Providing Highly Available NFS Services with HACMP

By Thomas Casey and Robert Metcalf

IBM's HACMP software is a platform for highly available services. This article will help current HANFS users determine if HACMP satisfies their requirements. It describes how each product provides highly available NFS services, including the advantages and disadvantages of each approach. It also provides a general "how to" for using HACMP to build a highly available NFS server, including special considerations and caveats.

Thomas Casey

Robert Metcalf

The Network File System (NFS) has gained universal acceptance as a general-purpose "shared" filesystem for client/server computing. AIX users have two options for making NFS services highly available: High Availability for Network File System (HANFS) and High Availability Cluster Multiprocessing (HACMP).

Each product has distinct strengths. HANFS provides a reliable NFS server capability by enabling a backup processor to recover current NFS activity if the primary NFS server fails. HACMP provides a general infrastructure on which to build highly available environments for mission-critical data and applications.

The main advantage of HANFS is that it uses AIX extensions to the standard NFS functionality, enabling it to handle duplicate requests correctly and restore lock state during NFS server fallover and reintegration. The main advantage of HACMP is its flexibility and scope. HACMP supports more fallover configurations, processors, disks, and networks than HANFS.

Currently, HANFS has an uncertain future. Although it continues to be included in AIX 3.2.5, IBM has not made it available on AIX 4.1. This gives users the choice of staying with the current version of HANFS and accepting the current AIX, hardware, and configuration support, or finding an alternative way of providing highly available network filesystem services.

HANFS

HANFS, an option of AIX 3.2, is designed to make NFS services highly available. HANFS supports the NFS protocol, can be used by NFS clients without modification, and provides reliable filesystem services by recovering from disk and server failures.

Recovering from Disk Failures

HANFS handles disk failure by using the Logical Volume Manager (LVM) mirroring facility to mirror data across different physical volumes. All copies of the data are on disks controlled by the same file server, which eliminates the overhead of ensuring consistency and coherence between the two servers. If a disk fails, the LVM redirects disk requests to a mirrored copy, making the disk failure transparent to users.

Recovering from Server Failures

HANFS handles server failures by connecting dual-ported disks to a pair of RISC System/6000 processors. One processor in the pair is designated as the server, the other as the backup. The server maintains enough information on the shared disks so that the current state at the time of failure can be reconstructed.

The two processors periodically exchange heartbeat messages to monitor the state of the other processor in the server pair. If the server

fails, the backup processor takes over the shared volume groups and uses the information stored there to reconstruct the lost state. The backup then impersonates the failed server and normal operations continue. NFS clients on the network are oblivious to the failure and access the filesystem at the same address.

HANFS Functional Overview

HANFS supports two processors arranged in one of two possible configurations:

- One processor is the server and the other processor is strictly a backup.
- Each processor is both a server and a backup.

When a client mounts a directory from the HANFS server, the client's name is registered in a special file on both the local and remote processors. If the server fails, the backup detects the failure, takes over for the server, and reestablishes NFS services. Using its client list, the backup contacts all clients, which then re-request any outstanding locks they held on the server. Later, when the primary server is operational, this procedure is reversed and the server once again provides the NFS services.

The HANFS software consists of three daemons:

- **HANFS** (`hanfsd`). The `hanfsd` daemon coordinates all actions between the server pair. It starts the `hacfgd` and `hapngd` daemons, and tells the `hacfgd` which operations to perform and when to perform them.
- **HANFS configuration** (`hacfgd`). The `hacfgd` daemon has three major functions: configuring the server pair, coordinating the fallover to the backup when the server fails, and reintegrating the server when it returns.
- **HANFS ping** (`hapngd`). The `hapngd` daemon issues ICMP ECHO requests to determine when its partner processor joins or fails, and reports changes in status to the `hanfsd` daemon.

