High Availability Cluster Multi-Processing for AIX

Concepts and Facilities Guide

Version 5.4.1

SC23-4864-10

Tenth Edition (October 2007)

Before using the information in this book, read the general information in Notices for HACMP Concepts and Facilities Guide.

This edition applies to HACMP for AIX, version 5.4.1 and to all subsequent releases of this product until otherwise indicated in new editions.

Contents

Chapter 5: Ensuring Application Availability 67

Chapter 6: HACMP Cluster Configurations 93

About This Guide

This guide introduces the High Availability Cluster Multi-Processing for AIX (HACMP™) software.

The following table provides version and manual part numbers for the *Concepts and Facilities Guide*.

HACMP Version	Book Name	Book Number
5.4.1	*Concepts and Facilities Guide*	SC23-4864-10
5.4	*Concepts and Facilities Guide*	SC23-4864-09
5.3: Last update 8/2005	*Concepts and Facilities Guide*	SC23-4864-07
5.3	*Concepts and Facilities Guide*	SC23-4864-06
5.2: Last update 10/2005	*Concepts and Facilities Guide*	SC23-4864-05
5.1: Last update 6/2004	*Concepts and Facilities Guide*	SC23-4864-02

Who Should Use This Guide

System administrators, system engineers, and other information systems professionals who want to learn about features and functionality provided by the HACMP software should read this guide.

Highlighting

This guide uses the following highlighting conventions:

Italic	Identifies new terms or concepts, or indicates emphasis.
Bold	Identifies routines, commands, keywords, files, directories, menu items, and other items whose actual names are predefined by the system.
`Monospace`	Identifies examples of specific data values, examples of text similar to what you might see displayed, examples of program code similar to what you might write as a programmer, messages from the system, or information that you should actually type.

ISO 9000

ISO 9000 registered quality systems were used in the development and manufacturing of this product.

HACMP Publications

The HACMP software comes with the following publications:

- *HACMP for AIX Release Notes* in **/usr/es/sbin/cluster/release_notes** describe issues relevant to HACMP on the AIX platform: latest hardware and software requirements, last-minute information on installation, product usage, and known issues.
- *HACMP on Linux Release Notes* in **/usr/es/sbin/cluster/release_notes.linux/** describe issues relevant to HACMP on the Linux platform: latest hardware and software requirements, last-minute information on installation, product usage, and known issues.
- *HACMP for AIX: Administration Guide,* SC23-4862
- *HACMP for AIX: Concepts and Facilities Guide,* SC23-4864
- *HACMP for AIX: Installation Guide,* SC23-5209
- *HACMP for AIX: Master Glossary,* SC23-4867
- *HACMP for AIX: Planning Guide,* SC23-4861
- *HACMP for AIX: Programming Client Applications*, SC23-4865
- *HACMP for AIX: Troubleshooting Guide,* SC23-5177
- *HACMP on Linux: Installation and Administration Guide,* SC23-5211
- *HACMP for AIX: Smart Assist Developer's Guide,* SC23-5210
- *IBM International Program License Agreement.*

HACMP/XD Publications

The HACMP Extended Distance (HACMP/XD) software solutions for disaster recovery, added to the base HACMP software, enable a cluster to operate over extended distances at two sites. HACMP/XD publications include the following:

- *HACMP/XD for Geographic LVM (GLVM): Planning and Administration Guide,* SA23-1338
- *HACMP/XD for HAGEO Technology: Concepts and Facilities Guide, SC23-1922*
- *HACMP/XD for HAGEO Technology: Planning and Administration Guide,* SC23-1886
- *HACMP/XD for Metro Mirror: Planning and Administration Guide,* SC23-4863.

HACMP Smart Assist Publications

The HACMP Smart Assist software helps you quickly add an instance of certain applications to your HACMP configuration so that HACMP can manage their availability. The HACMP Smart Assist publications include the following:

- *HACMP Smart Assist for DB2 User's Guide, SC23-5179*
- *HACMP Smart Assist for Oracle User's Guide, SC23-5178*
- *HACMP Smart Assist for WebSphere User's Guide, SC23-4877*
- *HACMP for AIX: Smart Assist Developer's Guide, SC23-5210*
- *HACMP Smart Assist Release Notes.*

IBM AIX Publications

The following publications offer more information about IBM technology related to or used by HACMP:

- *RS/6000 SP High Availability Infrastructure*, SG24-4838
- *IBM AIX v.5.3 Security Guide,* SC23-4907
- *IBM Reliable Scalable Cluster Technology for AIX and Linux: Group Services Programming Guide and Reference,* SA22-7888
- *IBM Reliable Scalable Cluster Technology for AIX and Linux: Administration Guide,* SA22-7889
- *IBM Reliable Scalable Cluster Technology for AIX: Technical Reference*, SA22-7890
- *IBM Reliable Scalable Cluster Technology for AIX: Messages*, GA22-7891.

Accessing Publications

Use the following Internet URLs to access online libraries of documentation:

AIX, IBM eServer Series p™, and related products:

http://www.ibm.com/servers/aix/library

AIX v.5.3 publications:

http://www.ibm.com/servers/eserver/pseries/library/

WebSphere Application Server publications:

Search the IBM website to access the WebSphere Application Server Library

DB2 Universal Database Enterprise Server Edition publications:

http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/v8pubs.d2w/en_main#V8PDF

Tivoli Directory Server publications:

http://publib.boulder.ibm.com/tividd/td/IBMDirectoryServer5.1.html

IBM Welcomes Your Comments

You can send any comments via e-mail to hafeedbk@us.ibm.com. Make sure to include the following in your comment or note:

- Title and order number of this book
- Page number or topic related to your comment.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States or other countries:

- AFS
- AIX
- DFS
- @server
- eServer Cluster 1600
- Enterprise Storage Server
- HACMP
- IBM
- NetView
- RS/6000
- Scalable POWERParallel Systems
- Series p
- Series x
- Shark
- SP
- xSeries
- WebSphere
- Red Hat Enterprise Linux (RHEL)
- SUSE Linux Enterprise Server
- RPM Package Manager for Linux and other Linux trademarks.

UNIX is a registered trademark in the United States and other countries and is licensed exclusively through The Open Group.

Linux is a registered trademark in the United States and other countries and is licensed exclusively through the GNU General Public License.

Other company, product, and service names may be trademarks or service marks of others.

Chapter 1: HACMP for AIX

This chapter discusses the concepts of high availability and cluster multi-processing, presents the HACMP cluster diagram, and describes an HACMP cluster from a functional perspective.

High Availability Cluster Multi-Processing for AIX

The IBM HACMP software provides a low-cost commercial computing environment that ensures quick recovery of mission-critical applications from hardware and software failures. HACMP has two major components: high availability (HA) and cluster multi-processing (CMP).

With HACMP software, critical resources remain available. For example, an HACMP cluster could run a database server program that services client applications. The clients send queries to the server program that responds to their requests by accessing a database, stored on a shared external disk.

This high availability system combines custom software with industry-standard hardware to minimize downtime by quickly restoring services when a system, component, or application fails. Although *not* instantaneous, the restoration of service is rapid, usually within 30 to 300 seconds.

In an HACMP cluster, to ensure the availability of these applications, the applications are put under HACMP control. HACMP takes measures to ensure that the applications *remain available* to client processes even if a component in a cluster fails. To ensure availability, in case of a component failure, HACMP moves the application (along with resources that ensure access to the application) to another node in the cluster.

HACMP and HACMP/ES

HACMP 5.4.1 includes all the features of the product that was previously referred to as HACMP/ES (Enhanced Scalability). The product previously referred to as HACMP (Classic) is no longer available.

Note: Prior to version 5.1, the HACMP for AIX software included four features: HAS and CRM with core filesets named `cluster.base*;` and ES and ESCRM with core filesets named `cluster.es*`. Starting with HACMP 5.1, the HAS, CRM and ES features were no longer available, and the ESCRM feature is now called HACMP.

To summarize, HACMP 5.4.1 has the following characteristics:

- Includes all the features of ESCRM 4.5 in addition to the new functionality added in HACMP version 5.1 through 5.4.1.

- Takes advantage of the enhanced Reliable Scalable Cluster Technology (RSCT). RSCT provides facilities for monitoring node membership; network interface and communication interface health; and event notification, synchronization and coordination via reliable messaging.
- Can have up to 32 nodes on HACMP clusters with both non-concurrent and concurrent access.
- Is supported on AIX versions 5.2 and 5.3.

High Availability and Hardware Availability

High availability is sometimes confused with simple hardware availability. Fault tolerant, redundant systems (such as RAID) and dynamic switching technologies (such as DLPAR) provide recovery of certain hardware failures, but do *not* provide the full scope of error detection and recovery required to keep a complex application highly available.

A modern, complex application requires access to all of these components:

- Nodes (CPU, memory)
- Network interfaces (including external devices in the network topology)
- Disk or storage devices.

Recent surveys of the causes of downtime show that actual hardware failures account for only a small percentage of unplanned outages. Other contributing factors include:

- Operator errors
- Environmental problems
- Application and operating system errors.

Reliable and recoverable hardware simply cannot protect against failures of all these different aspects of the configuration. Keeping these varied elements—and therefore the application—highly available requires:

- Thorough and complete planning of the physical and logical procedures for access and operation of the resources on which the application depends. These procedures help to avoid failures in the first place.
- A monitoring and recovery package that automates the detection and recovery from errors.
- A well-controlled process for maintaining the hardware and software aspects of the cluster configuration while keeping the application available.

High Availability vs. Fault Tolerance

Fault tolerance relies on specialized hardware to detect a hardware fault and instantaneously switch to a redundant hardware component—whether the failed component is a processor, memory board, power supply, I/O subsystem, or storage subsystem. Although this cutover is apparently seamless and offers non-stop service, a high premium is paid in both hardware cost and performance because the redundant components do no processing. More importantly, the fault tolerant model does *not* address software failures, by far the most common reason for downtime.

High availability views availability *not* as a series of replicated physical components, but rather as a set of system-wide, shared resources that cooperate to guarantee essential services. High availability combines software with industry-standard hardware to minimize downtime by quickly restoring essential services when a system, component, or application fails. While *not* instantaneous, services are restored rapidly, often in less than a minute.

The difference between fault tolerance and high availability, then, is this: A fault tolerant environment has no service interruption but a significantly higher cost, while a highly available environment has a minimal service interruption. Many sites are willing to absorb a small amount of downtime with high availability rather than pay the much higher cost of providing fault tolerance. Additionally, in most highly available configurations, the backup processors are available for use during normal operation.

High availability systems are an excellent solution for applications that must be restored quickly and can withstand a short interruption should a failure occur. Some industries have applications so time-critical that they cannot withstand even a few seconds of downtime. Many other industries, however, can withstand small periods of time when their database is unavailable. For those industries, HACMP can provide the necessary continuity of service without total redundancy.

Role of HACMP

HACMP helps you with the following:

- The HACMP planning process and documentation include tips and advice on the best practices for installing and maintaining a highly available HACMP cluster.
- Once the cluster is operational, HACMP provides the automated monitoring and recovery for all the resources on which the application depends.
- HACMP provides a full set of tools for maintaining the cluster while keeping the application available to clients.

HACMP lets you:

- Quickly and easily set up a basic two-node HACMP cluster by using the Two-Node Cluster Configuration Assistant.
- Set up an HACMP environment using online planning worksheets to simplify the initial planning and setup.
- Test your HACMP configuration by using the Cluster Test Tool. You can evaluate how a cluster behaves under a set of specified circumstances, such as when a node becomes inaccessible, a network becomes inaccessible, and so forth.
- Ensure high availability of applications by eliminating single points of failure in an HACMP environment.
- Leverage high availability features available in AIX.
- Manage how a cluster handles component failures.
- Secure cluster communications.
- Set up fast disk takeover for volume groups managed by the Logical Volume Manager (LVM).
- Monitor HACMP components and diagnose problems that may occur.

For a general overview of *all* HACMP features, see the IBM website:

http://www.ibm.com/servers/aix/products/ibmsw/high_avail_network/hacmp.html

For a list of *new features* in HACMP 5.4.1, see Chapter 8: HACMP 5.4.1: Summary of Changes.

Cluster Multi-Processing

Cluster multi-processing is a group of loosely coupled machines networked together, sharing disk resources. In a cluster, multiple server machines cooperate to provide a set of services or resources to clients.

Clustering two or more servers to back up critical applications is a cost-effective high availability option. You can use more of your site's computing power while ensuring that critical applications resume operations after a minimal interruption caused by a hardware or software failure.

Cluster multi-processing also provides a gradual, scalable growth path. It is easy to add a processor to the cluster to share the growing workload. You can also upgrade one or more of the processors in the cluster to a more powerful model. If you were using a fault tolerant strategy, you must add *two* processors, one as a redundant backup that does no processing during normal operations.

Availability Costs and Benefits Continuum

The following figure shows the costs and benefits of availability technologies.

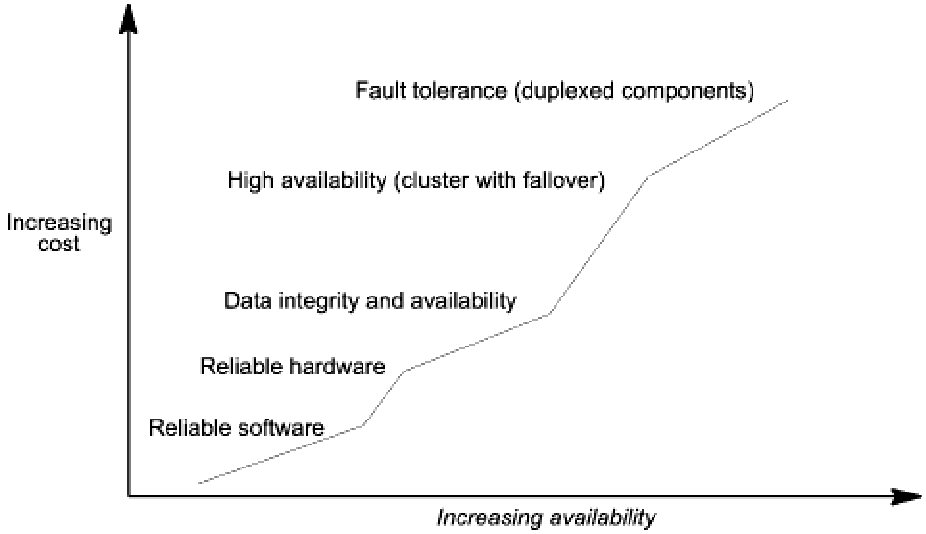

Figure 1. Cost and Benefits of Availability Technologies

As you can see, availability is *not* an all-or-nothing proposition. Think of availability as a continuum. Reliable hardware and software provide the base level of availability. Advanced features such as RAID devices provide an enhanced level of availability. High availability software provides near continuous access to data and applications. Fault tolerant systems ensure the constant availability of the entire system, but at a higher cost.

