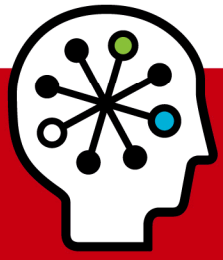
HP Serviceguard Cluster Configuration for HP-UX 11i or Linux Partitioned Systems

April 2009



Abstract 2

Partition Configurations 2

Serviceguard design assumptions 4
- Hardware redundancy 4
- Cluster membership protocol 4
- Quorum arbitration 5

Partition interactions 6

Cluster configuration considerations 7
- Quorum arbitration requirements 7
- Cluster configuration and partitions 9
- Cluster in a box 9
- I/O considerations 10
- Latency considerations for vPars 11
- Latency considerations for Integrity Virtual Machines 12

Other Linux differences 12

Summary and conclusion 12

For more information 13

Abstract

HP Serviceguard provides an infrastructure for the design and implementation of highly available HP-UX or Linux clusters that can quickly restore mission-critical application services after hardware or software failures. To achieve the highest level of availability, clusters must be configured to eliminate all single points of failure (SPOFs). This requires a careful analysis of the hardware and software infrastructure used to build the cluster. Partitioning technologies such as Superdome nPartitions, available on HP-UX 11i v2 or HP-UX 11i v3 or Linux, the HP-UX Virtual Partitions (vPars) and HP Integrity Virtual Machines (Integrity VM) present some unique considerations when utilizing them within a Serviceguard configuration. This document discusses these considerations.

Serviceguard on HP-UX can be used on clusters up to 16 nodes. The nodes within a single cluster can be HP Integrity servers, HP 9000 servers, or combinations of both. For details on specific versions on each server type supported for rolling upgrade refer to the Serviceguard Release Notes.

While not addressed by this white paper, related high -availability products supported on HP-UX 11i v2 and HP-UX 11i v3 include:

- Serviceguard Extension for RAC
- Serviceguard Extension for Faster Failover (Obsolete in Serviceguard 11.19)
- Serviceguard Extension for SAP
- Enterprise Cluster Master Toolkit
- Serviceguard Quorum Service
- Serviceguard Manager

Serviceguard for Linux is certified on Red Hat Enterprise Linux and SUSE Enterprise Server for clusters up to 16 nodes. The nodes within a single cluster can be HP Integrity or ProLiant servers. While this white paper refers to Superdome servers, for Linux, the nPartition configurations and restrictions apply to other HP Integrity servers with nPartition capability. Any differences for Linux are detailed in this white paper. Configurations, restrictions, etc. that are the same as HP-UX are not identified.

While not addressed by this white paper, related high -availability products supported on Linux include:

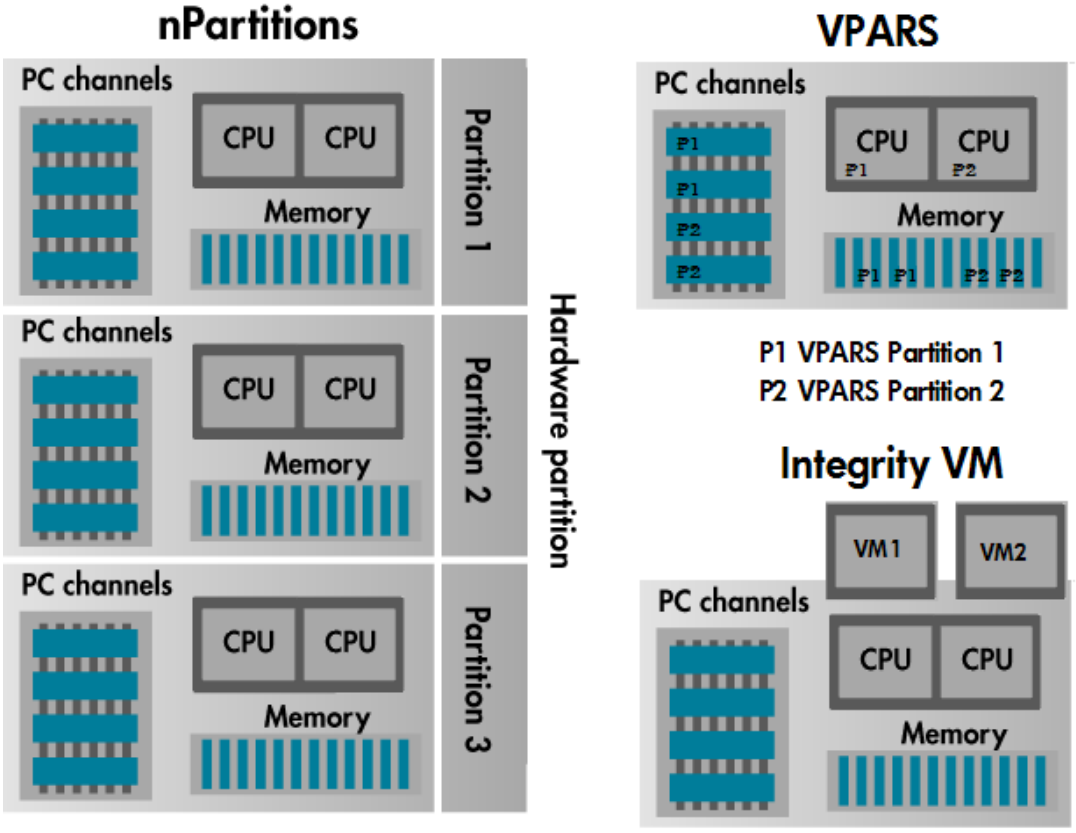
- Serviceguard Extension for SAP for Linux
- Serviceguard for Linux Oracle® toolkit
- Serviceguard Quorum Service
- Serviceguard Manager

