000 (512MB)	0x08000000 (128MB) 0x60000000 (1024MB)	0x10000000 (128MB) 0xA0000000 (1024MB)
I/O Hardware Paths	0/0/* 0/4/*	0/8/* 1/10/*	0/5/* 1/4/*
Special Devices	Boot: 0/0/2/0.6.0 Console: 0/0/4/0 LAN: 0/0/0/0	Boot: 0/8/0/0.5.0 Console: Virtual LAN: 1/10/0/0/4/0	Boot: 1/4/0/0.5.0 Console: Virtual LAN: 0/5/0/0/4/0
Kernel Image	/stand/vmunix	/stand/vmunix	/stand/vmunix

10.1 installing vPars on an existing system

1. Plan the vPar resource partitioning and install hardware.

For our example, this is done in **Table 1** (shown on the preceding page). “Bergen” will own the entire core I/O and therefore will use the built-in disk at 0/0/2/0.6.0 as its boot device and the built-in LAN at 0/0/0/0. “Oslo” will have a SCSI card installed in slot #4 (hardware path 0/8/0/0) for its boot disk and a network adapter card installed in slot #9 (hardware path 1/10/0/0). “Trondheim” will have a SCSI card installed in slot #9 (hardware path 1/4/0/0) for its boot disk and a network adapter in slot #1 (hardware path 0/5/0/0).

2. Install HP-UX on one vPar boot disk

Using normal HP-UX installation procedures, install a copy of HP-UX 11i, and vPar product patches, on one of the boot disks. Any of the disk formats supported by HP-UX may be used.

3. Create the vPar partition database

Since the primary boot path for the system corresponds to “Bergen’s” boot disk, we will create the partition database there. With the system booted on “Bergen’s” boot disk, we execute the following commands to create **/stand/vpdb**:

```
# vpcreate Bergen –C –P 33 –P 37 –M 0x01000000:112MB\
–M 0x40000000:512MB –i 0/0/* -i 0/4/* –i 0/0/2/0.6.0:boot\
–i 0/0/4/0:console –k /stand/vmunix –A auto:on

# vpcreate Oslo –C –P 41 –P 45 –M 0x08000000:128MB\
–M 0x60000000: 1024MB –i 0/8/* –i 1/10/* –i 0/8/0/0.5.0:boot\
–k /stand/vmunix –A auto:on

# vpcreate Trondheim –C –P 105 –P 109 –M 0x10000000:128MB\
–M 0xA0000000:1024MB –i 0/5/* –i 1/4/* –i 0/4//0/0.5.0:boot\
–k /stand/vmunix –A auto:on
```

The syntax for the above example is as follows:

-C	(Create database file if it doesn't exist)
-P 33–P 37	(Add processors at hardware paths 33 and 37)
-M 0x01000000:112MB	(Physical memory range, base:size)
-i 0/0/*	(Add I/O device at specific hardware path)
-i 0/0/2/0.6.0:boot	(boot device for vPar)
-i 0/0/4/0:console	(console device for vPar)
-k /stand/vmunix	(kernel image to boot)
-A auto:on	(autoboot attribute set to on)

4. Reboot the system to the vPar monitor

Reboot "Bergen," interrupting the boot process at ISL. At the ISL prompt, instead of launching **/stand/vmunix**, launch **/stand/vpmon** and have vpmon launch "Bergen":

```
ISL>hpux/stand/vpmon loadvp Bergen
```

The secondary loader (hpux) will locate, load and launch the vPar monitor (vpmon) which will then locate, load and launch "Bergen's" kernel (vmunix).

5. Install the remaining vPars

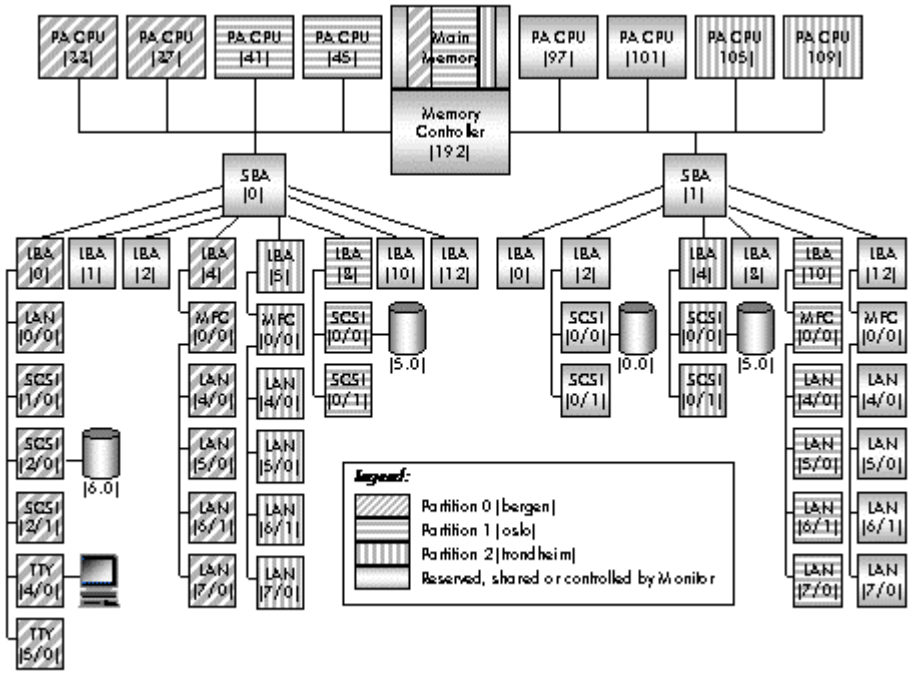
From "Bergen's" command line, boot the remaining vPars from the Ignite/UX install media:

```
# vpboot oslo –I- <IgniteServer>

# vpboot trondheim –I <IgniteServer>
```