Configuring the Server Pair

The `hacfgd` daemon performs the following tasks to configure the server pair:

1. Mounts the filesystems
2. Stops the NFS `nfsd`, `rpc.mountd`, and `rpc.lockd` daemons
3. Exports the filesystems listed in the `/etc/exports.hanfs` file
4. Sets up the primary adapter
5. Starts the NFS `rpc.mountd`, `rpc.lockd`, and `nfsd` daemons
6. Refreshes the `rpc.statd` daemon on the backup processor

JFS Logging and the NFS Duplicate Cache

A duplicate cache allows NFS to reject duplicate requests that, if executed, might have harmful side effects. For example, a duplicate delete of a file would fail but a duplicate read would succeed. The `nfs_dupget` and `nfs_dupsave` commands are AIX extensions to NFS that copy the duplicate cache into user memory and restore duplicate cache entries from user memory. For each entry in the NFS duplicate cache, the NFS daemon creates a Journaled File System (JFS) log entry so that the NFS duplicate cache can be rebuilt during the log redo portion of the filesystem consistency check. Since the cache is non-volatile, the backup processor can restore the cache after an NFS server has crashed.

The main advantage of HACMP is its flexibility and scope.

rpc.statd Extensions

AIX provides extensions to the NFS `rpc.statd` daemon that instruct it to inform the backup node when new clients request NFS services. This remote node then creates the `/etc/sm/<hostname>` files that shadow the server's `/etc/sm/<hostname>` files so that client lock requests can be tracked and reclaimed after fallover.

Fallover

After the server fails, the `hacfgd` daemon performs the following steps to acquire the primary server's NFS mount points:

1. Mounts the filesystems (includes `fsck` and log redo)
2. Refreshes `rpc.statd` on the remote processor
3. Stops the NFS `nfsd`, `rpc.mountd`, and `rpc.lockd` daemons
4. Exports the filesystems listed in the `/etc/exports.hanfs` file
5. Takes over the server's primary adapter address
6. Starts the NFS `nfsd`, `rpc.mountd`, and `rpc.lockd` daemons

Restoring NFS Duplicate Cache

The `fsck` and log redo performed on the filesystem rebuild the NFS duplicate cache on the backup node so that duplicate requests are handled correctly.

Restoring Lock State

Since the backup has a record of all clients that held locks on the server (the `rpc.statd` extensions), it informs the clients to resubmit their lock requests so that the lock state can be rebuilt. The NFS `rpc.lockd` daemon performs this operation during its initialization phase.

Reintegration

When the server is restarted after a failure, the backup performs the following operations:

1. Stops the NFS `nfsd`, `rpc.mountd`, and `rpc.lockd` daemons
2. Releases the filesystems
3. Releases the primary adapter of the server
4. Starts the NFS `nfsd` daemon
5. Refreshes the NFS `rpc.statd` daemon on the server
6. Starts the NFS `rpc.mountd` and `rpc.lockd` daemons
7. Copies the NFS duplicate cache to the server

When this completes, the primary server then performs the configuration steps described in the configuration section above.

Advantages and Disadvantages of HANFS

HANFS takes advantage of AIX extensions to the standard NFS functionality so that it can handle duplicate requests correctly and restore lock state during NFS server fallover and reintegration. Although NFS supposedly does not maintain any state information, most real-world implementations maintain a small amount of state information in a duplicate cache.

HANFS uses the AIX-supplied extensions to record duplicate request and outstanding lock information on the shared disk so that state information is not dependent on a single processor. If a failure occurs, the backup can read this information from the shared disk and reconstruct the duplicate cache and restore the current locks. In this way, failure and recovery are completely

Building a Highly Available NFS Server with HACMP

Building a highly available NFS server with HACMP requires almost as much time designing the cluster in front of a whiteboard as it does implementing the cluster in front of a console. As you plan the cluster, determine whether you need a server-to-server or server-to-client configuration and devise a strategy for fallover and reintegration. Here are some issues to consider and some pitfalls to avoid.