Partition configurations

Partitioning technologies such as nPartitions, vPars, and Integrity VM increase the flexibility and effectiveness of managing system resources. They can be used to provide hardware and/or software fault isolation between applications sharing the same hardware platform. These technologies also allow hardware resources to be more efficiently utilized based on application capacity requirements, and they provide the means to quickly redeploy the hardware resources should the application requirements change. Given this capability, it is natural to want to utilize these technologies when designing Serviceguard clusters. Care must be taken, however, as the use of partitioning does present some unique failure scenarios that must be considered when designing a cluster to meet specific uptime requirements.

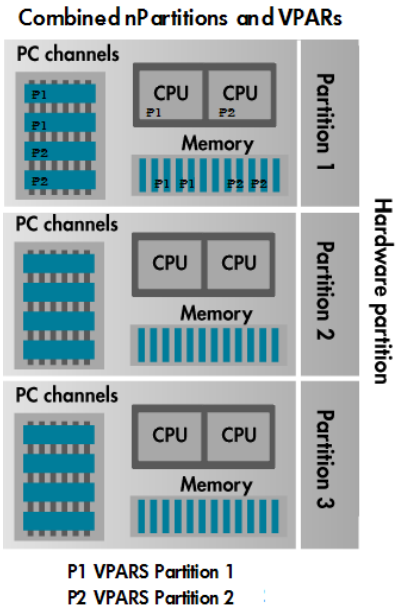
The partitioning provided by nPartitions—available in HP-UX 11i v2 and HP-UX 11i v3—is done at a hardware level and each partition is isolated from both hardware and software failures of other partitions. vPars partitioning and Integrity VM are implemented at a software level. This provides greater flexibility in dividing hardware resources as shown in Figure 1.

Figure 1. Sample nPartitions, VPARS, and Integrity VM configurations



vPars or Integrity VM can be combined with nPartitions to create a more complex configuration. This means that vPars software partitions or Integrity VM can be configured within the context of a hardware nPartition. Figure 2 illustrates an example of a configuration where an nPartition, hardware partition 1, contains two vPars.

Figure 2. Sample of combined nPartitions and vPars configurations



Serviceguard design assumptions

To best understand issues related to using partitioning within the cluster, it is helpful to start with a review of the Serviceguard design philosophy and assumption, including hardware redundancy, cluster membership protocol, and quorum arbitrations.

Hardware redundancy

Serviceguard, like all other high-availability (HA) clustering products, uses hardware redundancy to maintain application availability. For example, the Serviceguard configuration guidelines require redundant networking paths between the nodes in the cluster. This requirement protects against total loss of communication to a node if a networking interface card fails. If a card should fail, there is a redundant card that can take over for it.