Figure 8 shows a block diagram for the resulting partitioning.

Figure 8. Block Diagram of an N-Class Server with Three vPars



10.2 boot the vPar monitor

Booting the vPar Monitor is very similar to booting HP-UX. Instead of specifying **/stand/vmunix** on the command line or in the AUTO file, **/stand/vpmon** is specified.

To launch “Oslo” from the ISL prompt, the following command can be used:
`ISL> hpux/stand/vpmon loadvp Oslo`

To launch all vPars from the ISL command line, use the –all option to the **loadvp** command:
`ISL> hpux/stand/vpmon loadvp -all`

10.3 migrating CPUs between running vPars

Migrating CPUs among running vPars is supported in first release. A vPar must be brought down before changing other resources (such as memory).

In terms of CPU migration, the first release implements the concept of “bound” and “floating” CPUs. This is required since HP-UX does not currently have the capability of reassigning I/O interrupts dynamically between CPUs. CPUs that have I/O interrupts assigned to them are called “bound” CPUs because they are bound to a given vPar. CPUs not bound to any vPar are called “floating” CPUs and can be temporarily assigned to a vPar via **vpadd**.

Notice that in defining “Bergen”, “Oslo” and ”Trondheim” we allocated two CPUs to each as bound CPUs. Since it is an 8-way system, there are 2 CPUs left that are floating. As long as the configuration of a vPar is not marked as static (via the static attribute for the **vpcreate** or **vpmodify** commands), the floating CPUs can be added or removed to a vPar without rebooting.

To add a floating CPU to “Oslo”:

```
# vpadd Oslo –p 1
```

The –p option is used to specify that a single CPU is to be added, but the monitor chooses which CPU to use. Conversely, to remove a CPU from “Oslo”:

```
# vpremove Oslo –p 1
```

The commands will return an error if there are no more CPUs to add or remove. The top (1m) or glance commands can be used to see which CPUs are active in a vPar.

10.4 displaying resources associated with a vPar

vpdisplay shows all resources associated with a vPar. With the –a option, **vpdisplay** lists the resources for all defined vPars. Here's what the output looks like for "Bergen", "Oslo" and "Trondheim":

```
# vpdisplay –a

VP: Bergen
Attributes: AUTOBOOT
Bound CPU: 33
Bound CPU: 37
Memory range: Base:0x01000000 (112MB)
Memory range: Base:0x40000000 (512MB)
I/O: 0.0.*
I/O: 0.4.*
I/O: 0.0.2.0.6.0 BOOT
I/O: 0.0.4.0 CONSOLE
Kernel Image: /stand/vmunix

VP: Oslo
Attributes: AUTOBOOT
Bound CPU: 41
Bound CPU: 45
Memory range: Base:0x08000000 (128MB)
Memory range: Base:0x60000000 (1024MB)
I/O: 0.8.*
I/O: 1.10.*
I/O: 0.8.0.0.5.0 BOOT
Kernel Image: /stand/vmunix

VP: Trondheim
Attributes: AUTOBOOT
Bound CPU: 105
Bound CPU: 109
Memory range: Base:0x10000000 (128MB)
Memory range: Base:0xA0000000 (1024MB)
I/O: 0.5.*
I/O: 1.4.*
I/O: 1.4.0.0.5.0 BOOT
Kernel Image: /stand/vmunix
```

10.5 examining the status of a vPar

The **vpstatus** command is used to determine the state of other vPars. With no arguments, **vpstatus** indicates whether the current system is running in a vPar or not. To determine the state of all vPars, the –a option is used. The status of a specific vPar can be obtained by specifying it on the command line. **vpstatus** reports the following states: up, down, hung and crashing. “up” means that the monitor has launched it and has not detected a crash or hang. “hung” means that the vPar has stopped issuing heartbeats to the monitor. “crashing” means that the vPar has notified the monitor that it is going down ungracefully (panic or reset). “down” means that its resources have been returned to the monitor and the vPar is ready to be booted.

```
# vpstatus –a

0  Bergen : up
1  Oslo : up
2  Trondheim : down
```

10.6 resetting a hung vPar

Just as it is occasionally necessary to issue a hard reset or TOC command (soft reset) for an HP-UX system, it is occasionally necessary to reset a vPar that is hung.

