Configuring & Implementing a High Availability Solution on Distributed UNIX Systems

by
Raymond W. Riedel
Hewlett-Packard Company
250 N. Patrick Blvd. Suite 100
Brookfield, Wisconsin 53045

Paper #1018

This paper will discuss the components of a highly available distributed HP-UX system. This discussion will include information regarding disk mirroring and RAID disks, fail over of LANs, using multiple nodes for splitting workloads, UPS/power requirements, security, backup and recovery, policy and procedure changes, and some staffing requirements. There will be a discussion of some of the issues that arise during the design phase of

the highly available solution. During this phase, considerations must be given to the network type(i.e. FDDI, Ethernet), cluster size, and other hardware and software components. Because performance consideration is also a very important part of the design phase, there will be information pertaining to disk channel loading, types of disk interfaces, and CPU sizing. The final phase, testing and implementation, will also be discussed to help avoid problems prior to final production turnover.

Some customers are looking for alternatives to their mainframe by moving to distributed client server environments running mission critical applications, while other customers want to upgrade their existing client server applications to be mission critical. It is important to understand some of the more common high availability configurations in order to match them with your own specific requirements. In order to more fully understand these configurations, I would like to present some of my own simple definitions relating to highly available systems.

Availability -

Availability is a characteristic of service. This indicates the percentage of time that users have access to the system, at an agreed upon level of performance and

usability(i.e Service Level Agreement). This measure, however, is subject to an individual's interpretation.

High Availability -

High availability is the enhancement of a system, processes and environment to reduce services that become unavailable due to both planned and unplanned downtime. This may also equate to the delivery of an acceptable or agreed upon level of service to users during pre-prescribed hours.

Mean Time Between Failures(MTBF) -

Mean Time Between Failure is stated as the time between failures of component or a system. It is often used as a statement of the expected future performance of a component or system, based upon the past performance of like systems or components.

The term availability is often interpreted and viewed differently by many individuals. The systems administrator may have a totally different view of availability than the users do. For example, if 100 users are on the system, and a terminal controller fails, 10 users may be unable to perform their job. However, the other 90 users are still working. The 10 users would see 0% availability while the other 90 see 100% availability.

The system administrator may see the system 90% available, while from a system standpoint, the availability would be 100% because the system is still running. So interpretation is very important.

Planning and designing the High Availability solution is probably the most important task to be completed due to the different application and configuration

combinations that have to be reviewed in order to properly choose the correct solution hat satisfies your particular needs. During this phase, there are many discussions that take place between all the various application groups and outside application providers who will be ultimately affected by whatever solution is decided upon. Some of the issues and questions that surfaces during the planning phase include the following:

Will my current application interface properly with my chosen solution?

Will my current application revision have to change in order to support my solution?

Will my current revision of the operating system support my solution?

How much unplanned downtime can my business and or users tolerate?

How will my operation support and manage the High Availability solution?

The more redundancy built into the solution the more it will cost. However, unplanned downtime usually means significant financial loss.

How many fully active systems will there be? And what level of utilization will they be functioning at?

How will the performance of the cluster be managed?

These were just a few of the many important questions that have to be answered before a solution can be successfully configured. I have spent anywhere between 3 -7 days on-site with customers, there application staff, and outside application developers to determine whether or not a viable Highly Available solution could be designed. There are few application exceptions that would prevent a customer from creating an excellent overall hardware and software high availability solution. Most of

the major application developers have very good relationships with solution providers such as Hewlett-Packard to assist in testing various hardware and software solutions with a particular application to ensure compatibility. The possibility also exists where the individuals who will be part of the planning team will have to become more familiar with the hardware, software or both in order to assist in the design. it is very important that all parties involved fully understand how all the pieces of the solution fit together and how they function. Depending on the your particular operation, the planning and design should include the following staff :

IT Director

Operations Manager

Operators who support the hardware

Technical support staff

Department project managers

Networking staff

Development staff

Any outside application providers

Consultants providing implementation assistance

Cooperation, commitment, and involvement from everyone can yield a very smooth implementation of a well planned High Availability solution.