Enhancing Availability with the AIX Software

HACMP takes advantage of the features in AIX—the high-performance UNIX operating system.

AIX adds new functionality to further improve security and system availability. This includes improved availability of mirrored data and enhancements to Workload Manager that help solve problems of mixed workloads by dynamically providing resource availability to critical applications. Used with the IBM System p™, HACMP can provide both horizontal and vertical scalability without downtime.

The AIX operating system provides numerous features designed to increase system availability by lessening the impact of both planned (data backup, system administration) and unplanned (hardware or software failure) downtime. These features include:

- Journaled File System and Enhanced Journaled File System
- Disk mirroring
- Process control
- Error notification.

Journaled File System and Enhanced Journaled File System

The AIX native file system, the Journaled File System (JFS), uses database journaling techniques to maintain its structural integrity. System or software failures do *not* leave the file system in an unmanageable condition. When rebuilding the file system after a major failure, AIX uses the JFS log to restore the file system to its last consistent state. Journaling thus provides faster recovery than the standard UNIX file system consistency check (**fsck**) utility.

In addition, the Enhanced Journaled File System (JFS2) is available in AIX. For more information, see the section Journaled File System and Enhanced Journaled File System in Chapter 3: HACMP Resources and Resource Groups.

Disk Mirroring

Disk mirroring software provides data integrity and online backup capability. It prevents data loss due to disk failure by maintaining up to three copies of data on separate disks so that data is still accessible after any single disk fails. Disk mirroring is transparent to the application. No application modification is necessary because mirrored and conventional disks appear the same to the application.

Process Control

The AIX System Resource Controller (SRC) monitors and controls key processes. The SRC can detect when a process terminates abnormally, log the termination, pass messages to a notification program, and restart the process on a backup processor.

Error Notification

The AIX Error Notification facility detects errors, such as network and disk adapter failures, and triggers a predefined response to the failure.

HACMP builds on this AIX feature by providing:

- Automatically created error notification methods for volume groups that you configure in HACMP. These error notification methods let HACMP react to certain volume group failures and provide recovery.

  HACMP also configures automatic error notification methods for those volume groups that are configured in AIX (and are *not* configured in HACMP) but that contain the corresponding file systems configured in HACMP. This ensures that the file systems are kept highly available and allows HACMP to automatically respond to the volume group failures by recovering the file systems.

  For more information on error notification and how it is used in HACMP, see Eliminating Disks and Disk Adapters as a Single Point of Failure in Chapter 5: Ensuring Application Availability.
- Error emulation function. It allows you to test the predefined response without causing the error to occur. You also have an option to automatically configure notification methods for a set of device errors in one step.

Physical Components of an HACMP Cluster

HACMP provides a highly available environment by identifying a set of resources essential to uninterrupted processing. It also defines a protocol that nodes use to collaborate to ensure that these resources are available. HACMP extends the clustering model by defining relationships among cooperating processors where one processor provides the service offered by a peer should the peer be unable to do so.

As shown in the following figure, an HACMP cluster is made up of the following physical components:

- Nodes
- Shared external disk devices
- Networks
- Network interfaces
- Clients.

The HACMP software allows you to combine physical components into a wide range of cluster configurations, providing you with flexibility in building a cluster that meets your processing requirements.

This figure shows one example of an HACMP cluster. Other HACMP clusters could look very different—depending on the number of processors, the choice of networking and disk technologies, and so on.

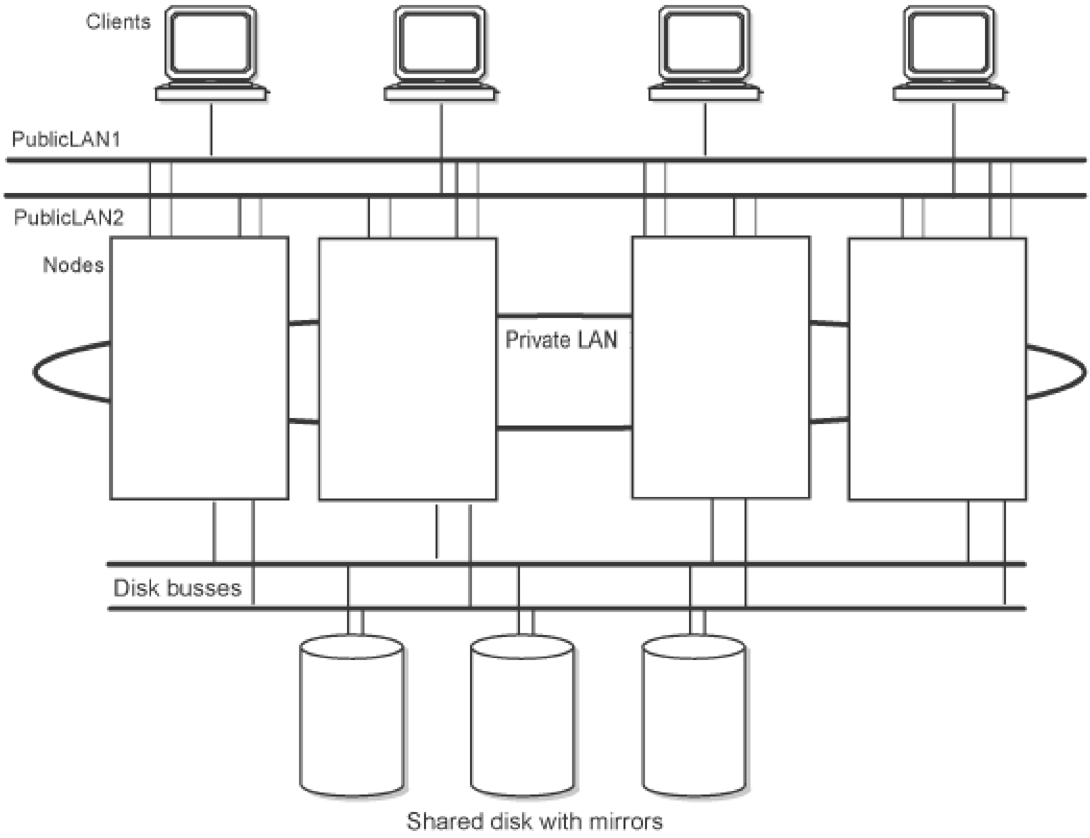

Figure 2. Example of an HACMP Cluster

Chapter 6: HACMP Cluster Configurations provides examples of cluster configurations supported by the HACMP software.

Nodes

Nodes form the core of an HACMP cluster. A node is a processor that runs AIX, the HACMP software, and the application software.

In an HACMP cluster, up to 32 System p™ servers divided into LPARS, RS/6000 or System p™ standalone systems, systems that run Parallel System Support Program (PSSP), or a combination of these cooperate to provide a set of services or resources to other entities. Clustering these servers to back up critical applications is a cost-effective high availability option. A business can use more of its computing power while ensuring that its critical applications resume running after a short interruption caused by a hardware or software failure.

In an HACMP cluster, each node is identified by a unique name. A node may own a set of resources—disks, volume groups, file systems, networks, network addresses, and applications. Cluster resources are discussed in detail in Chapter 3: HACMP Resources and Resource Groups. Typically, a node runs a server or a back end application that accesses data on the shared external disks. Applications are discussed in Chapter 5: Ensuring Application Availability.

A node in an HACMP cluster has several layers of software components. For the detailed description of these components, see the section Software Components of an HACMP Node in Chapter 4: HACMP Cluster Hardware and Software.

Shared External Disk Devices

Each node has access to one or more shared external disk devices. A *shared external disk device* is a disk physically connected to multiple nodes. The shared disk stores mission-critical data, typically mirrored or RAID-configured for data redundancy. A node in an HACMP cluster must also have internal disks that store the operating system and application binaries, but these disks are *not* shared.

Depending on the type of disk used, the HACMP software supports the following types of access to shared external disk devices—non-concurrent access and concurrent access.

- *In non-concurrent access environments*, only one connection is active at any given time, and the node with the active connection owns the disk. When a node fails, the node that currently owns the disk leaves the cluster, disk takeover occurs and a surviving node assumes ownership of the shared disk.
- *In concurrent access environments*, the shared disks are actively connected to more than one node simultaneously. Therefore, when a node fails, disk takeover is *not* required.

  Note that in such environments, either *all* nodes defined in the cluster can be part of the concurrent access, or *only a subset* of cluster nodes can have access to shared disks. In the second case, you configure resource groups only on those nodes that have shared disk access.

The differences between these methods are explained more fully in the section Shared External Disk Access in Chapter 4: HACMP Cluster Hardware and Software.

Networks

As an independent, layered component of AIX, the HACMP software is designed to work with any TCP/IP-based network. Nodes in an HACMP cluster use the network to:

- Allow clients to access the cluster nodes
- Enable cluster nodes to exchange heartbeat messages
- Serialize access to data (in concurrent access environments).

The HACMP software has been tested with Ethernet, Token-Ring, ATM, and other networks.

Types of Networks

The HACMP software defines two types of communication networks, characterized by whether these networks use communication interfaces based on the TCP/IP subsystem (TCP/IP-based) or communication devices based on non-TCP/IP subsystems (device-based).

- *TCP/IP-based network*. Connects two or more server nodes, and optionally allows client access to these cluster nodes, using the TCP/IP protocol. Ethernet, Token-Ring, ATM, HP Switch and SP Switch networks are defined as TCP/IP-based networks.

- *Device-based network*. Provides a point-to-point connection between two cluster nodes for HACMP control messages and *heartbeat* traffic. Device-based networks do *not* use the TCP/IP protocol and, therefore, continue to provide communications between nodes in the event the TCP/IP subsystem on a server node fails.

  Target mode SCSI devices, Target Mode SSA devices, *disk heartbeat* devices, or RS232 point-to-point devices are defined as device-based networks.

Clients

A client is a processor that can access the nodes in a cluster over a local area network. Clients each run a “front end” or client application that queries the server application running on the cluster node.

The HACMP software provides a highly available environment for critical data and applications on cluster nodes. *The HACMP software does not make the clients themselves highly available.* AIX clients can use the Cluster Information (Clinfo) services to receive notice of cluster events. Clinfo provides an API that displays cluster status information.

HACMP provides a cluster status utility, the **/usr/es/sbin/cluster/clstat**. It is based on Clinfo and reports the status of key cluster components—the cluster itself, the nodes in the cluster, the network interfaces connected to the nodes, and the resource groups on each node. The cluster status interface of the **clstat** utility includes web-based, Motif-based and ASCII-based versions.

See Cluster Information Program in Chapter 4: HACMP Cluster Hardware and Software, for more information about how Clinfo obtains cluster status information.

Goal of HACMP: Eliminating Scheduled Downtime

The primary goal of high availability clustering software is to minimize, or ideally, eliminate, the need to take your resources out of service during maintenance and reconfiguration activities.

HACMP software optimizes availability by allowing for the *dynamic reconfiguration* of running clusters. Most routine cluster maintenance tasks, such as adding or removing a node or changing the priority of nodes participating in a resource group, can be applied to an active cluster without stopping and restarting cluster services.

In addition, you can keep an HACMP cluster online while making configuration changes by using the *Cluster Single Point of Control (C-SPOC)* facility. C-SPOC makes cluster management easier, as it allows you to make changes to shared volume groups, users, and groups across the cluster from a single node. The changes are propagated transparently to other cluster nodes.

Chapter 2: HACMP Cluster Nodes, Sites, Networks, and Heartbeating

This chapter introduces major cluster topology-related *concepts* and definitions that are used throughout the documentation and in the HACMP user interface.

The information in this chapter is organized as follows:

- Cluster Nodes and Cluster Sites
- Cluster Networks
- Subnet Routing Requirements in HACMP
- IP Address Takeover
- IP Address Takeover via IP Aliases
- IP Address Takeover via IP Replacement
- Heartbeating over Networks and Disks.

Cluster Nodes and Cluster Sites

A typical HACMP cluster environment consists of nodes that can serve as clients or servers. If you are using the HACMP/XD software or LVM cross-site mirroring, sites or groups of nodes become part of the cluster topology.

Nodes

A node is a processor that runs both AIX and the HACMP software. Nodes may share a set of resources—disks, volume groups, file systems, networks, network IP addresses, and applications.

The HACMP software supports from two to thirty-two nodes in a cluster. In an HACMP cluster, each node is identified by a unique name. In HACMP, a node name and a hostname can usually be the same.

Nodes serve as core physical components of an HACMP cluster. For more information on nodes and hardware, see the section Nodes in Chapter 1: HACMP for AIX.

Two types of nodes are defined:

- *Server nodes* form the core of an HACMP cluster. Server nodes run services or back end applications that access data on the shared external disks.
- *Client nodes* run front end applications that retrieve data from the services provided by the server nodes. Client nodes can run HACMP software to monitor the health of the nodes, and to react to failures.

Server Nodes

A cluster *server node* usually runs an application that accesses data on the shared external disks. Server nodes run HACMP daemons and keep resources highly available. Typically, applications are run, storage is shared between these nodes, and *clients* connect to the server nodes through a *service IP address*.

Client Nodes

A full high availability solution typically includes the client machine that uses services provided by the servers. Client nodes can be divided into two categories: naive and intelligent.