- If you choose a server-to-server implementation, make sure that your takeover node has the capacity to perform its original duties and the additional work of the failed node. Also, on a server-to-server cluster, devise a plan to deal with the increased fallover time.
- During fallover, the takeover node resets the disks or controllers of the shared disks and varies on the volume group. The filesystem is then mounted locally and NFS-exported to clients. In a server-to-server configuration, the reset is preceded by the `clnfskill` command followed by a umount on the NFS-mounted filesystem from the failed host. You can customize this default behavior through script changes (not recommended) and event management (recommended).
- You can use pre- and post-event processing to reduce fallover time in server-to-server configurations. Use pre-events to batch the umounts rather than run them serially, which significantly reduces umount time. Use post-events to handle all NFS processing on the takeover node and to granularize NFS handling beyond default HACMP behavior.
- For fallover to appear seamless to the clients, major numbers must be the same for NFS volume groups on all server nodes.
- By default, HACMP runs the `exportfs -i` command during fallover, ignoring the `/etc/exports` file. You may need to modify this processing at your site.
- Be sure you consider and plan for reintegration of a failed node. If you choose to have a node reintegrate, your clients will see an interruption in service while the disk, volume group, and filesystem are dropped by the takeover node and reacquired by the original node.
- HACMP provides the `clnfskill` utility to kill processes with files open in an NFS-mounted filesystem. Read the man page for this important command to be sure you understand how and when to use it.

transparent to applications running on the file server's clients.

HANFS also has some disadvantages:

- HANFS is not included in AIX 4.1.
- An HANFS configuration is limited to two processors. If both processors fail, NFS services cannot be restored to clients.
- HANFS is limited to SCSI support only. Other disk technologies, such as IBM 9333 serial link disks, are not supported.
- Since HANFS does not support AIX 4.1, you cannot use symmetric multiprocessors with HANFS.
- HANFS is limited to NFS functionality only.
- HANFS does not support server-to-server (cross-mount) NFS functionality; it supports server-client only.

HACMP

HACMP provides high availability and cooperative computing services for clusters of two to eight RISC System/6000s. HACMP software combined with industry-standard hardware reduces downtime by quickly restoring services when a critical system, component, or application fails. A cooperative computing system enables distributed applications to take advantage of the increased computing power available in the cluster.

The key facet of a highly available system is its ability to detect and respond to changes that could impair essential services. HACMP enables a cluster to continue to provide critical data and applications services even though a key system component, such as a network adapter, is no longer available. When a component becomes unavailable, HACMP detects the loss and shifts that component's workload to another component in the cluster.

Although the switchover is not instantaneous, services are restored rapidly, usually within one to five minutes. Since high availability is not fault tolerant, many sites are willing to absorb a small amount of downtime rather than pay the much higher cost for fault tolerance.

A highly available cluster is built by eliminating single points of failure in the cluster. A single point of failure exists when a critical function is provided by only one component. If that component fails, the cluster has no other way to provide that function and its services become unavailable. For example, if all the data for an application resides on a single non-mirrored disk and that disk fails, the disk is a single point of failure for the entire cluster. Clients cannot access that application until access to the data on the disk is restored.

HACMP provides recovery options for the following system components:

- Processors
- Networks and network adapters
- Disk and disk adapters
- Applications

The HACMP Cluster Manager

The cluster manager agent is the central control mechanism for providing highly available services. It runs as a daemon process on each processor, or node, in an HACMP cluster. The cluster manager monitors local hardware and software subsystems, tracks the availability of other nodes in the cluster, and coordinates the takeover and release of cluster resources in response to changes in the cluster topology, known as cluster events. A cluster event can be triggered by a change affecting a network adapter, network, or node.

HACMP can deal with various types of failures including network adapter, network, node (including application failure), and disk and disk adapter failure (using LVM mirroring).

HACMP provides high availability and cooperative computing services for clusters of two to eight RISC System/6000s.

Recovering from Network Adapter Failures

A service adapter is the primary connection between a node and a network. The cluster manager monitors network adapters by sending keep-alive packets every half second over the service adapters. The cluster manager uses the presence or absence of keep-alive activity over a service adapter as a sign of the adapter's health. If there is no keep-alive activity over an adapter for five seconds, the cluster manager instructs a standby adapter to take over the IP address of the service adapter.