As can be readily seen, this strategy of hardware redundancy relies on an important underlying assumption: the failure of one component is independent of the failure of other components. If the two networking cards were somehow related, a single failure event could disable them both. This represents a SPOF and effectively defeats the purpose of having redundant cards. It is for this reason that the Serviceguard configuration rules do not allow both heartbeat networks on a node to travel through multiple ports on the same multi-ported networking interface. A single networking interface card failure would disable both heartbeat networks.

Cluster membership protocol

This same philosophy of hardware redundancy is reflected in the clustering concept. If a node in the cluster fails, another node is available to take over applications that were active on the failed node. Determining with certainty which nodes in the cluster are currently operational is accomplished through a cluster membership protocol whereby the nodes exchange heartbeat messages and maintain a cluster quorum.

After a failure that results in loss of communication between the nodes, active cluster nodes execute a cluster re-formation algorithm that is used to determine the new cluster quorum. This new quorum, in conjunction with the previous quorum, is used to determine which nodes remain in the new active cluster.

The algorithm for cluster re-formation generally requires a cluster quorum of a strict majority—more than 50% of the nodes that were previously running. However, exactly 50% of the previously running nodes are allowed to re-form as a new cluster, provided there is a guarantee that the other 50% of the previously running nodes do not also re-form. In these cases, some form of quorum arbitration or tie-breaker is needed. For example, if there is a communication failure between the nodes in a two-node cluster and each node is attempting to re-form the cluster, Serviceguard must only allow one node to form the new cluster. This is accomplished by configuring a cluster lock or quorum service.

The important concept to note here is that if more than 50% of the nodes in the cluster fail at the same time, the remaining nodes have insufficient quorum to form a new cluster and fail themselves. This is irrespective of whether or not a cluster lock has been configured. It is for this reason that cluster configuration must be carefully analyzed to prevent failure modes that are common among the cluster nodes. One example of this concern is the power circuit considerations that are documented in HP 9000 Enterprise Servers Configuration Guide, Chapter 6 and in the Serviceguard for Linux Order and Configuration Guide (for details contact your HP Sales Representative). Another area where it is possible to have a greater than 50% node failure is in the use of partitioned systems within the cluster. Configuration considerations for preventing this situation are described in the section "Partition Interactions."

Quorum arbitration

Should two equal-sized groups of nodes (exactly 50% of the cluster in each group) become separated from each other, quorum arbitration allows one group to achieve quorum and form the cluster, while the other group is denied quorum and cannot start a cluster. This prevents the possibility of split-brain activity—two sub-clusters running at the same time. If the two sub-clusters are of unequal size, the sub-cluster with greater than 50% of the previous quorum forms the new cluster and the cluster lock is not used.

For obvious reasons, two-node cluster configurations are required to configure some type of quorum arbitration. By definition, failure of a node or loss of communication in a two-node cluster results in a 50% partition. Due to the assumption that nodes fail independently of each other (independent failure assumption), the use of quorum arbitration for cluster configurations with three or more nodes is optional, though highly recommended.

There are several techniques for providing quorum arbitration in Serviceguard clusters:

- On HP-UX 11i v2 and HP-UX 11i v3 through a cluster lock disk which must be accessed during the arbitration process. The cluster lock disk is a disk area located in a volume group that is shared by all nodes in the cluster. Each sub-cluster attempts to acquire the cluster lock. The sub-cluster that gets the cluster lock forms the new cluster and the nodes that were unable to get the lock cease activity. A cluster lock disk can be used in Serviceguard clusters of up to four nodes.
- On Linux, HP-UX 11iv2 and HP-UX 11i v3 through a Lock LUN which must be accessed during the arbitration process. The Lock LUN is a logical Unit, usually a "disk" defined in an Array that is shared by all nodes in the cluster. Each sub-cluster attempts to acquire the Lock LUN. The sub-cluster that gets the Lock LUN forms the new cluster and the nodes that were unable to get the lock cease activity. A Lock LUN can be used in Linux Serviceguard clusters of up to four nodes.
- Through an arbitrator node which provides tie breaking when an entire site fails, as in a disaster scenario. An arbitrator node is a cluster member typically located in a separate data center. Its main function is to increase the Serviceguard cluster size so that an equal partition of nodes is unlikely between production data centers. This can be used in Serviceguard clusters running HP-UX or Linux.