- A naive client views the cluster as a single entity. If a server fails, the client must be restarted, or at least must reconnect to the server.
- An intelligent client is cluster-aware. A cluster-aware client reacts appropriately in the face of a server failure, connecting to an alternate server, perhaps masking the failure from the user. Such an intelligent client must have knowledge of the cluster state.

HACMP extends the cluster paradigm to clients by providing both dynamic cluster configuration reporting and notification of cluster state changes, such as changes in subsystems or node failure.

Sites

You can define a group of one or more server nodes as belonging to a *site.* The site becomes a component, like a node or a network, that is known to the HACMP software. HACMP supports clusters divided into two sites.

Using sites, you can configure cross-site LVM mirroring. You configure logical volume mirrors between physical volumes in separate storage arrays, and specify to HACMP which physical volumes are located at each site. Later, when you use C-SPOC to create new logical volumes, HACMP automatically displays the site location of each defined physical volume, making it easier to select volumes from different sites for LVM mirrors. For more information on cross-site LVM mirroring, see the *Planning Guide*.

In addition, the HACMP/XD (Extended Distance) feature provides three distinct software solutions for disaster recovery. These solutions enable an HACMP cluster to operate over extended distances at two sites.

- **HACMP/XD for Metro Mirror** increases data availability for IBM TotalStorage Enterprise Storage Server (ESS) volumes that use Peer-to-Peer Remote Copy (PPRC) to copy data to a remote site for disaster recovery purposes. HACMP/XD for Metro Mirror takes advantage of the PPRC fallover/fallback functions and HACMP cluster management to reduce downtime and recovery time during disaster recovery.

  When PPRC is used for data mirroring between sites, the physical distance between sites is limited to the capabilities of the ESS hardware.
- **HACMP/XD for Geographic Logical Volume Manager (GLVM)** increases data availability for IBM volumes that use GLVM to copy data to a remote site for disaster recovery purposes. HACMP/XD for GLVM takes advantage of the following components to reduce downtime and recovery time during disaster recovery:

  - AIX and HACMP/XD for GLVM data mirroring and synchronization. Both standard and enhanced concurrent volume groups can be made geographically mirrored with the GLVM utilities.
  - TCP/IP-based unlimited distance network support (up to four XD_data data mirroring networks can be configured).
  - HACMP cluster management. HACMP ensures that in case of component failures, a mirrored copy of the data is accessible at either local or remote site. Both concurrent and non-concurrent resource groups can be configured in an HACMP cluster with GLVM, however, inter-site policy should *not* be concurrent.
- **HACMP/XD for HAGEO Technology** uses the TCP/IP network to enable *unlimited distance* for data mirroring between sites. (Note that although the distance is unlimited, practical restrictions exist on the bandwidth and throughput capabilities of the network).

  This technology is based on the IBM High Availability Geographic Cluster for AIX (HAGEO) v 2.4 product. HACMP/XD for HAGEO Technology extends an HACMP cluster to encompass two physically separate data centers. Data entered at one site is sent across a point-to-point IP network and mirrored at a second, geographically distant location.

Each site can be a backup data center for the other, maintaining an updated copy of essential data and running key applications. If a disaster disables one site, the data is available within minutes at the other site. The HACMP/XD software solutions thus increase the level of availability provided by the HACMP software by enabling it to recognize and handle a site failure, to continue processing even though one of the sites has failed, and to reintegrate the failed site back into the cluster.

For information on HACMP/XD for Metro Mirror, HACMP/XD for GLVM, and HACMP/XD for HAGEO, see the documentation for each of those solutions.

Cluster Networks

Cluster nodes communicate with each other over communication networks. If one of the physical network interface cards on a node on a network fails, HACMP preserves the communication to the node by transferring the traffic to another physical network interface card on the same node. If a “connection” to the node fails, HACMP transfers resources to another node to which it has access.

In addition, RSCT sends heartbeats between the nodes over the cluster networks to periodically check on the health of the cluster nodes themselves. If HACMP detects no heartbeats from a node, a node is considered as failed and resources are automatically transferred to another node. For more information, see Heartbeating over Networks and Disks in this chapter.

We highly recommend configuring multiple communication paths between the nodes in the cluster. Having multiple communication networks prevents *cluster partitioning*, in which the nodes within each partition form their own entity. In a partitioned cluster, it is possible that nodes in each partition could allow simultaneous non-synchronized access to the same data. This can potentially lead to different views of data from different nodes.

Physical and Logical Networks

A physical network connects two or more physical network interfaces. There are many types of physical networks, and HACMP broadly categorizes them as those that use the TCP/IP protocol, and those that do *not*:

- *TCP/IP-based*, such as Ethernet or Token Ring
- *Device-based*, such as RS232 or TM SSA.

As stated in the previous section, configuring multiple TCP/IP-based networks helps to prevent cluster partitioning. Multiple device-based networks also help to prevent partitioned clusters by providing additional communications paths in cases when the TCP/IP-based network connections become congested or severed between cluster nodes.

Note: If you are considering a cluster where the physical networks use *external* networking devices to route packets from one network to another, consider the following: When you configure an HACMP cluster, HACMP verifies the connectivity and access to all interfaces defined on a particular physical network. However, HACMP cannot determine the presence of external network devices such as bridges and routers in the network path between cluster nodes. If the networks have external networking devices, ensure that you are using devices that are highly available and redundant so that they do *not* create a *single point of failure* in the HACMP cluster.

A logical network is a portion of a physical network that connects two or more logical network interfaces/devices. A logical network interface/device is the software entity that is known by an operating system. There is a one-to-one mapping between a physical network interface/device and a logical network interface/device. Each logical network interface can exchange packets with each logical network interface on the same logical network.

If a subset of logical network interfaces on the logical network needs to communicate with each other (but with no one else) while sharing the same physical network, *subnets* are used. A s*ubnet mask* defines the part of the IP address that determines whether one logical network interface can send packets to another logical network interface on the same logical network.

Logical Networks in HACMP

HACMP has its own, similar concept of a logical network. All logical network interfaces in an HACMP network can communicate HACMP packets with each other directly. Each logical network is identified by a unique name. If you use an automatic discovery function for HACMP cluster configuration, HACMP assigns a name to each HACMP logical network it discovers, such as `net_ether_01.`

An HACMP logical network may contain one or more subnets. RSCT takes care of routing packets between logical subnets.

For more information on RSCT, see Chapter 4: HACMP Cluster Hardware and Software.

Global Networks

A global network is a combination of multiple HACMP networks. The HACMP networks may be composed of any combination of physically different networks, and/or different logical networks (subnets), as long as they share the same network type, (for example, ethernet). HACMP treats the combined global network as a single network. RSCT handles the routing between the networks defined in a global network.

Global networks cannot be defined for all IP-based networks but only for those IP-based networks that are used for heartbeating.

Having multiple heartbeat paths between cluster nodes reduces the chance that the loss of any single network will result in a partitioned cluster. For example, multiple heartbeat paths between cluster nodes would be useful in a typical configuration of the SP Administrative Ethernet on two separate SP systems.

Local and Global Network Failures

When a failure occurs on a cluster network, HACMP uses *network failure events* to manage such cases. HACMP watches for and distinguishes between two types of network failure events: local network failure and global network failure events.

Local Network Failure

A *local network failure* is an HACMP event in which packets cannot be sent or received by *one* node over an HACMP logical network. This may occur, for instance, if all of the node’s network interface cards participating in the particular HACMP logical network fail. Note that in the case of a local network failure, the network is still in use by other nodes.

To handle local network failures, HACMP selectively moves the resources (on that network) from one node to another. This operation is referred to as *selective fallover.*

Global Network Failure

A global network failure is an HACMP event in which packets cannot be sent or received by *any* node over an HACMP logical network. This may occur, for instance, if the physical network is damaged.

Note: It is important to distinguish between these two terms in HACMP: a “global network” and a “global network failure event.” A global network is a combination of HACMP networks; a global network failure event refers to a failure that affects all nodes connected to any logical HACMP network, *not* necessarily a global network.

HACMP Communication Interfaces

An *HACMP communication interface* is a grouping of a logical network interface, a service IP address and a service IP label *that you defined in HACMP*. HACMP communication interfaces combine to create IP-based networks.

An HACMP communication interface is a combination of:

- *A logical network interface* is the name to which AIX resolves a port (for example, `en0`) of a physical network interface card.

- *A service IP address* is an IP address (for example, `129.9.201.1`) over which services, such as an application, are provided, and over which client nodes communicate.
- *A service IP label* is a label (for example, a hostname in the **/etc/hosts** file, or a logical equivalent of a service IP address, such as `node_A_en_service`) that maps to the service IP address.

Communication interfaces in HACMP are used in the following ways:

- A communication interface refers to IP-based networks and NICs. The NICs that are connected to a common physical network are combined into logical networks that are used by HACMP.
- Each NIC is capable of hosting several TCP/IP addresses. When configuring a cluster, you define to HACMP the IP addresses that HACMP monitors (base or boot IP addresses), and the IP addresses that HACMP keeps highly available (the service IP addresses).
- Heartbeating in HACMP occurs over communication interfaces. HACMP uses the heartbeating facility of the RSCT subsystem to monitor its network interfaces and IP addresses. HACMP passes the network topology you create to RSCT, while RSCT provides failure notifications to HACMP.

HACMP Communication Devices

HACMP also monitors network devices that are *not* capable of IP communications. These devices include RS232 connections and Target Mode (disk-based) connections.

Device-based networks are point-to-point connections that are free of IP-related considerations such as subnets and routing—each device on a node communicates with only one other device on a remote node.

Communication devices make up device-based networks. The devices have names defined by the operating system (such as `tty0`). HACMP allows you to name them as well (such as `TTY1_Device1`).

For example, an RS232 or a point-to-point connection would use a device name of `/dev/tty2` as the device configured to HACMP on each end of the connection. Two such devices need to be defined—one on each node.

Note: The previous sections that described local and global network failures are true for TCP/IP-based HACMP logical networks. For device-based HACMP logical networks, these concepts do *not* apply. However, the heartbeating process occurs on device-based networks.

Subnet Routing Requirements in HACMP

A *subnet route* defines a path, defined by a subnet, for sending packets through the logical network to an address on another logical network. AIX lets you add multiple routes for the same destination in the kernel routing table. If multiple matching routes have equal criteria, routing can be performed alternatively using one of the several subnet routes.

It is important to consider subnet routing in HACMP because of the following considerations:

- HACMP does *not* distinguish between logical network interfaces that share the same subnet route. If a logical network interface shares a route with another interface, HACMP has no means to determine its health. For more information on network routes, please see the AIX man page for the **route** command.
- Various constraints are often imposed on the IP-based networks by a network administrator or by TCP/IP requirements. The subnets and routes are also constraints within which HACMP must be configured for operation.

Note: We recommend that each communication interface on a node belongs to a unique subnet, so that HACMP can monitor each interface. This is *not* a strict requirement in all cases, and depends on several factors. In such cases where it is a requirement, HACMP enforces it. Also, ask your network administrator about the class and subnets used at your site.

Service IP Label/Address

A service IP label is a label that maps to the service IP address and is used to establish communication between client nodes and the server node. Services, such as a database application, are provided using the connection made over the service IP label.

A service IP label can be placed in a *resource group* as a resource, which allows HACMP to monitor its health and keep it highly available, either within a node or, if *IP Address Takeover* is configured, between the cluster nodes by transferring it to another node in the event of a failure.

Note: A service IP label/address is configured as part of configuring cluster resources, *not* as part of topology.

IP Alias

An *IP alias* is an IP label/address that is configured onto a network interface card *in addition to* the originally-configured IP label/address on the NIC. IP aliases are an AIX function that is supported by HACMP. AIX supports multiple IP aliases on a NIC. Each IP alias on a NIC can be configured on a separate subnet.

IP aliases are used in HACMP both as service and non-service addresses for *IP address takeover*, as well as for the configuration of the heartbeating method.

See the following sections for information on how HACMP binds a service IP label with a communication interface depending on which mechanism is used to recover a service IP label.

IP Address Takeover

If the physical network interface card on one node fails, and if there are no other accessible physical network interface cards on the same network on the same node (and, therefore, swapping IP labels of these NICs within the same node cannot be performed), HACMP may use the IP Address Takeover (IPAT) operation.

IP Address Takeover is a mechanism for recovering a service IP label by moving it to another NIC on another node, when the initial NIC fails. IPAT is useful because it ensures that an IP label over which services are provided to the client nodes remains available.

HACMP supports two methods for performing IPAT:

- *IPAT via IP Aliases* (this is the default)
- *IPAT via IP Replacement* (this method was known in previous releases as IPAT, or traditional IPAT).

Both methods are described in the sections that follow.

IPAT and Service IP Labels

The following list summarizes how IPAT manipulates the service IP label:

When IPAT via IP Aliases is used

The service IP label/address is aliased onto the same network interface as an existing communications interface.

That is, multiple IP addresses/labels are configured on the same network interface at the same time. In this configuration, all IP addresses/labels that you define must be configured on different subnets.

This method can save hardware, but requires additional subnets.

When IPAT via IP Replacement is used

The service IP label/address replaces the existing IP label/address on the network interface.

That is, only one IP label/address is configured on the same network interface at the same time. In this configuration, two IP addresses/labels on a node can share a subnet, while a backup IP label/address on the node must be on a different subnet.

This method can save subnets but requires additional hardware.

IP Address Takeover via IP Aliases

You can configure IP Address Takeover on certain types of networks using the IP aliasing network capabilities of AIX. Defining IP aliases to network interfaces allows creation of more than one IP label and address on the same network interface. IPAT via IP Aliases utilizes the gratuitous ARP capabilities available on many types of networks.

In a cluster with a network configured with *IPAT via IP Aliases*, when the resource group containing the service IP label falls over from the primary node to the target node, the initial IP labels that are used at boot time are added (and removed) as alias addresses on that NIC, or on other NICs that are available. Unlike in *IPAT via IP Replacement*, this allows a single NIC to support more than one service IP label placed on it as an alias. Therefore, the same node can host more than one *resource group* at the same time.