Once detected, swapping adapters takes about three seconds to complete. Client applications do not lose their connection. A slight delay may be noticeable, but often there is no discernible effect beyond normal workload variations. Therefore, a network adapter failure is detected and fully recovered in less than 10 seconds.

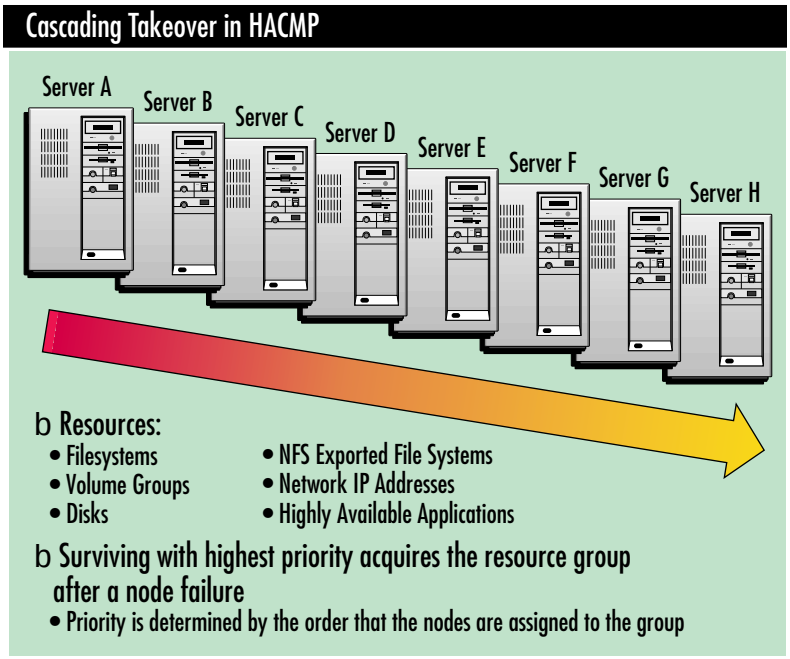

Figure 1. Cascading takeover available in HACMP

Recovering from Network Failures

HACMP uses multiple communications links to eliminate a network as a single point of failure. The cluster manager detects when a network fails and when it later returns to service. Detecting a local network event takes three seconds; detecting a remote network event takes six seconds. The cluster manager sends a mail message to the system console when a network event occurs, but takes no further action since the appropriate response depends on the specific network configuration. Using HACMP's event customization facility, you can tailor the fallover processing to your site.

Recovering from Node Failures

An HACMP node owns a set of resources: disks, volume groups, filesystems, networks, network addresses, and applications. When a node fails or leaves the cluster, some or all of its resources are distributed among the surviving nodes.

The cluster manager monitors the availability of the other nodes in the cluster by exchanging heartbeats with its neighboring nodes. If a node stops sending heartbeats, the cluster managers on the surviving nodes in the cluster take the necessary actions to get the critical applications up and running and to ensure that data has not been corrupted or lost. The available cluster managers take over the network interface, the volume groups, or disk drives, and then restart the applications.

As part of the takeover, the cluster managers delete and re-create their routes, and refresh the Address Resolution Protocol (ARP) cache of any clients. This allows clients to connect to the backup node using the same address originally used to connect to the primary node.

Any client transaction in mid-flight during a node failure must be resubmitted. This does not cause data corruption since the in-flight transaction has not yet been committed.

Clients experience a brief interruption in service during node failure—typically from one to five minutes—and then continue processing. The takeover time varies depending on the amount of resources that need to be shifted to the takeover node and the amount of application recovery processing required.

Recovering from Disk and Disk Adapter Failures

Like HANFS, HACMP does not directly recover disk and disk adapter failures. These failures are handled within AIX by mirroring across different physical volumes on disks and enclosures, and by internal data redundancy and redundant adapters on disk arrays. In addition, HACMP provides a System Management Interface Tool (SMIT) interface to the AIX Error Notification facility. This allows a system administrator to detect a failure not specifically monitored by HACMP—such as a disk adapter failure—and to program a response to the failure.