- Through a quorum service, for Serviceguard clusters of any size or type. Quorum services are provided by a quorum server process running on a machine outside of the cluster. The quorum server listens to connection requests from the Serviceguard nodes on a known port. The server maintains a special area in memory for each cluster, and when a node obtains the cluster lock, this area is marked so that other nodes will recognize the lock as "taken." A single quorum server running on either HP-UX or Linux can manage multiple HP-UX and Linux Serviceguard clusters.

Partition interactions

With this background in mind, we next need to examine to what extent the partitioning schemes either meet or violate the independent failure assumption.

The partitioning provided by nPartitions is done at a hardware level, and each partition is isolated from both hardware and software failures of other partitions. This provides isolation between the OS instances running within the partitions. In this sense, nPartitions meets the assumption that the failure of one node (partition) will not affect other nodes. However, within the Superdome infrastructure and other servers supporting nPartitions, there exists a very small possibility of a failure that can affect all partitions within the cabinet. So, to the extent that this infrastructure failure exists, nPartitions violates the independent failure assumption. However, depending on the specific configuration, nPartitions can be used within a Serviceguard cluster.

The vPars form of partitioning is implemented at a software level. While this provides greater flexibility in dividing hardware resources between partitions, it does not provide any isolation of hardware failures between the virtual partitions. The failure of a hardware component being used by one vPar can bring down other vPars within the same hardware platform.

In addition to the failure case interactions, vPars exhibit a behavior that should also be considered when including a vPars as a node in a Serviceguard cluster. Due to the nature of the hardware/firmware sharing between vPars, it is possible for one vPars partition to induce latency in other vPars partitions within the same nPartition or node. For example, during bootup, when the booting partition requests the system firmware to initialize the boot disk, it is possible for other partitions running in the same machine to become blocked until the initialization operation completes. The effect of this type of latency is discussed in the section "Latency Considerations."

Integrity VM is a soft partitioning and virtualization technology that allows you to run multiple virtual machines (or "guests") on a single physical HP Integrity server or nPartition. Like vPars, Integrity VM provides operating system isolation with shared hardware. Multiple virtual machines hosted in a single partition are vulnerable to a single hardware failure. To protect virtual machines from this SPOF, run Integrity VM on multiple cluster nodes and create virtual machine packages that can fail over from one cluster node to another. For the cluster to survive, distribute active virtual machines across VM Hosts so at least half of them remain running after a hardware failure.

Alternatively, Serviceguard can be used to protect the applications running on guests. In this configuration, install Serviceguard on the guest and create application packages that can fail over to another virtual machine or to a physical server (or nPar) that is not running Integrity VM. Each configuration provides different levels of protection:

1. Cluster in a box consists of two or more virtual machines running on the same VM Host system, each of which is an HP-UX guest running Serviceguard. Applications are packaged on the guest and the cluster members are virtual machines. This configuration is useful for testing but provides no protection against SPOF.
2. Cluster across host consists of two or more VM Host systems (HP Integrity servers or nPars). Serviceguard cluster members are guests on two or more VM Host systems. Application packages can fail over to a guest running on a different VM Host system. This provides protection against hardware failure. One example is to run active applications on guests on separate VM Host

systems, while maintaining a third VM Host system for backup. Should one or more active applications fail, they can fail over to the backup VM Host system, which has the capacity to run all the guests at one time if necessary. The backup system might have limited performance but the applications would resume operation after a hardware failure.

3. Virtual-physical cluster consists of a VM Host system running Serviceguard and an HP Integrity server or nPar that is not running Integrity VM, but is running Serviceguard. The application packages on the guests running on the physical server can fail over to the VM Host. This is an efficient backup configuration, as the VM host can provide virtual machines that serve as adoptive/standby systems for many SG clusters.

Cluster configuration considerations

Using the information from the preceding sections, we can now assess any impacts or potential issues that arise from utilizing partitions and virtual machines (nPartitions, vPars, or virtual machines) as part of a Serviceguard cluster. From a Serviceguard perspective, an OS instance running in a partition or virtual machine is not treated any differently than OS instances running on non-partitioned, physical nodes. Thus, partitioning or using virtual machines does not alter the basic Serviceguard configuration rules as described in HP 9000 Enterprise Servers Configuration Guide, Chapter 6 and the Serviceguard for Linux Order and Configuration Guide. Details can be obtained through your local HP Sales Representative.