If the IP configuration mechanism for an HACMP network is via IP Aliases, the communication interfaces for that HACMP network must use routes that are different from the one used by the service IP address.

IPAT via IP Aliases provides the following advantages over the IPAT via IP Replacement scheme:

- Running IP Address Takeover via IP Aliases is faster than running IPAT via IP Replacement, because moving the IP address and the hardware address takes considerably longer than simply moving the IP address.
- IP aliasing allows co-existence of multiple service labels on the same network interface—you can use fewer physical network interface cards in your cluster. Note that upon fallover, HACMP equally distributes aliases between available network interface cards.

IPAT via IP Aliases is the default mechanism for keeping a service IP label highly available.

Distribution Preference for Service IP Label Aliases

By default, HACMP uses the IP Address Takeover (IPAT) via IP Aliases method for keeping the service IP labels in resource groups highly available.

At cluster startup, by default HACMP distributes all service IP label aliases across all available boot interfaces on a network using the principle of the “least load.” HACMP assigns any new service address to the interface that has the least number of aliases or *persistent IP labels* already assigned to it.

However, in some cases, it may be desirable to specify other types of allocation, or to ensure that the labels continue to be allocated in a particular manner, *not* only during startup but also during the subsequent cluster events.

For instance, you may want to allocate all service IP label aliases to the same boot interface as the one currently hosting the persistent IP label for that node. This option may be useful in VPN firewall configurations where only one interface is granted external connectivity and all IP labels (persistent and service IP label aliases) must be placed on the same interface to enable the connectivity.

You can configure a distribution preference for the aliases of the service IP labels that are placed under HACMP control.

A *distribution preference for service IP label aliases* is a network-wide attribute used to control the placement of the service IP label aliases on the physical network interface cards on the nodes in the cluster. Configuring a distribution preference for service IP label aliases does the following:

- Lets you customize the load balancing for service IP labels in the cluster, taking into account the *persistent IP labels* previously assigned on the nodes. See Persistent Node IP Labels in Chapter 7: HACMP Configuration Process and Facilities.
- Enables HACMP to redistribute the alias service IP labels according to the preference you specify.
- Allows you to configure the type of distribution preference suitable for the VPN firewall external connectivity requirements.

- Although the service IP labels may move to another network interface, HACMP ensures that the labels continue to be allocated according to the specified distribution preference. That is, the distribution preference is maintained during startup and the subsequent cluster events, such as a fallover, fallback or a change of the interface on the same node. For instance, if you specified the labels to be mapped to the same interface, the labels will remain mapped on the same interface, even if the initially configured service IP label moves to another node.
- The distribution preference is exercised as long as acceptable network interfaces are available in the cluster. HACMP always keeps service IP labels active, even if the preference cannot be satisfied.

For information on the types of distribution preference you can specify in HACMP, see the *Planning Guide.*

For information on configuring the distribution preference for service IP labels, see the *Administration Guide.*

IP Address Takeover via IP Replacement

The IP Address Takeover via IP Replacement facility moves the service IP label (along with the IP address associated with it) off a NIC on one node to a NIC on another node, should the NIC on the first node fail. IPAT via IP Replacement ensures that the service IP label that is included as a resource in a resource group in HACMP is accessible through its IP address, no matter which physical network interface card this service IP label is currently placed on.

If the IP address configuration mechanism is IP Replacement, only one communication interface for that HACMP network must use a route that is the same as the one used by the service IP address.

In conjunction with IPAT via IP Replacement (also, previously known as *traditional IPAT*), you may also configure *Hardware Address Takeover (HWAT)* to ensure that the mappings in the ARP cache are correct on the target adapter.

Heartbeating over Networks and Disks

A *heartbeat* is a type of a communication packet that is sent between nodes. Heartbeats are used to monitor the health of the nodes, networks and network interfaces, and to prevent cluster partitioning.

Heartbeating in HACMP: Overview

In order for an HACMP cluster to recognize and respond to failures, it must continually check the health of the cluster. Some of these checks are provided by the heartbeat function. Each cluster node sends heartbeat messages at specific intervals to other cluster nodes, and expects to receive heartbeat messages from the nodes at specific intervals. If messages stop being received, HACMP recognizes that a failure has occurred.

Heartbeats can be sent over:

- TCP/IP networks
- Point-to-point networks

- Shared disks.

The heartbeat function is configured to use specific paths between nodes. This allows heartbeats to monitor the health of all HACMP networks and network interfaces, as well as the cluster nodes themselves.

The TCP/IP heartbeat paths are set up automatically by RSCT; you have the option to configure point-to-point and disk paths as part of HACMP configuration.

HACMP passes the network topology you create to RSCT. RSCT Topology Services provides the actual heartbeat service, setting up the heartbeat paths, then sending and receiving the heartbeats over the defined paths. If heartbeats are *not* received within the specified time interval, Topology Services informs HACMP.

Heartbeating over TCP/IP Networks

RSCT Topology Services uses the HACMP network topology to dynamically create a set of heartbeat paths that provide coverage for all TCP/IP interfaces and networks. These paths form *heartbeat rings*, so that all components can be monitored without requiring excessive numbers of heartbeat packets.

In order for RSCT to reliably determine where a failure occurs, it must send and receive heartbeat packets over specific interfaces. This means that each NIC configured in HACMP must have an IP label on a separate subnet. There are two ways to accomplish this:

- Configure *heartbeating over IP interfaces*. If this method is used, you configure all service and non-service IP labels on separate subnets.
- Configure *heartbeating over IP Aliases*. If this method is used, you specify a base address for the heartbeat paths. HACMP then configures a set of IP addresses and subnets for heartbeating, which are totally separate from those used as service and non-service addresses. With this heartbeating method, all service and non-service IP labels can be configured on the same subnet or on different subnets. Since HACMP automatically generates the proper addresses required for heartbeating, all other addresses are free of any constraints.

  Heartbeating over IP Aliases provides the greatest flexibility for configuring boot (base) and service IP addresses at the cost of reserving a unique address and subnet range that is used specifically for heartbeating.

  Note: Although heartbeating over IP Aliases bypasses the subnet requirements for HACMP to perform the heartbeating function correctly, the existence of multiple routes to the same subnet (outside of HACMP) may produce undesired results for your application. For information on subnet requirements, see Subnet Routing Requirements in HACMP.

Heartbeating over Point-to-Point Networks

You can also configure non-IP point-to-point network connections that directly link cluster nodes. These connections can provide an alternate heartbeat path for a cluster that uses a single TCP/IP-based network. They also prevent the TCP/IP software itself from being a single point of failure in the cluster environment.

Point-to-point networks that you plan to use for heartbeating should be free of any other traffic for the exclusive use by HACMP.

You can configure non-IP point-to-point heartbeat paths over the following types of networks:

- Serial (RS232)
- Target Mode SSA
- Target Mode SCSI
- Disk heartbeating (over an enhanced concurrent mode disk).

Heartbeating over Disks

Heartbeating is supported on any shared disk that is part of an enhanced concurrent mode volume group.

Note: The volume group does *not* need to be configured as an HACMP resource.

Heartbeating over an enhanced concurrent mode disk operates with any type of disk—including those that are attached by fibre channel. This avoids the distance limitations (especially when using fibre channel connections) associated with RS232 links, making this solution more cost effective.

A single common disk serves as the heartbeat path between two cluster nodes. Enhanced concurrent mode supports concurrent read and write access to the non-data portion of the disk. Nodes use this part of the disk to periodically write heartbeat messages and read heartbeats written by the other node.

Chapter 3: HACMP Resources and Resource Groups

This chapter introduces resource-related *concepts and definitions* that are used throughout the documentation, and also in the HACMP user interface.

The information in this chapter is organized as follows:

- Cluster Resources: Identifying and Keeping Available
- Types of Cluster Resources
- Cluster Resource Groups
- Resource Group Policies and Attributes
- Resource Group Dependencies
- Sites and Resource Groups.

Cluster Resources: Identifying and Keeping Available

The HACMP software provides a highly available environment by:

- Identifying the set of *cluster resources* that are essential to processing.
- Defining the *resource group policies and attributes* that dictate how HACMP manages resources to keep them highly available at different stages of cluster operation (startup, fallover and fallback).

By identifying resources and defining resource group policies, the HACMP software makes numerous cluster configurations possible, providing tremendous flexibility in defining a cluster environment tailored to individual requirements.

Identifying Cluster Resources

Cluster resources can include both hardware and software:

- Disks
- Volume Groups
- Logical Volumes
- File Systems
- Service IP Labels/Addresses
- Applications
- Tape Resources
- Communication Links
- Fast Connect Resources, and other resources.

A processor running HACMP owns a user-defined set of resources: disks, volume groups, file systems, IP addresses, and applications. For the purpose of keeping resources highly available, sets of interdependent resources may be configured into *resource groups*.

Resource groups allow you to combine related resources into a single logical entity for easier configuration and management. The Cluster Manager handles the resource group as a unit, thus keeping the interdependent resources together on one node, *and* keeping them highly available.

Types of Cluster Resources

This section provides a brief overview of the resources that you can configure in HACMP and include into resource groups to let HACMP keep them highly available.

Volume Groups

A *volume group* is a set of physical volumes that AIX treats as a contiguous, addressable disk region. Volume groups are configured to AIX, and can be included in resource groups in HACMP. In the HACMP environment, a *shared volume group* is a volume group that resides entirely on the external disks that are shared by the cluster nodes. Shared disks are those that are physically attached to the cluster nodes and logically configured on all cluster nodes.

Logical Volumes

A *logical volume* is a set of *logical partitions* that AIX makes available as a single storage unit—that is, the logical volume is the "logical view" of a physical disk. Logical partitions may be mapped to one, two, or three physical partitions to implement mirroring.

In the HACMP environment, logical volumes can be used to support a journaled file system (non-concurrent access), or a raw device (concurrent access). Concurrent access does *not* support file systems. Databases and applications in concurrent access environments must access raw logical volumes (for example, `/dev/rsharedlv`).

A *shared logical volume* must have a unique name within an HACMP cluster.

Note: A shared volume group cannot contain an active paging space.

File Systems

A file system is written to a single logical volume. Ordinarily, you organize a set of files as a file system for convenience and speed in managing data.

Shared File Systems

In the HACMP system, a *shared file system* is a journaled file system that resides entirely in a shared logical volume.

For non-concurrent access, you want to plan shared file systems so that they will be placed on external disks shared by cluster nodes. Data resides in file systems on these external shared disks in order to be made highly available.

For concurrent access, you cannot use journaled file systems. Instead, use raw logical volumes.

Journaled File System and Enhanced Journaled File System

An Enhanced Journaled File System (JFS2) provides the capability to store much larger files than the Journaled File System (JFS). JFS2 is the default file system for the 64-bit kernel. You can choose to implement either JFS, which is the recommended file system for 32-bit environments, or JFS2, which offers 64-bit functionality.

JFS2 is more flexible than JFS because it allows you to dynamically increase and decrease the number of files you can have in a file system. JFS2 also lets you include the file system log in the same logical volume as the data, instead of allocating a separate logical volume for logs for all file systems in the volume group.

For more information on JFS2, see the *AIX Differences Guide Version 5.3*:

http://www.redbooks.ibm.com/pubs/pdfs/redbooks/sg24743.pdf

Applications

The purpose of a highly available system is to ensure that critical services are accessible to users. Applications usually need no modification to run in the HACMP environment. Any application that can be successfully restarted after an unexpected shutdown is a candidate for HACMP.

For example, all commercial DBMS products checkpoint their state to disk in some sort of transaction journal. In the event of a server failure, the fallover server restarts the DBMS, which reestablishes database consistency and then resumes processing.

If you use Fast Connect to share resources with non-AIX workstations, you can configure it as an HACMP resource, making it highly available in the event of node or network interface card failure, and making its correct configuration verifiable.

Applications are managed by defining the application to HACMP as an *application server* resource. The application server includes application start and stop scripts. HACMP uses these scripts when the application needs to be brought online or offline on a particular node, to keep the application highly available.

Note: The start and stop scripts are the main points of control for HACMP over an application. It is very important that the scripts you specify operate correctly to start and stop all aspects of the application. If the scripts fail to properly control the application, other parts of the application recovery may be affected. For example, if the stop script you use fails to completely stop the application and a process continues to access a disk, HACMP will *not* be able to bring the volume group offline on the node that failed or recover it on the backup node.

Add your application server to an HACMP resource group only after you have thoroughly tested your application start and stop scripts.

The resource group that contains the application server should also contain all the resources that the application depends on, including service IP addresses, volume groups, and file systems. Once such a resource group is created, HACMP manages the entire resource group and, therefore, all the interdependent resources in it as a single entity. (Note that HACMP coordinates the application recovery and manages the resources in the order that ensures activating all interdependent resources *before* other resources.)

In addition, HACMP includes application monitoring capability, whereby you can define a monitor to detect the unexpected termination of a process or to periodically poll the termination of an application and take automatic action upon detection of a problem.

You can configure multiple application monitors and associate them with one or more application servers. By supporting multiple monitors per application, HACMP can support more complex configurations. For example, you can configure one monitor for each instance of an Oracle parallel server in use. Or, you can configure a custom monitor to check the health of the database, and a process termination monitor to instantly detect termination of the database process.

You can also specify a mode for an application monitor. It can either track how the application is being run (running mode), or whether the application has started successfully (application startup mode). Using a monitor to watch the application startup is especially useful for complex cluster configurations.

Service IP Labels/Addresses

A *service IP label* is used to establish communication between client nodes and the server node. Services, such as a database application, are provided using the connection made over the service IP label.

A service IP label can be placed in a resource group as a resource that allows HACMP to monitor its health and keep it highly available, either within a node or, if IP address takeover is configured, between the cluster nodes by transferring it to another node in the event of a failure.

For more information about service IP labels, see Service IP Label/Address in Chapter 2: HACMP Cluster Nodes, Sites, Networks, and Heartbeating.