Advantages and Disadvantages of HACMP

Using HACMP to provide highly available NFS services has many advantages:

- HACMP is a mature product with a large installed base (over 3,000 licenses). The product's functionality and usability have improved significantly since its initial release in 1992. Future releases will continue to address needs of the RISC System/6000 community.
- HACMP provides extensive processor, disk subsystem, and network support. HACMP supports RISC System/6000 uniprocessor and Symmetric Multiprocessor (SMP) system units. HACMP also supports the POWERparallel SP2 machine. HACMP supports SCSI, SCSI-2, 9333 serial link disk subsystems, as well as 3514, 7135, and 7137 disk arrays. HACMP supports Ethernet, Token Ring, Serial Line Internet Protocol (SLIP), Serial Optical Channel Converter

(SOCC), Fiber-optic Distributed Data Interchange (FDDI), and Fiber Channel Standard (FCS) networks. New processors, disk devices, and communication devices are regularly tested and approved for HACMP.

- An HACMP cluster providing highly available NFS services is not limited to NFS. Other highly available services (such as local disk, concurrent access, and communication) can be included in an HACMP cluster designed primarily for NFS services.
- HACMP supports numerous cluster configurations: traditional *hot standby* configurations where one or more standby machines back up one or more servers; *mutual takeover* configurations, which partition the workload across the cluster; and *concurrent access* configurations where multiple machines simultaneously access a shared database.
- HACMP offers cascading takeover. A single NFS server in a cluster can be backed up by all the other servers in the cluster, for a total of eight nodes. If the NFS-serving node fails, its role is taken over by another node. If that node fails, the NFS-server role cascades to another node, and so on, as shown in Figure 1.
- HACMP's concurrent access feature allows two to eight processors to simultaneously access a database residing on a shared external disk. Using concurrent access, a cluster can provide near-continuous availability that rivals fault tolerance, but at a much lower cost. Additionally, concurrent access provides higher performance, eases application development, and allows horizontal growth.
- The HACMP lock manager is a cooperative computing service that allows distributed applications to serialize access to concurrent resources with locks. The lock manager interface consists of a daemon that maintains a common cluster-wide database of resources against which locks can be taken, and an Application Programming Interface (API) that clients use to request locks.
- The cluster information facilities provided with HACMP enable a developer to monitor the cluster state and to write cluster-aware applications. These facilities include the `clsmuxpd` and `clinfo` daemons. The `clsmuxpd` daemon maintains a network Management Information Base (MIB) accessible to clients via the standard Simple Network Management Protocol (SNMP) interface for network management applications.

  Cluster-aware applications that do not want to use the complex SNMP interface can obtain cluster information using the simpler `clinfo` interface. The `clinfo` interface consists of a daemon that updates cluster topology information in a shared memory with information retrieved from the `clsmuxpd` daemon, and an API that the client applications can use to retrieve this information.