An example of these existing configuration requirements is the need to have dual communication paths to both storage and networks. The use of partitioning or virtual machines does, however, introduce interesting configuration situations that necessitate additional configuration requirements. These are discussed below.

Quorum arbitration requirements

As previously mentioned, existing Serviceguard configuration rules for non-partitioned, physical systems require the use of a cluster lock only in the case of a two-node cluster. This requirement is in place to protect against failures that result in a 50% quorum with respect to the membership prior to the failure. Clusters with more than two nodes do not have this as a strict requirement because of the independent failure assumption. However, this assumption is no longer valid when dealing with partitions and virtual machines. Cluster configurations that contain OS instances running within a partition or virtual machine must be analyzed to determine the impact on cluster membership based on complete failure of hardware components that support more than one partition or virtual machine.

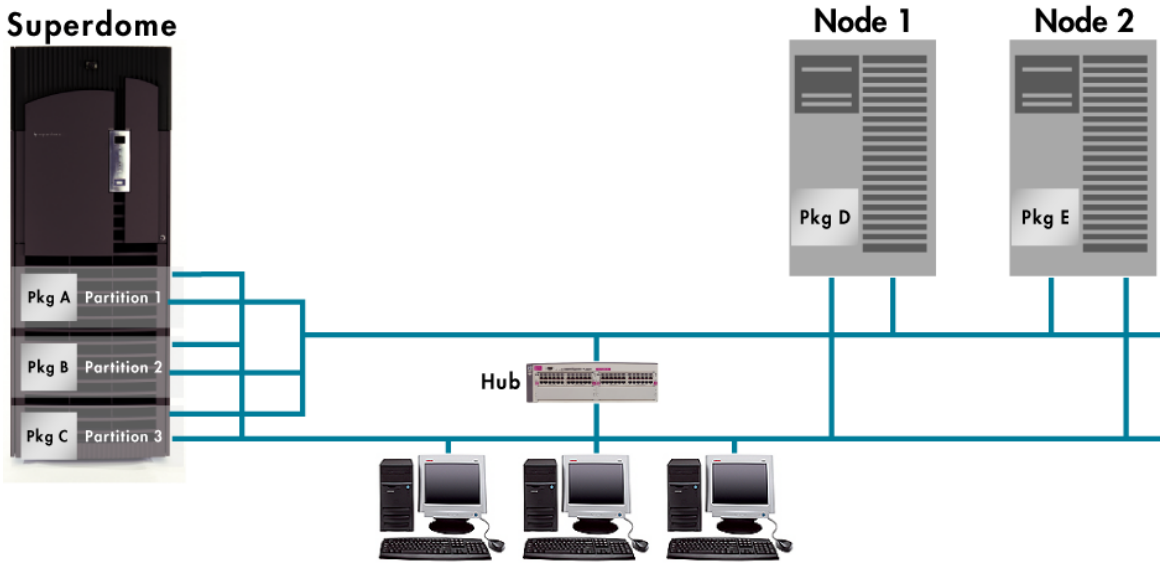
Rule 1. Configurations containing the potential for a loss of more than 50% of the membership resulting from a single failure are not supported. These include configurations with the majority of nodes as partitions or virtual machines within a single hardware cabinet. This implies that when there are two cabinets, the partitions or virtual machines must be symmetrically divided between the cabinets.

For example, given three systems as shown in figure 3, creating a five-node cluster with three nPars (or hard partitions) in one and no partitioning in each of the other systems would not be supported because the failure of the partitioned system would represent the loss of greater than 50% of quorum (3 out of 5 nodes). Alternatively, the cluster would be supported if the systems without nPartitions each contained two vPars or virtual machines, resulting in a seven-node cluster. In this case, the failure of any partitioned system (within a single hardware cabinet) would not represent greater then 50% of quorum.

Exception: All cluster nodes are running within partitions in a single cabinet (such as the so-called cluster in a box configuration). The configuration is supported as long as users understand and accept

the possibility of a complete cluster failure. This configuration is discussed in the section “Cluster in a box.”