Note: Certain subnet requirements apply for configuring service IP labels as resources in different types of resource groups. For more information, see the *Planning Guide*.

Tape Resources

You can configure a SCSI or a Fibre Channel tape drive as a cluster resource in a non-concurrent resource group, making it highly available to two nodes in a cluster. Management of shared tape drives is simplified by the following HACMP functionality:

- Configuration of tape drives using the SMIT configuration tool
- Verification of proper configuration of tape drives
- Automatic management of tape drives during resource group start and stop operations
- Reallocation of tape drives on node failure and node recovery
- Controlled reallocation of tape drives on cluster shutdown
- Controlled reallocation of tape drives during a dynamic reconfiguration of cluster resources.

Communication Links

You can define the following communication links as resources in HACMP resource groups:

- SNA configured over LAN network interface cards
- SNA configured over X.25
- Pure X.25.

By managing these links as resources in resource groups, HACMP ensures their high availability. Once defined as members of an HACMP resource group, communication links are protected in the same way other HACMP resources are. In the event of a LAN physical network interface or an X.25 link failure, or general node or network failures, a highly available communication link falls over to another available network interface card on the same node, or on a takeover node.

- *SNA configured over LAN.* To be highly available, "SNA configured over LAN" resources need to be included in those resource groups that contain the corresponding service IP labels in them. These service IP labels, in turn, are defined on LAN network interface cards, such as Ethernet and Token Ring. In other words, the availability of the "SNA configured over LAN" resources is dependent upon the availability of service IP labels included in the resource group. If the NIC being used by the service IP label fails, and the service IP label is taken over by another interface, this interface will also take control over an "SNA configured over LAN" resource configured in the resource group.
- *SNA configured over X.25 links* and *pure X.25 links.* These links are usually, although *not* always, used for WAN connections. They are used as a means of connecting dissimilar machines, from mainframes to dumb terminals. Because of the way X.25 networks are used, these physical network interface cards are really a different class of devices that are *not* included in the cluster topology and are *not* controlled by the standard HACMP topology management methods. This means that heartbeats are *not* used to monitor X.25 link status, and you do *not* define X.25-specific networks in HACMP. To summarize, you can include X.25 links as resources in resource groups, keeping in mind that the health and availability of these resources also relies on the health of X.25 networks themselves (which are *not* configured within HACMP.)

Cluster Resource Groups

To be made highly available by the HACMP software, each resource must be included in a resource group. Resource groups allow you to combine related resources into a single logical entity for easier management.

This first section includes the basic terms and definitions for HACMP resource group attributes and contains the following topics:

- Participating Nodelist
- Default Node Priority
- Home Node
- Startup, Fallover and Fallback.

Later sections of this chapter explain how HACMP uses resource groups to keep the resources and applications highly available.

Participating Nodelist

The *participating nodelist* defines a list of nodes that can host a particular resource group. You define a nodelist when you configure a resource group.

- The participating nodelist for non-concurrent resource groups can contain *some or all* nodes in the cluster.
- The participating nodelist for concurrent resource groups should contain *all* nodes in the cluster.

Typically, this list contains all nodes sharing the same data and disks.

Default Node Priority

Default node priority is identified by the position of a node in the nodelist for a particular resource group. The first node in the nodelist has the *highest node priority*; it is also called the *home node* for a resource group. The node that is listed before another node has *a higher node priority* than the current node.

Depending on a *fallback policy* for a resource group, when a node with a higher priority for a resource group (that is currently being controlled by a lower priority node) joins or reintegrates into the cluster, it takes control of the resource group. That is, the resource group moves from nodes with lower priorities to the higher priority node.

At any given time, the resource group can have a default node priority specified by the participating nodelist. However, various resource group policies you select can override the default node priority and “create” the actual node priority according to which a particular resource group would move in the cluster.

Dynamic Node Priority

Setting a *dynamic node priority policy* allows you to use an RSCT resource variable such as “lowest CPU load” to select the takeover node for a non-concurrent resource group. With a dynamic priority policy enabled, the order of the takeover nodelist is determined by the state of the cluster at the time of the event, as measured by the selected RSCT resource variable. You can set different policies for different groups or the same policy for several groups.

Home Node

The *home node (*or *the highest priority node for this resource group)* is the first node that is listed in the participating nodelist for a non-concurrent resource group. The home node is a node that normally owns the resource group. A non-concurrent resource group may or may *not* have a home node—it depends on the startup, fallover and fallback behaviors of a resource group.

Note that due to different changes in the cluster, the group may *not* always start on the home node. It is important to differentiate between the *home node* for a resource group and the *node that currently owns it*.

The term home node is *not* used for concurrent resource groups as they are owned by multiple nodes.

Startup, Fallover and Fallback

HACMP ensures the availability of cluster resources by moving resource groups from one node to another when the conditions in the cluster change. HACMP manages resource groups by activating them on a particular node or multiple nodes at cluster startup, or by moving them to another node if the conditions in the cluster change. These are the stages in a cluster lifecycle that affect how HACMP manages a particular resource group:

- *Cluster startup.* Nodes are up and resource groups are distributed between them according to the resource group startup policy you selected.
- *Node failure.* Resource groups that are active on this node fall over to another node.
- *Node recovery.* A node reintegrates into the cluster and resource groups could be reacquired, depending on the resource group policies you select.
- *Resource failure and recovery.* A resource group may fall over to another node, and be reacquired, when the resource becomes available.
- *Cluster shutdown.* There are different ways of shutting down a cluster, one of which ensures that resource groups move to another node.

During each of these cluster stages, the behavior of resource groups in HACMP is defined by the following:

- Which node, or nodes, activate the resource group at cluster startup
- How many resource groups are allowed to be acquired on a node during cluster startup
- Which node takes over the resource group when the node that owned the resource group fails and HACMP needs to move a resource group to another node
- Whether a resource group falls back to a node that has just joined the cluster or stays on the node that currently owns it.

The resource group policies that you select determine which cluster node originally controls a resource group and which cluster nodes take over control of the resource group when the original node relinquishes control.

Each combination of these policies allows you to specify varying degrees of control over which node, or nodes, control a resource group.

To summarize, the focus of HACMP on resource group ownership makes numerous cluster configurations possible and provides tremendous flexibility in defining the cluster environment to fit the particular needs of the application. The combination of startup, fallover and fallback policies summarizes all the management policies available in previous releases without the requirement to specify the set of options that modified the behavior of “predefined” group types.

When defining resource group behaviors, keep in mind that a resource group can be taken over by one or more nodes in the cluster.

Startup, fallover and *fallback* are specific behaviors that describe how resource groups behave at different cluster stages. It is important to keep in mind the difference between fallover and fallback. These terms appear frequently in discussion of the various resource group policies.

Startup

Startup refers to the activation of a resource group on a node (or multiple nodes) on which it currently resides, or on the home node for this resource group. Resource group startup occurs during cluster startup, or initial acquisition of the group on a node.

Fallover

Fallover refers to the movement of a resource group from the node that currently owns the resource group to another active node after the current node experiences a failure. The new owner is *not* a reintegrating or joining node.

Fallover is valid only for *non-concurrent resource groups*.

Fallback

Fallback refers to the movement of a resource group from the node on which it currently resides (which is *not* a home node for this resource group) to a node that is joining or reintegrating into the cluster.

For example, when a node with a higher priority for that resource group joins or reintegrates into the cluster, it takes control of the resource group. That is, the resource group falls back from nodes with lesser priorities to the higher priority node.

Defining a fallback behavior is valid only for *non-concurrent resource groups*.

Resource Group Policies and Attributes

In HACMP 5.2 and up, you configure resource groups to use specific startup, fallover and fallback policies.

This section describes resource group attributes and scenarios, and helps you to decide which resource groups suit your cluster requirements.

This section contains the following topics:

- Overview
- Resource Group Startup, Fallover and Fallback
- Settling Time, Dynamic Node Priority and Fallback Timer
- Distribution Policy
- Cluster Networks and Resource Groups.

Overview

In HACMP 5.4.1, the policies for resource groups offer a wide variety of choices. Resource group policies can now be tailored to your needs. This allows you to have a greater control of the resource group behavior, increase resource availability, and better plan node maintenance.

The process of configuring a resource group is two-fold. First, you configure startup, fallover and fallback policies for a resource group. Second, you add specific resources to it. HACMP prevents you from configuring invalid combinations of behavior and resources in a resource group.

In addition, using resource groups in HACMP 5.4.1 potentially increases availability of cluster resources:

- You can configure resource groups to ensure that they are brought back online on reintegrating nodes during off-peak hours.
- You can specify that a resource group that contains a certain application is the only one that will be given preference and be acquired during startup on a particular node. You do so by specifying the node distribution policy. This is relevant if multiple non-concurrent resource groups can potentially be acquired on a node, but a specific resource group owns an application that is more important to keep available.
- You can specify that specific resource groups be kept together online on the same node, or kept apart online on different nodes, at startup, fallover, and fallback.
- You can specify that specific replicated resource groups be maintained online on the same site when you have a cluster that includes nodes and resources distributed across a geographic distance. (Usually this means you have installed one of the HACMP/XD products.)

For resource group planning considerations, see the chapter on planning resource groups in the *Planning Guide*.

Resource Group Startup, Fallover and Fallback

In HACMP 5.4.1, the following policies exist for individual resource groups:

Startup	• **Online on Home Node Only.** The resource group is brought online only on its home (highest priority) node during the resource group startup. This requires the highest priority node to be available (first node in the resource group's nodelist). • **Online on First Available Node.** The resource group comes online on the first participating node that becomes available. • **Online on All Available Nodes.** The resource group is brought online on all nodes. • **Online Using Distribution Policy.** Only one resource group is brought online on each node.
Fallover	• **Fallover to Next Priority Node in the List.** The resource group follows the default node priority order specified in the resource group's nodelist. • **Fallover Using Dynamic Node Priority.** Before selecting this option, select one of the three predefined dynamic node priority policies. These are based on RSCT variables, such as the node with the most memory available. • **Bring Offline (on Error Node Only).** Select this option to bring a resource group offline on a node during an error condition.
Fallback	• **Fallback to Higher Priority Node in the List.** A resource group falls back when a higher priority node joins the cluster. If you select this option, you can use the delayed fallback timer. If you do *not* configure a delayed fallback policy, the resource group falls back immediately when a higher priority node joins the cluster. • **Never Fallback**. A resource group does *not* fall back to a higher priority node when it joins the cluster.

For more information on each policy, see the *Planning* and *Administration Guide*s.

Settling Time, Dynamic Node Priority and Fallback Timer

You can configure some additional parameters for resource groups that dictate how the resource group behaves at startup, fallover and fallback. They are:

- *Settling Time.* You can configure a startup behavior of a resource group by specifying the settling time for a resource group that is currently offline. When the settling time is *not* configured, the resource group starts on the first available higher priority node that joins the cluster. If the settling time is configured, HACMP waits for the duration of the settling time period for a higher priority node to join the cluster before it activates a resource group. Specifying the settling time enables a resource group to be acquired on a node that has a higher priority, when multiple nodes are joining simultaneously. The settling time is a cluster-wide attribute that, if configured, affects the startup behavior of *all* resource groups in the cluster for which you selected Online on First Available Node startup behavior.
- *Distribution Policy.* You can configure the startup behavior of a resource group to use the node-based distribution policy. This policy ensures that during startup, a node acquires only one resource group. See the following section for more information.
- *Dynamic Node Priority.* You can configure a fallover behavior of a resource group to use one of three dynamic node priority policies. These are based on RSCT variables such as the most memory or lowest use of CPU. To recover the resource group HACMP selects the node that best fits the policy at the time of fallover.
- *Delayed Fallback Timer.* You can configure a fallback behavior of a resource group to occur at one of the predefined recurring times: daily, weekly, monthly and yearly, or on a specific date and time, by specifying and assigning a delayed fallback timer. This is useful, for instance, for scheduling the resource group fallbacks to occur during off-peak business hours.

Distribution Policy

On cluster startup, you can use a *node-based distribution policy* for resource groups. If you select this policy for several resource groups, HACMP tries to have each node acquire only *one* of those resource groups during startup. This lets you distribute your CPU-intensive applications on different nodes.

For more information on this resource group distribution policy and how it is handled during migration from previous releases, see the *Planning Guide*.

For configuration information, and for information on resource group management, see the *Administration Guide.*

Cluster Networks and Resource Groups

Starting with HACMP 5.2, all resource groups support service IP labels configured on either IPAT via IP replacement networks or on aliased networks.

A service IP label can be included in any non-concurrent resource group—that resource group could have any of the allowed startup policies except Online on All Available Nodes.

Resource Group Dependencies

HACMP supports resource group ordering and customized serial processing of resources to accommodate cluster configurations where a dependency exists between applications residing in different resource groups. With customized serial processing, you can specify that a given resource group be processed before another resource group. HACMP offers an easy way to configure parent/child dependencies between resource groups (and applications that belong to them) to ensure proper processing during cluster events.

As of HACMP 5.3, new *location dependency policies* were available for you to configure resource groups so they are distributed the way you expect *not* only when you start the cluster, but also during fallover and fallback. You can configure dependencies so that specified groups come online on *different* nodes or on the *same* nodes. HACMP processes the dependent resource groups in the proper order using parallel processing where possible and serial as necessary. You do *not* have to customize the processing.

You can configure different types of dependencies among resource groups:

- Parent/child dependencies
- Location dependencies.

The dependencies between resource groups that you configure are:

- Explicitly specified using the SMIT interface
- Established cluster-wide, *not* just on the local node
- Guaranteed to be honored in the cluster.

Child and Parent Resource Groups Dependencies

Configuring a resource group parent/child dependency allows for easier cluster configuration and control for clusters with multi-tiered applications where one application depends on the successful startup of another application, and both applications are required to be kept highly available with HACMP.

The following example illustrates the parent/child dependency behavior:

- If resource group A depends on resource group B, upon node startup, resource group B must be brought online before resource group A is acquired on any node in the cluster. Upon fallover, the order is reversed: Resource group A must be taken offline before resource group B is taken offline.
- In addition, if resource group A depends on resource group B, during a node startup or node reintegration, resource group A cannot be taken online before resource group B is brought online. If resource group B is taken offline, resource group A will be taken offline too, since it depends on resource group B.