To issue a hard reset on a system, the administrator types a CNTL-B at the console to connect to a service processor and then types the command “rs” to initiate the hard reset. This still works with vPars, but it resets the entire system including the monitor. Using **vpreset**, a vPar can be sent an emulated hard reset as follows:

```
# vpreset Bergen –h
```

The –h causes an emulation of a hard reset. It also has the side effect of overriding the current autoboot setting for that vPar—therefore, the vPar must be manually restarted via **vpboot**.

To issue a soft reset (TOC) on a system, a “tc” command is sent to the service processor instead of “rs”. A soft reset allows HP-UX to attempt to capture some state and potentially create a crash dump. If a TOC is issued on a system with vPars, a system soft reset is initiated and no vPar state is captured. A vPar can be soft reset using **vpreset** without the –h option. As with a TOC on a non-vPar system, HP-UX gets a change to capture state and save a crash dump. That vPar then either shuts down or reboots according to the setting of the autoboot attribute for that vPar.

Other vPars are unaffected when a vPar is reset.

11.0 related information

The following list of references provides useful background information on related products and topics:

- **vPar Product Information**
  http://www.hp.com/go/servicecontrol
- **vPar User's Guide**
  http://docs.hp.com/hpux/
- **Instant Capacity on Demand (iCOD)**
  http://www.hp.com/go/icod

For more information, contact any of our worldwide sales offices or HP Channel.

For the location of the nearest sales office call:
United States of America: +1 800 637 7740
Canada:
Hewlett-Packard Ltd.
5150 Spectrum Way
Mississauga, Ontario L4W 5G1
+1 905 206 4725
Japan:
Hewlett-Packard Japan, Ltd.
Japan Country H.Q.
3-29-21, Takaido-Higashi, Suginami-ku,
Tokyo, 160-8585 Japan
+81 3 3331 6111
Latin America:
Hewlett-Packard
Latin American Region Headquarters
Waterford Building, 9th Floor
5200 Blue Lagoon Drive
Miami, Florida 33126 USA
+1 305 267 4220
Refer to country phone numbers
Australia/New Zealand:
Hewlett-Packard Australia Ltd.
31-41 Joseph Street
Blackburn, Victoria 3130
Australia (A.C.N. 004 394 763)
+61 3 9272 2895
Asia Pacific:
Hewlett-Packard Asia Pacific Ltd.
17-21/F, Shell Tower
Times Square
1 Matheson Street
Causeway Bay
Hong Kong
+8522 599 7777
Europe/Africa/Middle East:
Hewlett-Packard S.A.
150, Route du Nant-d'Avril
CH-1217 Meyrin 2
Geneva, Switzerland
+41 22 780 81 11
European Multicountry: +41 22 780 81 11
Middle East and Africa: +41 22 780 71 11
European Headquarters: +41 22 780 81 81
Refer to country phone numbers

For direct country contact call:

Argentina: +541 787 7145

Austria: +43 1 25 000 0

Belgium and Luxembourg: +32 2 778 31 11

Brazil: +5511 7296 8000

Chile: +562 203 3233

Colombia: +571 629 5030

Denmark: +45 45 99 10 00

East Central Europe, CIS, and Yugoslavia: +43 1 25 000 0

Finland: +358 9 887 21

France: +33 1 69 82 60 60

Germany: +49 7031 140

Greece: +30 1 689 644

Hungary: +36 1 252 7300

Iceland: High Performance Systems hf. +354 1 67 10 00

Ireland: +353 1 615 8200

Israel: Computation and Measurement Systems (CMS) Ltd. +972 3 5380 333

Italy: +39 2 92122770

Mexico: +525 326 4600

Netherlands: +31 20 547 6911

Norway: +47 22 7356 00

Poland: +48 22 608 77 00

Portugal: +351 1301 7343

Russia and the CIS, excl. Ukraine: +7 095 923 5001

Slovenia: +38 61 55 84 72

Spain: +34 1 631 1600

Sweden: +46 8 444 2000

Switzerland: +411 735 7111

South Africa: Hewlett-Packard South Africa (Pty) Ltd.+27 11 806 1000

Turkey: +90 212 224 5925

United Kingdom: +44 1344 369231

Venezuela: +582 239 4133

Full information on HP products is available at **www.hp.com**

M0900

invent