Figure 3. Unsupported configuration

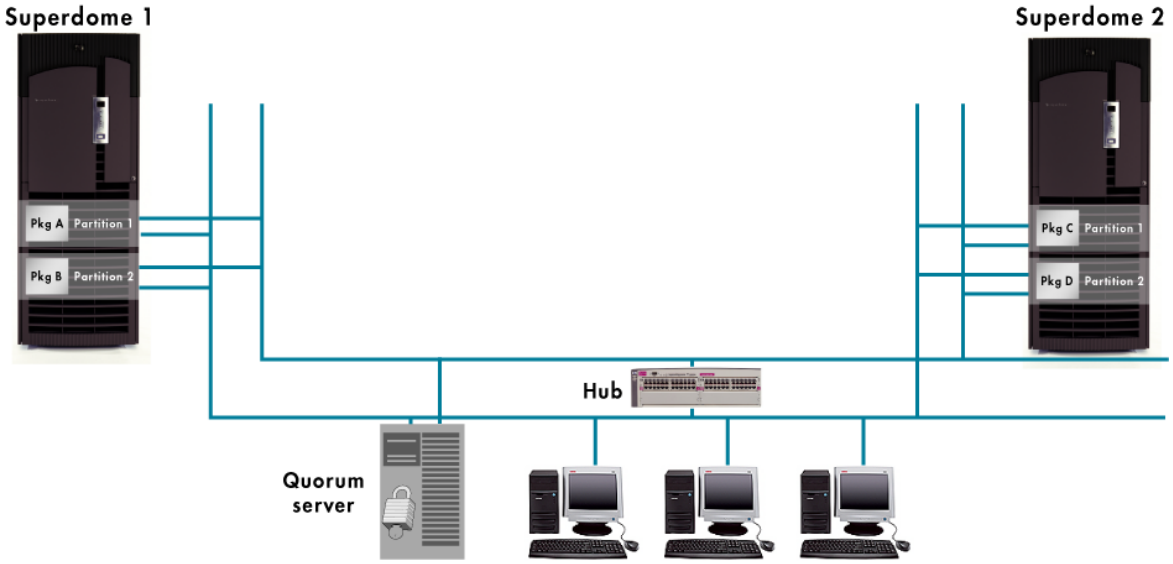


Rule 2. Configurations containing the potential for a loss of exactly 50% of the membership resulting from a single failure require the use of quorum arbitration. This includes:

- Cluster configurations where the nodes are running in partitions or virtual machines that are wholly contained within and equally divided between two hardware cabinets
- Cluster configurations where the nodes are running as vPars partitions or virtual machines that are wholly contained within and equally divided between two nPartitions.
- Cluster configurations where the half the nodes are running in a single hardware cabinet.

For example, to be supported, a four-node cluster consisting of two nPartitions in each of two Superdome cabinets would require a quorum arbitration device. In figure 4, there are two Superdomes, each with two partitions. Each partition is running one package. In this example the quorum arbitration is provided by a quorum server.

Figure 4. A 4—node cluster from two Superdomes with quorum server



Cluster configuration and partitions

Given the configuration requirements described in Rules1 and 2, a few interesting observations can be made of clusters utilizing partitioning:

- If it is determined that a cluster lock is needed for a particular configuration, the cluster must be configured so the cluster lock is isolated from failures affecting the cluster nodes. This means that the lock device must be powered independently of the cluster nodes (including the hardware cabinets containing the partitions that make up the cluster).
- Clusters wholly contained within two hardware cabinets and that utilize the cluster lock disk for quorum arbitration are limited to either two or four nodes. This is due to a combination of the existing Serviceguard rule that limits support of the cluster lock disk to four nodes and Rule 1.
- Cluster configurations can contain a mixture of vPars, nPartitions, virtual machines, and independent nodes as long as quorum requirements are met.
- For a cluster configuration to contain no single points of failure, it must extend beyond a single hardware cabinet, and it must comply with both the quorum rules and the Serviceguard configuration rules described in HP 9000 Enterprise Servers Configuration Guide, Chapter 6 and the Serviceguard for Linux Order and Configuration Guide. In other words, “cluster in a box” is not supported.