Dependencies between resource groups offer a predictable and reliable way of building clusters with multi-tier applications. For more information on typical cluster environments that can use dependent resource groups, see Cluster Configurations with Multi-Tiered Applications in Chapter 6: HACMP Cluster Configurations.

These terms describe parent/child dependencies between resource groups:

- *A parent resource group* has to be in an online state before the resource group that depends on it (*child*) can be started.
- *A child resource group* depends on a parent resource group. It will get activated on any node in the cluster *only after* the parent resource group has been activated. Typically, the child resource group depends on some application services that the parent resource group provides.

  Upon resource group release (during fallover or stopping cluster services, for instance) HACMP brings offline a child resource group before a parent resource group is taken offline.

The following graphic illustrates the parent/child dependency relationship between resource groups.

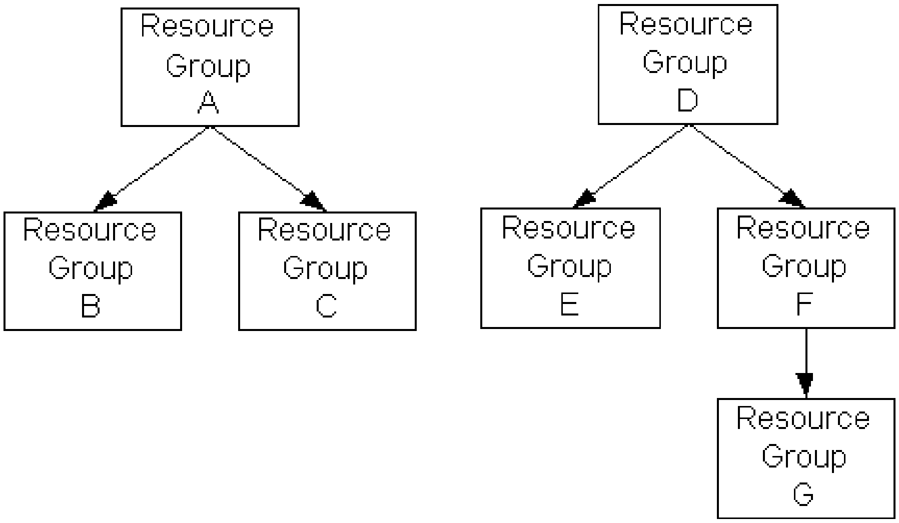

Figure 3. Example of Two and Three Levels of Dependencies between Resource Groups

The example shows relationships that were structured under these guidelines and limitations:

- You can configure a type of dependency where a parent resource group must be online on any node in the cluster before a child (dependent) resource group can be activated on a node.
- A resource group can serve as both a parent and a child resource group, depending on which end of a given dependency link it is placed.
- You can specify three levels of dependencies for resource groups.
- You cannot specify circular dependencies between resource groups.

These guidelines and limitations also apply to parent/child dependencies between resource groups:

- You can add, change or delete a dependency between resource groups, while the cluster services are running.
- When you delete a dependency between two resource groups, only the link between these resource groups is removed from the HACMP Configuration Database. The resource groups are *not* deleted.

- During fallover of a parent resource group, a child resource group containing the application temporarily goes offline and then online on any available node. The application that belongs to the child resource group is also stopped and restarted.

Resource Group Location Dependencies

In addition to various policies for individual resource groups and parent/child dependencies, HACMP 5.4.1 offers policies to handle overall resource group interdependencies. HACMP recognizes these relationships and processes the resource groups in the proper order. You can configure resource groups so that:

- Two or more specified resource groups will always be online on the same node. They start up, fall over, and fall back to the same node.
- Two or more specified resource groups will always be online on different nodes. They start up, fall over, and fall back to different nodes. You assign priorities to the resource groups so that the most critical ones are handled first in case of fallover and fallback.
- Two or more specified resource groups (with replicated resources) will always be online on the same site.

Once you configure individual resource groups with a given location dependency, they form a *set* that is handled as a unit by the Cluster Manager. The following rules apply when you move a resource group explicitly with the **clRGmove** command:

- If a resource group participates in an **Online On Same Node Dependency** set, then it can be brought online only on the node where all other resource groups from the same node set are currently online. (This is the same rule for the Cluster Manager.)
- If a resource group participates in an **Online On Same Site Dependency** set, then you can bring it online only on the site where the other resource groups from the same site set are currently online. (This is the same rule for the Cluster Manager.)
- If a resource group participates in an **Online On Different Nodes Dependency** set, then you can bring it online only on a node that does *not* host any other resource group in the different node dependency set. (This is the same rule for the Cluster Manager.) However, when you move a resource group that belongs to this set, priorities are treated as of equal value, whereas when HACMP brings these groups online it takes priorities into account.

Sample Location Dependency Model

Consider the following example, which the figure below illustrates: XYZ Publishing company follows a business continuity model that involves prioritizing the different platforms used to develop the web content. Location policies are used to keep some resource groups strictly on separate nodes and others together on the same node.

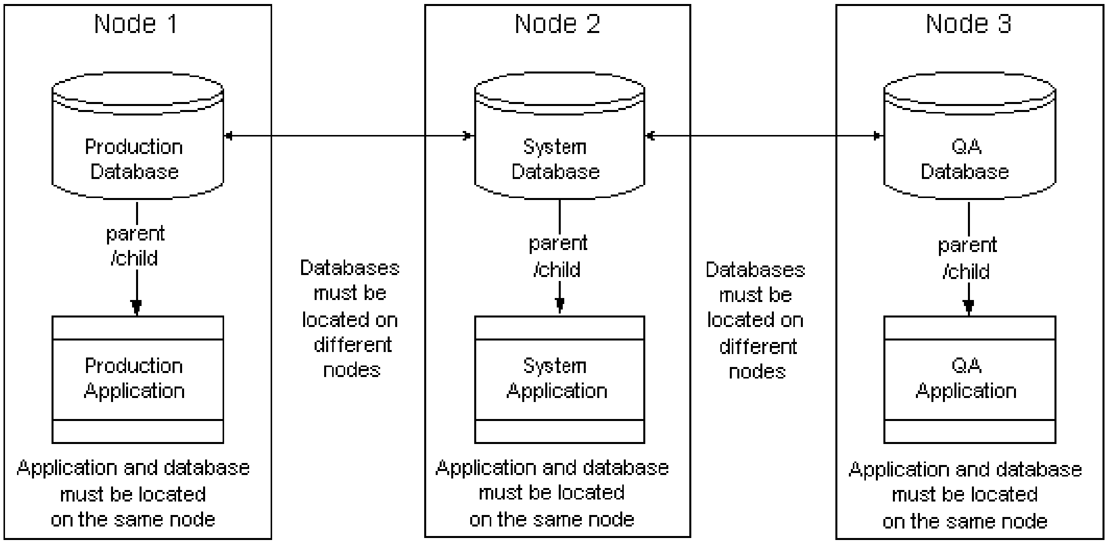
Node 1
Production Database
parent /child
Production Application
Application and database must be located on the same node
Databases must be located on different nodes
Node 2
System Database
parent /child
System Application
Application and database must be located on the same node
Databases must be located on different nodes
Node 3
QA Database
parent /child
QA Application
Application and database must be located on the same node

Figure 4. Nodes 1, 2, and 3 Being Used for Separate Databases

The figure shows how Nodes 1, 2, and 3 are used for separate databases: production, system applications and QA while their respective databases must be kept on the same node as the application.

For more information on planning location dependencies between resource groups, the application behavior in dependent resource groups and configuring dependencies that work successfully, see the *Planning Guide* and *Administration Guide*.

Sites and Resource Groups

Most HACMP configurations do *not* include sites and use the default inter-site management policy IGNORE. If you have installed an HACMP/XD component for disaster recovery, you distribute the cluster nodes between geographically separated *sites* and select one of the inter-site management policies.

You include the resources you want to replicate in resource groups. You define the startup, fallover, and fallback policies for the *primary instance* of a replicated resource group. The primary instance is where the resource group is online. The node with the primary instance of the resource group activates all the group's resources. The *secondary instance* (the replication) is activated on a node on the other site as a backup. The inter-site management policy in combination with the resource group startup, fallover, fallback policies determines the site where the primary instance is first located, and how fallover and fallback between sites is handled.

In HACMP 5.4.1, the following options exist for configuring resource group inter-site management policies:

- Prefer Primary Site
- Online On Either Site
- Online On Both Sites.

If you define sites for the cluster, then when you define the startup, fallover, and fallback policies for each resource group you want to replicate, you assign the resource group to a node on the primary site, and to a node at the other (secondary) site. The *primary instance* of the resource group runs on the primary site, the *secondary instance* runs at the secondary site.

If you have a concurrent resource group, you define it to run on all nodes. In this case, you can select the inter-site management policy Online on Both Sites. Then, the instances on both sites are active (there are no secondary instances). You can also select the other inter-site management policies so that a concurrent resource group is online on all nodes at one site, and has backup instances on the other site.

Starting with HACMP 5.3, you can also move the primary instance of a resource group across site boundaries with the **clRGmove** utility. HACMP then redistributes the peer secondary instances as necessary (or gives you a warning if the move is disallowed due to a configuration requirement).

For more information on inter-site management policies and how they work with the startup, fallover, and fallback resource group policies, see the *Planning Guide*. See also Appendix B in the *Administration Guide* for detailed examples of resource group behavior with dependencies and sites configured.

Chapter 4: HACMP Cluster Hardware and Software

This chapter describes:

- The IBM hardware that is used with HACMP and implements the base level of a highly available environment. The following IBM hardware is used with HACMP:

  Nodes
  - IBM System p™
  - RS/6000 SP System

  Disks subsystems
  - IBM Serial Storage Architecture Disk Subsystem
  - IBM 2105 Enterprise Storage Server
  - IBM 2104 Expandable Storage Plus
  - IBM TotalStorage DS8000 and DS6000 Storage Devices
  - IBM TotalStorage DS4000 Storage Servers
  - SCSI Disks
  - OEM Disks. For an OEM disks overview, and for information on installing OEM disks, see Appendix D: OEM Disk, Volume Group, and File Systems Accommodation in the *Installation Guide*.

  Networks
  Ethernet, Token-Ring, ATM, SP Switch, and other networks. For information on administering particular types of networks, see the *Installation Guide.*

  For detailed information, follow the links in the preceding table, or see the section Enhancing Availability with IBM Hardware.

  Other sections in this chapter are:

- HACMP Required and Supported Hardware
- HACMP Cluster Software

Enhancing Availability with IBM Hardware

Building a highly available cluster begins with reliable hardware. Within the AIX environment, the IBM System p™ family of machines as well as the SP and its supported disk subsystems provide a robust, stable platform for building highly available clusters.

IBM System p™

Some IBM System p™ servers, such as the 690 (Regatta), let you configure multiple logical partitions that can be used as separate nodes. The System p™ 690 delivers true logical partitioning (LPAR). Each system can be divided into as many as 16 virtual servers, each with its own set of system resources such as processors, memory and I/O. Unlike partitioning techniques available on other UNIX servers, LPAR provides greater flexibility in matching resources to workloads. Based on business requirements, these resources can be allocated or combined for business-critical applications resulting in more efficient use of the system.

With the IBM System p™ Cluster 1600 and AIX operating system, you can mix or match up to 128 units (512 via special order) including up to 32 System p™ 690 systems. An LPAR of a System p™ 690 is viewed by a Cluster 1600 as just another node or server in the cluster. Up to 16 LPARs per system and up to 128 LPARs per cluster are supported on System p™ 690. Up to 4 LPARs per system are supported on System p™ 650, and up to 2 LPARs are supported on System p™ 630.

HACMP now supports the following:

- Micro-partioning under AIX 5.3 on Power5 systems
- IBM 2 Gigabit Fibre Channel PCI-X Storage Adapter.

IBM System p™ p5 models 510, 520, 550, and 570

HACMP supports the new POWER5 -based IBM p5 servers running AIX v.5.2 and up. The System p™ p5 Express family uses the IBM POWER5™ microprocessor. The POWER5 processor can run both 32- and 64-bit applications simultaneously.

The optional IBM Virtualization Engine™ systems technologies feature provides innovations like Micro-Partitioning™ that allow the system to be finely tuned to consolidate multiple independent applications. Virtual systems can be defined as small as 1/10th of a processor and changed in increments as small as 1/100th of a processor.

IBM System p™ p5 model 575

HACMP supports the IBM p5-575 (9118-575) high-bandwidth cluster node with applicable APARs.

The IBM p5 575 delivers an 8-way, 1.9 GHz POWER5 high-bandwidth cluster node, ideal for many high-performance computing applications.

IBM System p™ p5 models 590 and 595

HACMP 5.2 and above support the IBM System p™ p5-590 and IBM System p™ p5 595. The p5 590 and p5-595 servers are powered by the IBM 64-bit Power Architecture™ microprocessor—the IBM POWER5™ microprocessor—with simultaneous multi-threading that makes each processor function as two to the operating system.

The p5-595 features a choice of IBM's fastest POWER5 processors running at 1.90 GHz or 1.65 GHz, while the p5-590 offers 1.65 GHz processors.

IBM System p ™ i5 models 520, 550 and 570 iSeries and System p ™ Convergence

HACMP supports the IBM System p™ i5, which is a new hardware platform of iSeries and System p™ convergence. You can run native AIX v.5.2 or v.5.3 with its own kernel (versus current PASE's SLIC kernel) in an LPAR partition. This is an excellent way to consolidate AIX applications and other UNIX-based applications, running in a separate System p™ box or other UNIX box, onto a single i5 platform.

You can run AIX on i5 in logical partitions allowing you to optimize your investments: Share processor and memory resources, move resources to where they are needed, exploit i5/OS storage subsystem, and leverage skills and best practices.