The basic components of any Highly Available solution should be very robust and reliable. Simply using the standard high reliability of the systems will not alone meet the availability needs of many of the mission critical applications. For example, even though disk drives are very reliable, they are mechanical. They go through a period of higher failures in the early stages and proceed to stabilize to a lower annual

failure rate until they reach the end of the product life cycle, when the failure rate rises again. These single points of failure or *SPOFs* are the focal points of the design and planning stage and of the configuration stage. The ultimate Highly Available solution would have little or no *SPOFs*.

This as I stated earlier, will depend on your critical business factors and how they relate to your IT operation and your cost tolerance.

Disk Mirroring as part of the overall High Availability solution would be a critical component. This software solution prevents data loss due to disk failures by "mirroring" a copy of the data from one disk to another separate disk. There is also the possibility of utilizing 3-way mirroring which would enable you to backup a copy of the data while still maintaining "mirroring" on another set of disks. Mirroring generally provides faster performance than RAID. Read performance is increased due to the fact that all read requests are "queued" to the least busy drive in the mirror set. There is also more "spindles" to spread the I/O load which decreases the possibility of having disk bottlenecks. However, the possibility of having unplanned downtime may increase in the number of disk drives. RAID devices, on the other hand are typically self sufficient each having redundant components.

As part of the cluster configuration in Hewlett-Packard's MC/ServiceGuard High Availability software solution, all disk devices which are owned by a particular application must have a communication path to any other node in the cluster which could run that application. This presents the problem of disk channel loading. Since these devices are considered to be on a *shared bus*,

the maximum number of devices on that shared bus must be carefully planned according to anticipated peak activity.

As we move into higher speed communication between these devices(i.e.fiber channel), the number of devices configured in the cluster as well as the distance between the nodes in the cluster will significantly increase. Fiber channel will allow nodes and disk devices to possibly be in different locations, which would allow node fail over to systems in completely remote locations rather than in the same building.

In Hewlett-Packard's High Availability solution, communication between the nodes in the cluster to insure the cluster's health is handled through the networking interfaces. These interfaces are typically Ethernet or FDDI. To eliminate the possibility of a single-point-of-failure in the network, the recommended configuration would include multiple Ethernet cards which would act as standby interfaces and also multiple FDDI cards. There is an active connection and an idle one which picks up if the other goes down. For instance, with a FDDI interface a concentrator would be used to increase the availability of the cluster to provide fail over capabilities to a standby FDDI interface. Having multiple cards would also allow us to configure a dedicated communication path for the cluster management software to help eliminate unwanted fail overs.

One of the final steps in the configuration of the cluster, is CPU sizing.

We should already have a good idea what the performance requirements are for the applications we will be running on each node in the cluster. However, there must also be some form of capacity planning done for the cluster in order to determine if the

application on node #1 can be successfully run on node #2 in addition to the current load already on node #2. This would have to be determined for all the nodes in the cluster which would be taking over applications from any other failed node. Completing these studies will determine the recommended size of the cluster and on which nodes the applications will be run.

The final phase would be the implementation, testing, and documentation of the highly available cluster. This should include the following tasks:

Define a set of failure scenarios.

Create, verify and execute a test plan which would fully validate the defined set of scenarios.

Demonstrate full functionality of the cluster

To fully test the functionality of the cluster and the test scenarios will more than likely require some significant off-hours time where the system(s) could be brought down without causing the interruption of users.

Completion of complete documentation of what the configuration looks like and how the cluster is structured from an application standpoint.

For optimal availability, enterprise clusters and the processing environment they support should be designed from top to bottom for high availability. The cluster should be constructed in a way that eliminates any single point of failure. System management tools should be deployed to manage the cluster(s), and environmental factors such as power failures should be addressed.

Configuring & Implementing a High Availability Solution on Distrib. Unix

1018 - 9