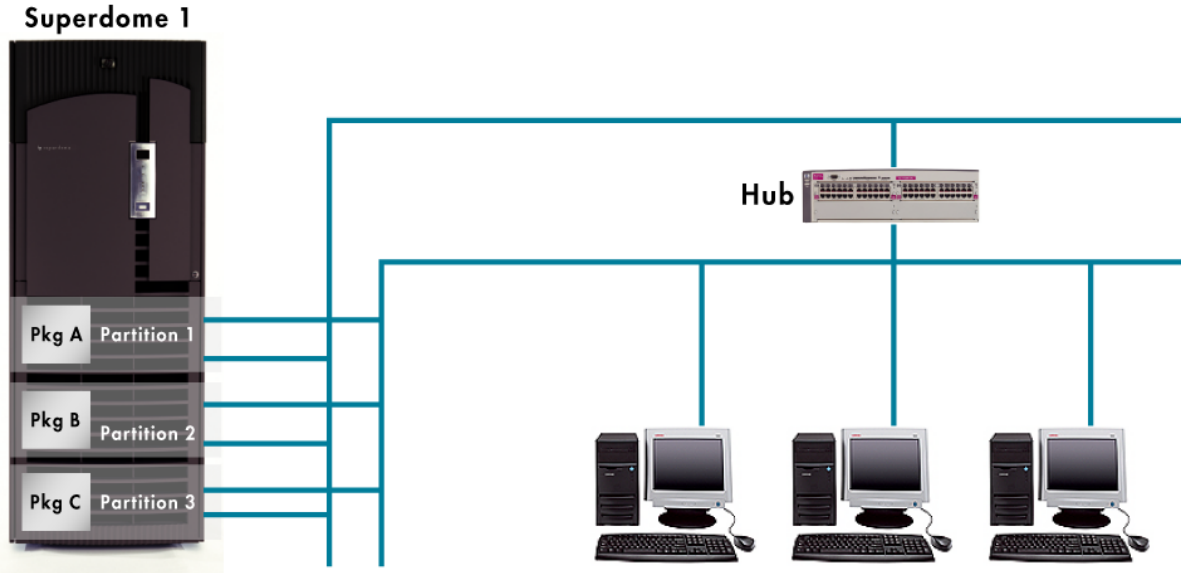
Cluster in a box

One unique possible cluster configuration enabled by partitioning is called cluster in a box. In this case, all the OS instances (nodes) of the cluster are running in partitions within the same hardware cabinet. While this configuration is subject to single points of failure, it may provide adequate availability characteristics for some applications and is thus considered a supported Serviceguard configuration. Users must carefully assess the potential impact of a complete cluster failure on their availability requirements before choosing to deploy this type of cluster configuration.

A cluster in-a-box configuration consisting exclusively of vPars or virtual machines running in a single nPartition or non-partitioned box is more vulnerable to complete cluster failure than is a cluster made

up exclusively of nPartitions since an nPartition is isolated from hardware and software failures of other nPartitions.

Figure 5. Cluster-in-a-box configuration



I/O considerations

Serviceguard does not treat OS instances running in a nPar or vPar any differently than those running on an independent node. Thus, partitions do not provide any exemptions from the normal Serviceguard connectivity rules (such as redundant paths for heartbeat networks, and to storage) nor do they impose any new requirements. With Integrity VM, Serviceguard automatically waits additional time to allow I/O on the failed node to complete before resuming cluster operations.

There are a couple of interesting aspects related to partitioned systems that should be noted:

- While not strictly a "partitioning" issue, the Superdome platform that supports nPartitions contains its interface cards in an I/O chassis, and there can be more than one I/O chassis per partition. To avoid making the I/O chassis a single point of failure, configure redundant I/O paths in separate I/O chassis. Generally speaking, Superdome provides enough I/O capacity that the Serviceguard redundant path requirement should not constrain the use of partitioning within the cluster.
- vPars on the other hand must share essentially one node's worth of I/O slots. In this case, the redundant path requirement can be a limiting factor in determining the number of partitions that can be configured on a single hardware platform.
- Virtual machines can share the same I/O devices (storage and networking). This allows more virtual machines to be created since they aren't limited by having independent I/O devices (as compared to vPars). However, the failure of a single physical I/O interface impacts all virtual machines that share it. For performance, implementing network and storage redundancy at the VM host level rather than at the VM guest level is recommended.