Server-to-Server NFS Cross-mounting

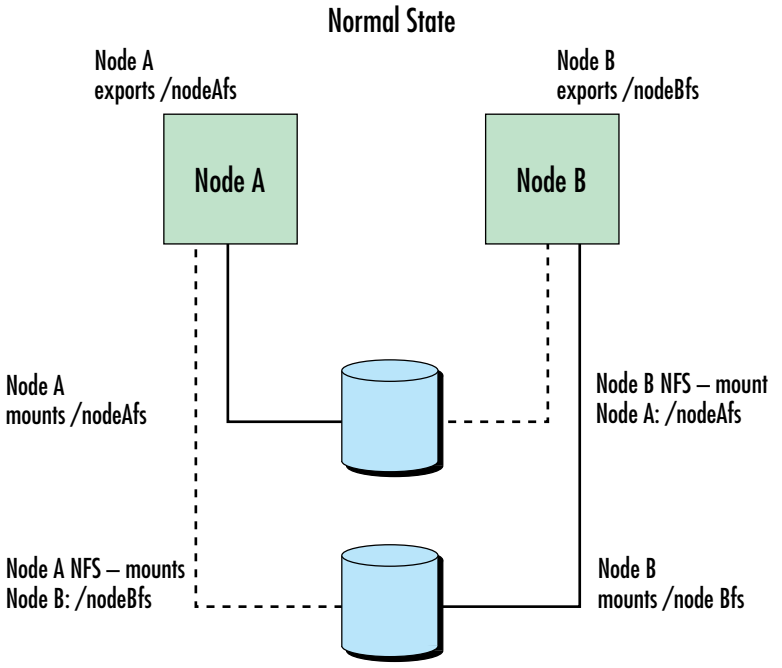

Figure 2. Normal state of server-to-server NFS cross-mounting

- HACMP supports server-to-server (cross-mount) functionality, as well as server-to-client. In other words, Node A can mount `/nodeAfs` locally, and export it to Node B. Node A can then NFS-mount a filesystem (`/nodeBfs`) offered by Node B. Node B can mount `/nodeBfs` locally, and export it to Node A. Node B can then NFS-mount the filesystem (`/nodeAfs`) offered by Node A, as shown in Figure 2. If either node fails, the other can then locally mount and re-export the failed filesystem to other NFS clients, as noted in Figure 3.

Server-to-Server NFS Cross-mounting

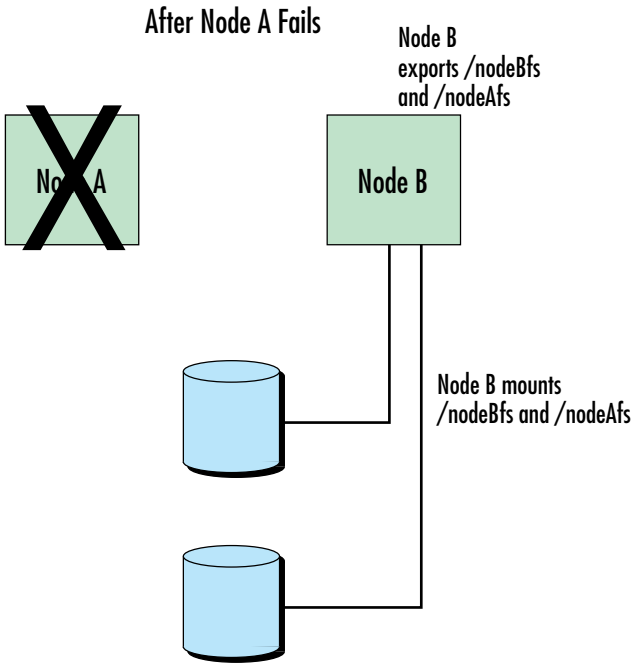

Figure 3. After a node fails in server-to-server NFS cross-mounting

There are also some drawbacks to using HACMP for highly available NFS services:

- Unlike HANFS, HACMP does not take advantage of the NFS extensions provided by AIX. If a failure occurs, applications using file locking need to ensure that locking information gets to the takeover node.

  In a server-to-server configuration where the node doing the NFS mount is using file locking, HACMP may need to remove the `/etc/sm.bak/$host` files before attempting fallover. Use the HACMP `cl_deactivate_nfs` utility to remove these files, followed by a `stopsrc/startsrc` of the `rpc.statd` and `rpc.lock` daemons.

- In server-to-server clusters where filesystems are cross-mounted for one server to another, fallover time is increased. Using the default HACMP scripts, the `clnfskill` utility is run on the NFS-mounted filesystem, followed by a umount. The umount takes 45 to 60 seconds per filesystem as it waits for the RPC timeout from the failed node. If the default HACMP NFS processing is used and there are many cross-mounted filesystems, the fallover time increases significantly.

Thomas Casey, CLAM Associates, Inc., 101 Main Street, Cambridge, MA 02142. Internet: tom@clam.com. Mr. Casey is the manager of CLAM's technical writing group. He has a BS from Trinity College in Hartford, Connecticut and an MS from Emerson College in Boston.

Robert Metcalf, CLAM Associates, Inc., 101 Main Street, Cambridge, MA 02142. Internet: bobmet@clam.com. Mr. Metcalf is a senior member of CLAM's technical support staff. He has a BS from Suffolk University and an MS from Simmons College, both in Boston.