RS/6000 SP System

The SP is a parallel processing machine that includes from two to 128 processors connected by a high-performance switch. The SP leverages the outstanding reliability provided by the RS/6000 series by including many standard RS/6000 hardware components in its design. The SP's architecture then extends this reliability by enabling processing to continue following the failure of certain components. This architecture allows a failed node to be repaired while processing continues on the healthy nodes. You can even plan and make hardware and software changes to an individual node while other nodes continue processing.

Disk Subsystems

The disk subsystems most often shared as external disk storage in cluster configurations are:

- IBM Serial Storage Architecture Disk Subsystem
- IBM 2105 Enterprise Storage Server
- IBM 2104 Expandable Storage Plus
- IBM TotalStorage DS8000 and DS6000 Storage Devices
- IBM TotalStorage DS4000 Storage Servers
- SCSI disks.

See Appendix B on OEM disks in the *Installation Guide* for information on installing and configuring OEM disks.

For complete information on IBM Storage Solutions, see URL: http://www.storage.ibm.com

IBM Serial Storage Architecture Disk Subsystem

You can use IBM 7133 and 7131-405 SSA disk subsystems as shared external disk storage devices in an HACMP cluster configuration.

If you include SSA disks in a volume group that uses LVM mirroring, you can replace a failed drive without powering off the entire subsystem.

IBM 2105 Enterprise Storage Server

IBM 2105 Enterprise Storage Server provides multiple concurrent attachment and sharing of disk storage for a variety of open systems servers. IBM System p™ processors can be attached, as well as other UNIX and non-UNIX platforms. Attachment methods include Fibre Channel and SCSI.

The ESS uses IBM SSA disk technology (internally). ESS provides many availability features. RAID technology protects storage. RAID-5 techniques can be used to distribute parity across all disks in the array. *Sparing* is a function that allows you to assign a disk drive as a spare for availability. Predictive Failure Analysis techniques are utilized to predict errors *before* they affect data availability. Failover Protection enables one partition, or *storage cluster,* of the ESS to take over for the other so that data access can continue.

The ESS includes other features such as a web-based management interface, dynamic storage allocation, and remote services support. For more information on ESS planning, general reference material, and attachment diagrams, see URL: http://www.storage.ibm.com/disk/ess

IBM 2104 Expandable Storage Plus

The IBM 2104 Expandable Storage Plus (EXP Plus) provides flexible, scalable, and low-cost disk storage for RS/6000 and System p™ servers in a compact package. This new disk enclosure is ideal for enterprises—such as Internet or application service providers—that need high-performance external disk storage in a small footprint.

- Scales from up to 2055 GB of capacity per drawer or tower to more than 28 TB per rack
- Shared storage for all major types of servers
- Single or split-bus configuration flexibility to one or two servers
- High-performance Ultra3 SCSI disk storage with 160 MB/sec throughput
- Up to fourteen 10,000 RPM disk drives, with capacities of 9.1 GB, 18.2GB, 36.4 GB and 73.4GB and 146.8GB
- High availability to safeguard data access
- Scalability for fast-growing environments.

IBM TotalStorage DS8000 and DS6000 Storage Devices

HACMP supports the IBM TotalStorage DS8000 and DS6000 Series Disk Storage Devices with applicable APARs.

The DS8000 series incorporates IBM's POWER5 processor technology to provide functionality, flexibility, and performance for enterprise disk storage systems at improved levels of cost effectiveness while the DS6000 series brings this level of enterprise-class technology into a modular package.

IBM TotalStorage DS4000 Storage Servers

The DS4000 series (formerly named the FAStT series) has been enhanced to complement the entry and enterprise disk system offerings with DS4000 Storage Manager V9.10, enhanced remote mirror option, DS4100 Midrange Disk System (formerly named TotalStorage FAStT100 Storage Server, model 1724-100) for larger capacity configurations, and EXP100 serial ATA expansion units attached to DS4400s.

The IBM TotalStorage DS4300 (formerly FAStT600) is a mid-level disk system that can scale to over eight terabytes of fibre channel disk using three EXP700s, over sixteen terabytes of fibre channel disk with the Turbo feature using seven EXP700s. It uses the latest in storage networking technology to provide an end-to-end 2 Gbps Fibre Channel solution.

IBM DS4400 Storage Server (formerly FAStT700) delivers superior performance with 2 Gbps Fibre Channel technology. The DS4400 is designed to offer investment protection with advanced functions and flexible features. It scales from 36GB to over 32TB to support growing storage requirements created by e-business applications. DS4400 offers advanced replication services to support business continuance.

IBM DS4500 (formerly FAStT900) delivers offers up to 67.2TB of fibre channel disk storage capacity with 16 EXP700s or 16 EXP710s. DS4500 offers advanced replication services to support business continuance and disaster recovery.

The IBM System Storage DS4800 is designed with 4 gigabit per second Fibre Channel interface technology that can support up to 224 disk drives in IBM System Storage EXP810, EXP710, EXP700, or EXP100 disk units. Additionally, the DS4800 supports high-performance Fibre Channel and high-capacity serial ATA (SATA) disk drives.

For complete information about IBM Storage Solutions, see the following URL:

http://www.storage.ibm.com

HACMP Required and Supported Hardware

For a complete list of required and supported hardware, see the sales guide for the product. You can locate this document from the following URL:

http://www.ibm.com/common/ssi

After selecting your country and language, select HW and SW Desc (SalesManual, RPQ) for a Specific Information Search.

HACMP Cluster Software

This section describes the HACMP software that implements a highly available environment.

It contains the following subsections:

- Software Components of an HACMP Node
- HACMP Software Components
- Complementary Cluster Software.

Software Components of an HACMP Node

The following figure shows the layers of software on a node in an HACMP cluster:

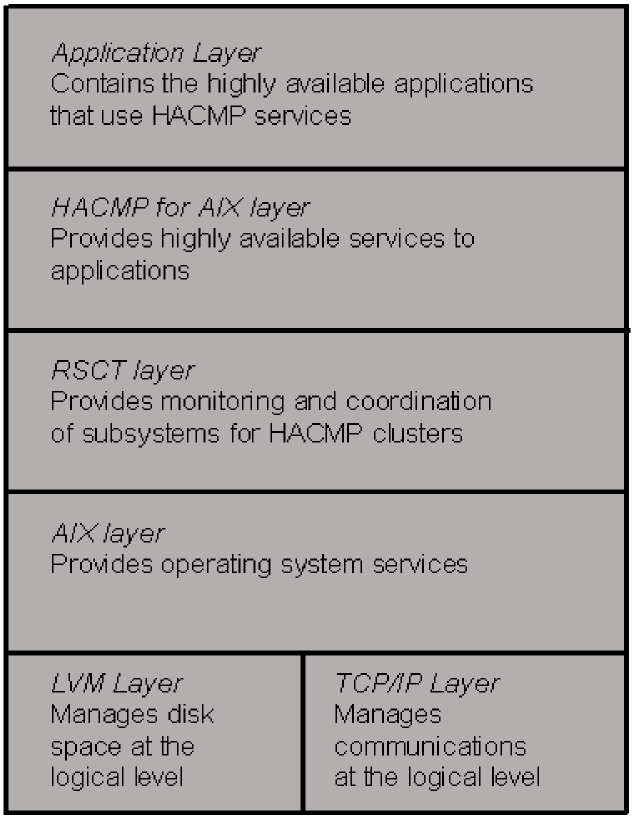
Application Layer
Contains the highly available applications that use HACMP services
HACMP for AIX layer
Provides highly available services to applications
RSCT layer
Provides monitoring and coordination of subsystems for HACMP clusters
AIX layer
Provides operating system services
LVM Layer
Manages disk space at the logical level
TCP/IP Layer
Manages communications at the logical level

Figure 5. Model of an HACMP Cluster Node

The following list describes each layer:

- Application layer. Any applications made highly available through services provided by HACMP for AIX.
- HACMP for AIX layer. Software that recognizes changes within a cluster and coordinates the use of AIX features to create a highly available environment for critical data and applications. HACMP complements lower layers by providing additional services to enable applications with their associated communications, and data storage services to be highly available.
- RSCT layer. The IBM Reliable Scalable Cluster Technology services previously packaged with HACMP/ES (prior to HACMP v 5.1) are now packaged with AIX. The RSCT layer provides facilities for monitoring node membership; network interface and communication interface health; event notification, synchronization and coordination via reliable messaging, in distributed or cluster environments. RSCT includes the Resource Monitoring and Control (RMC), Group Services, and Topology Services components. For more information, see the section IBM Reliable Scalable Cluster Technology Availability Services in this chapter and the RSCT documentation.
- AIX layer. Software that provides the underlying support for an HACMP cluster, including:

- Logical Volume Manager (LVM) subsystem layer, which manages data storage at the logical level.
- TCP/IP subsystem layer, which provides communications support for an HACMP cluster.

HACMP Software Components

The HACMP software has the following components:

- Cluster Manager
- Cluster Secure Communication Subsystem
- IBM Reliable Scalable Cluster Technology Availability Services
- Cluster Manager and SNMP Monitoring Programs
- Cluster Information Program
- Highly Available NFS Server
- Shared External Disk Access
- Concurrent Resource Manager.

Cluster Manager

Changes in the state of the cluster are referred to as *cluster events*. On each node, the Cluster Manager monitors local hardware and software subsystems for these events, such as an *application failure* event. In response to such events, the Cluster Manager runs one or more event scripts, such as a *restart application* script. Cluster Managers on all nodes exchange messages to coordinate any actions required in response to an event.

The Cluster Manager is a daemon that runs on each cluster node. The main task of the Cluster Manager is to respond to unplanned events, such as recovering from software and hardware failures, or user-initiated events, such as a joining node event. The RSCT subsystem informs the Cluster Manager about node and network-related events.

Beginning with HACMP 5.3, the Cluster Manager starts and runs independently of the RSCT stack and therefore starts and runs immediately after installation. The HACMP MIB is accessible as soon as the system comes up, even without a configured cluster. When started, the Cluster Manager responds to SNMP (Simple Network Monitoring Control) requests; however, until a full configuration is read, only a subset of MIB variables is available.

Note: In HACMP clusters, the RSCT software components—Group Services, Resource Monitoring and Control (RMC), and Topology Services—are responsible for most of the cluster monitoring tasks. For more information, see the RSCT documentation. For information on the architecture of the HACMP 5.4.1 (which includes the Enhanced Scalability functionality) product system, see the diagram in the section IBM Reliable Scalable Cluster Technology Availability Services.

Cluster Manager Connection to Other HACMP Daemons

The Cluster Manager gathers information relative to cluster state changes of nodes and interfaces. The Cluster Information Program (Clinfo) gets this information from the Cluster Manager and allows clients communicating with Clinfo to be aware of a cluster's state changes. This cluster state information is stored in the HACMP Management Information Base (MIB).

If your system is running TME 10 NetView, the connection to the Cluster Manager also allows the HAView utility to obtain cluster state information and to display it graphically through the NetView map. See Chapter 7: HACMP Configuration Process and Facilities, for information about how HAView communicates with the Cluster Manager.

Cluster Secure Communication Subsystem

HACMP has a common communication infrastructure increases the security of intersystem communication. Cluster utilities use the Cluster Communications daemon that runs on each node for communication between the nodes. Because there is only one common communications path, all communications are reliably secured.

Although most components communicate through the Cluster Communications daemon, the following components use another mechanism for inter-node communications:

Component	Communication Method
Cluster Manager	RSCT
Heartbeat messaging	RSCT
Cluster Information Program (Clinfo)	SNMP

For users who require additional security, HACMP 5.2 and up provides message authentication and encryption for messages sent between cluster nodes.

Note: HACMP 5.4.1 integrates the functionality of the SMUX Peer daemon (**clsmuxpd**) into the Cluster Manager. This integrated function eliminates the **clsmuxpd** daemon.

Connection Authentication

HACMP provides two modes for connection authentication:

- **Standard**. Standard security mode checks the source IP address against an access list, checks that the value of the source port is between 571 and 1023, and uses the principle of least-privilege for remote command execution. Standard security is the default security mode.
- **Kerberos**. Kerberos security mode uses Kerberos security for authentication. Kerberos security is available only on systems running the PSSP software (SP or IBM System p™ Cluster 1600).

For added security, you can set up a VPN for connections between nodes for HACMP inter-node communications.

Message Authentication and Encryption

Message authentication and message encryption provide additional security for HACMP messages sent between cluster nodes. Message authentication ensures the origination and integrity of a message. Message encryption changes the appearance of the data as it is transmitted and translates it to its original form when received by a node that authenticates the message.

You can configure the security options and options for distributing encryption keys using the SMIT interface.

IBM Reliable Scalable Cluster Technology Availability Services

The IBM Reliable Scalable Cluster Technology (RSCT) high availability services provide greater scalability, notify distributed subsystems of software failure, and coordinate recovery and synchronization among all subsystems in the software stack.

RSCT handles the heartbeats and network failure detection. The HACMP and RSCT software stack runs on each cluster node.

The HACMP Cluster Manager obtains indications of possible failures from several sources:

- RSCT monitors the state of the network devices
- AIX LVM monitors the state of the volume groups and disks
- Application monitors monitor the state of the applications.

The HACMP Cluster Manager drives the cluster recovery actions in the event of a component failure. RSCT running on each node exchanges a heartbeat with its peers so that it can monitor the availability of the other nodes in the cluster. If the heartbeat stops, the peer systems drive the recovery process. The peers take the necessary actions to get the critical applications running and to ensure that data has *not* been corrupted or lost.

RSCT services include the following components:

- Resource Monitoring and Control (previous versions of HACMP use the Event Management subsystem). A distributed subsystem providing a set of high availability services. It creates events by matching information about the state of system resources with information about resource conditions of interest to client programs. Client programs in turn can use event notifications to trigger recovery from system failures.
- Group Services. A system-wide, highly available facility for coordinating and monitoring changes to the state of an application running on a set of nodes. Group Services helps in both the design and implementation of highly available applications and in the consistent recovery of multiple applications. It accomplishes these two distinct tasks in an integrated framework.
- Topology Services. A facility for generating heartbeats over multiple networks and for providing information about network interface membership, node membership, and routing. Network interface and node membership provide indication of NIC and node failures respectively. Reliable Messaging uses the routing information to route messages between nodes around adapter failures.