For example, assume we would like to create a cluster-in-a-box configuration using a Fibre Channel–based storage device. The redundant path requirement means that each partition would need two Fibre Channel interface cards for storage. Each partition would also need a minimum of two network

interface cards for the heartbeat LANs. Assuming that combination Fibre Channel/network cards are not used, each partition would require a minimum of four interface cards. To support a 2 partition cluster-in-a-box the system would need to have a total of eight I/O slots.

The use of "combination" cards that combine both network and storage can help in some situations. However, redundant paths for a particular device must be split across separate interface cards (for example, using multiple ports on the same network interface card for the heartbeat LANs is not supported).

For Integrity VM, the following types of storage units can be used as virtual storage by virtual machine packages:

- Files inside logical volumes (LVM, VxVM, Veritas cluster volume manager (CVM))
- Files on Cluster File System (CFS)
- Raw logical volumes (LVM, VxVM, CVM)
- Whole disks

The following storage types are not supported:

- Files outside logical volume
- Disk partitions

When LVM, VxVM, or CVM disk groups are used for guest storage, each guest must have its own set of LVM volume groups or VxVM (or CVM) disk groups. For CFS, all storage units are available to all running guests at the same time. For configurations where Serviceguard runs on the virtual machine, the storage units must be whole disks.

Latency considerations for vPars

As mentioned previously, there is a latency issue, unique to vPars that must be considered when configuring a Serviceguard cluster to utilize vPars.

There are certain operations performed by one vPars partition (such as initializing the boot disk during bootup) that can induce delays in other vPars partitions within the same nPartition or node. The net result to Serviceguard is the loss of cluster heartbeats if the delay exceeds the configured NODE_TIMEOUT (pre – A.11.19) or MEMBER_TIMEOUT (A.11.19 or later) parameter.

If heartbeats are not received within NODE_TIMEOUT (pre – A 11.19) , the cluster begins the cluster re-formation protocol and, providing the delay is within the failover time, the delayed node simply rejoins the cluster. This results in cluster re-formation messages appearing in the syslog(1m) file along with diagnostic messages from the Serviceguard cluster monitor (cmcld) describing the length of the delay. For this reason, it is recommended that clusters containing nodes running in a vPars partition, be carefully tested using representative workloads to determine the appropriate NODE_TIMEOUT (pre – A 11.19) parameter that eliminates cluster reformations caused by vPars interactions.

If heartbeats are not received within MEMBER_TIMEOUT (A.11.19 or later), the delayed node will be removed from cluster and will restart. Thus appropriate value of MEMBER_TIMEOUT becomes more important in vPars to avoid node failures due to latency. For this reason, it is recommended that clusters containing nodes running in a vPars partition, be carefully tested using representative workloads to determine the appropriate MEMBER_TIMEOUT parameter that eliminates unnecessary failovers caused by vPars interactions.

Note:
This does not eliminate the cmcld diagnostic messages that record delays of greater than certain values.

Latency considerations for Integrity Virtual Machines

With Integrity Virtual Machines there is a risk that a heavy processing load will cause Serviceguard to lose cluster heartbeats and trigger a cluster-reformation or node failure. If you encounter this situation, adjust the Serviceguard NODE_TIMEOUT or MEMBER_TIMEOUT upwards. This can result in longer failover times.

Other Linux differences

There are some restrictions listed in this document that are considered strong recommendations for Linux configurations. If these restrictions are violated then some failures will cause the failure of a node when only an interface or network card has failed. For example, Serviceguard for Linux will allow the use of just one dual channel Fibre Channel interface cards for storage connectivity as long as the customer is willing to accept that the failure of this card will cause the entire server to fail.

Serviceguard for Linux does not require that redundant I/O paths be configured in separate I/O chassis although it is strongly recommended. If redundant paths are configured in a single I/O chassis, then failure of that chassis will result in the failure of the server.

Summary and conclusion

With careful consideration of hardware redundancy, elimination of single points of failure, use of arbitration (as needed), and appropriate I/O and networking configuration, using Serviceguard with partitioning or virtual machine technologies can provide you with great protection against unavailable software and hardware.

For more information

To learn more about HP Serviceguard Solutions for HP-UX 11i, please visit:
www.hp.com/go/serviceguardsolutions

To learn more about HP Serviceguard Solutions for Linux, please visit:
www.hp.com/go/sglx

For technical documentation:
www.docs.hp.com/hpux/ha

April 2009