For more information on these services, see the following URL:

http://www.ibm.com/servers/eserver/pseries/library/clusters/aix.html

The following figure shows the main components that make up the HACMP architecture:

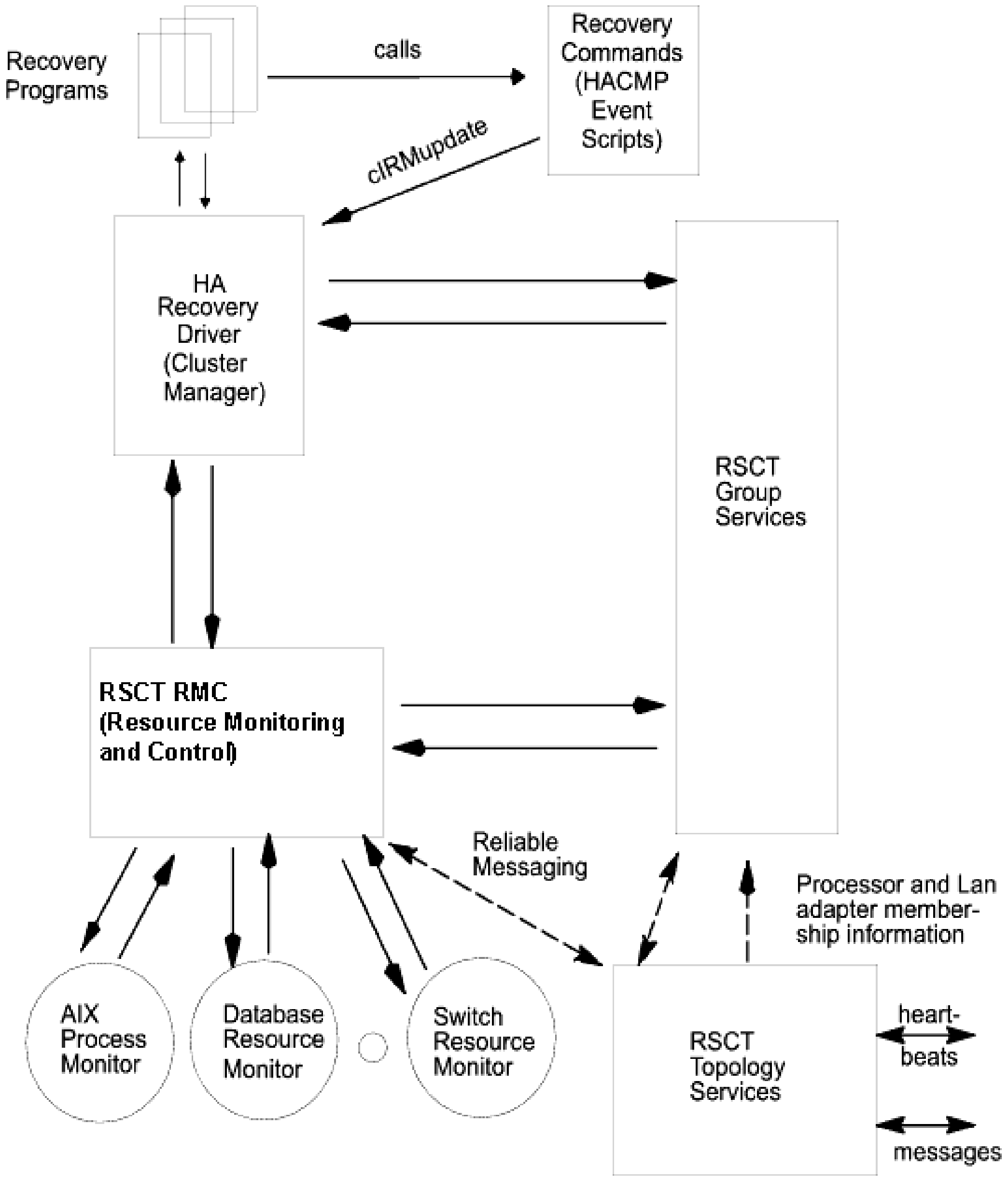
Recovery Programs
calls
Recovery Commands (HACMP Event Scripts)
clRMupdate
HA Recovery Driver (Cluster Manager)
RSCT Group Services
RSCT RMC (Resource Monitoring and Control)
Reliable Messaging
Processor and Lan adapter membership information
AIX Process Monitor
Database Resource Monitor
Switch Resource Monitor
RSCT Topology Services
heart-beats
messages

Figure 6. HACMP Comprises IBM RSCT Availability Services and the HA Recovery Driver

Cluster Manager and SNMP Monitoring Programs

An HACMP cluster is dynamic and can undergo various transitions in its state over time. For example, a node can join or leave the cluster, or another IP label can replace a service IP label on a physical network interface card. Each of these changes affects the composition of the cluster, especially when highly available clients and applications must use services provided by cluster nodes.

SNMP Support

The Cluster Manager provides Simple Network Management Protocol (SNMP) support to client applications. SNMP is an industry-standard specification for monitoring and managing TCP/IP-based networks. SNMP includes a protocol, a database specification, and a set of data objects. This set of data objects forms a Management Information Base (MIB). SNMP provides a standard MIB that includes information such as IP addresses and the number of active TCP connections. The standard SNMP agent is the **snmpd** daemon.

You can extend SNMP to include *enterprise-specific* MIBs that contain information relating to a discrete environment or application. In HACMP, the Cluster Manager maintains information about the objects defined in its MIB and passes this information on to a specialized network monitoring or network management station.

HACMP MIB

The Cluster Manager maintains cluster status information in a special HACMP MIB. When the Cluster Manager starts on a cluster node, it registers with the SNMP daemon **snmpd**, and then continually gathers cluster information. The Cluster Manager maintains an updated topology of the cluster in the HACMP MIB as it tracks events and the resulting states of the cluster. For more information on the HACMP MIB, see the *Programming Client Applications Guide*.

Cluster Information Program

The Cluster Information Program (Clinfo), the **clinfo** daemon, is an SNMP-based monitor. Clinfo, running on a client machine or on a cluster node, queries the MIB for updated cluster information. Through Clinfo, information about the state of an HACMP cluster, nodes, and networks can be made available to clients and applications.

Clients can be divided into two categories: naive and intelligent.

- A *naive* client views the cluster complex as a single entity. If a cluster node fails, the client must be restarted (or at least must reconnect to the node), if IP address takeover (IPAT) is *not* enabled.
- An *intelligent* client, on the other hand, is cluster-aware—it reacts appropriately to node failure, connecting to an alternate node and perhaps masking the failure from the user. Such an intelligent client must have knowledge of the cluster state.

Note: For conceptual information about IP address takeover, see Chapter 2: HACMP Cluster Nodes, Sites, Networks, and Heartbeating.

The HACMP software extends the benefits of highly available servers, data, and applications to clients by providing notification of cluster state changes to clients through the Cluster Manager and Clinfo API functions. See Chapter 1: Cluster Information Program in the *Programming Client Applications Guide* for more information on Clinfo.

Responding to Cluster Changes

Clinfo calls the **/usr/es/sbin/cluster/etc/clinfo.rc** script whenever a cluster, network, or node event occurs. By default, the **clinfo.rc** script flushes the system’s ARP cache to reflect changes to network IP addresses, and it does *not* update the cache until another address responds to a **ping** request. Flushing the ARP cache typically is *not* necessary if the HACMP *hardware address swapping* facility is enabled because hardware address swapping maintains the relationship between an IP address and a hardware address. Hardware address swapping is described in more detail in Chapter 5: Ensuring Application Availability.

In a switched Ethernet network, you may need to flush the ARP cache to ensure that the new MAC address is communicated to the switch, or use the procedure described in the *Troubleshooting Guide* to ensure that the hardware address is communicated correctly. (See “MAC Address Is Not Communicated to the Ethernet Switch” in Chapter 4 of the *Troubleshooting Guide*.

You can add logic to the **clinfo.rc** script if further action is desired.

Clinfo APIs

The Clinfo APIs provide application developers with both a C and a C++ language interface for accessing cluster status information. The HACMP software includes two versions of the Clinfo APIs: one for single-threaded applications and one for multi-threaded applications.

Clinfo and its associated APIs enable developers to write applications that recognize and respond to changes in a cluster. For more information, see the *Programming Client Applications Guide.*

Highly Available NFS Server

The highly available NFS server functionality is included in the HACMP product subsystem. A highly available NFS server allows a backup processor to recover current NFS activity should the primary NFS server fail. The NFS server special functionality includes highly available modifications and locks on network file systems (NFS). You can do the following:

- Use the reliable NFS server capability that preserves locks and dupcache (2-node clusters only)
- Specify a network for NFS cross-mounting
- Define NFS exports and cross-mounts at the directory level
- Specify export options for NFS-exported directories and file systems
- Configure two nodes to use NFS.

Note: While HACMP clusters can contain up to 32 nodes, clusters that use NFS can have a maximum of two nodes.

Shared External Disk Access

The HACMP software supports two methods of shared external disk access: non-concurrent and concurrent. Both methods of shared external disk access are described in the following sections.

Non-Concurrent Shared External Disk Access

In a non-concurrent environment, only one node has access to a shared external disk at a given time. If this node fails, one of the peer nodes acquires the disk, mounts file systems defined as resources, and restarts applications to restore critical services. Typically, this takes from 30 to 300 seconds, depending on the number and size of the file systems.

A *non-concurrent* configuration can use:

- SCSI disks
- SCSI disk arrays
- serial disks
- SSA disks as shared external disks

- Fibre Channel direct-attached disks
- Fibre Channel SAN-attached disks.

For more information about supported devices, see the section Disk Subsystems in this chapter.

To prevent a failed disk from becoming a single point of failure, each logical volume in a shared volume group should be mirrored using the AIX LVM facility. If you are using an IBM Enterprise Storage System or other supported RAID array, do *not* use LVM mirroring. RAID devices provide their own data redundancy.

Most software that can run in single-machine mode can be managed by the HACMP software without modification.

Non-concurrent access typically does *not* require any code changes to server programs (a database management system, for example), or to applications to provide a highly available solution. To end users, node failure looks like a very fast machine reboot. One of the surviving nodes takes ownership of the failed node’s resource groups and restarts the highly available applications. The Journaled File System, the native AIX file system, guarantees file system integrity. The server program guarantees transaction data integrity.

End users simply log onto one of the surviving nodes and restart the application. The logon and application restart procedures can be driven by the HACMP software. In some HACMP configurations, users can continue without having to take any action—they simply experience a delay during fallover.

Concurrent Shared External Disk Access

Note: For information on enhanced concurrent access, see the section Enhanced Concurrent Mode in this chapter.

The concurrent access feature enhances the benefits provided by an HACMP cluster. *Concurrent access* allows simultaneous access to a volume group on a disk subsystem attached to multiple (up to 32) nodes. Using concurrent access, a cluster can offer nearly continuous availability of data that rivals fault tolerance, but at a much lower cost. Additionally, concurrent access provides higher performance, eases application development, and allows horizontal growth.

Since concurrent access provides simultaneous access to data from multiple nodes, additional tools may be required to prevent multiple nodes from modifying the same block of data in a conflicting way. The HACMP software provides the Clinfo program that prepares an application to run in a concurrent access environment. The Clinfo API provides an API through which applications may become "cluster-aware". The Clinfo tool is described earlier in this chapter.

The benefits of concurrent shared external disk access include the following:

- *Transparent Recovery Increases Availability.* Concurrent access significantly reduces the time for a fallover—sometimes to just a few seconds—because the peer systems already have physical access to the shared disk and are running their own instances of the application.

  In a concurrent access environment, fallover basically involves backing out in-flight transactions from the failed processor. The server software running on the surviving nodes is responsible for recovering any partial transactions caused by the crash.

Since all nodes have concurrent access to the data, a client/server application can immediately retry a failed request on the surviving nodes, which continue to process incoming transactions.

- *Harnessing Multiple Processors Increases Throughput.* Applications are no longer limited to the throughput of a single processor. Instead, multiple instances of an application can run simultaneously on multiple processors. As more processing power is required, more systems can be added to the cluster to increase throughput.
- *Single Database Image Eases Application Development and Maintenance.* In a non-concurrent environment, the only route to improving performance is to partition an application and its data. Breaking code and data into pieces makes both application development and maintenance more complex.

  Splitting a database requires a high degree of expertise to make sure that the data and workload are evenly distributed among the processors.

  Partitioning code and data is *not* necessary in a concurrent access environment. To increase throughput, multiple instances of the same application running on different processors can simultaneously access a database on a shared external disk.

A *concurrent* configuration can use:

- SCSI disks
- SCSI disk arrays
- serial disks
- SSA disks as shared external disks
- Fibre Channel direct-attached disks
- Fibre Channel SAN-attached disks.

For more information about supported devices, see the section Disk Subsystems in this chapter.

When creating concurrent access logical volumes, use LVM mirroring to avoid having the disks be a single point of failure, except for RAID disk subsystems that supply their own mirroring.

Concurrent access does *not* support the use of the Journaled File System. Therefore, the database manager must write directly to the raw logical volumes or `hdisks` in the shared volume group.

An application must use some method to arbitrate all requests for shared data. Most commercial UNIX databases provide a locking model that makes them compatible with the HACMP software. Check with your database vendor to determine whether a specific application supports concurrent access processing.

Concurrent Resource Manager

The Concurrent Resource Manager of HACMP provides concurrent access to shared disks in a highly available cluster, allowing tailored actions to be taken during takeover to suit business needs.

Concurrent Resource Manager adds enhanced-concurrent support for shared volume groups on all types of disks, and concurrent shared-access management for supported RAID and SSA disk subsystems.

Enhanced Concurrent Mode

AIX provides a new form of concurrent mode: *enhanced concurrent mode*. In enhanced concurrent mode, the instances of the Concurrent Logical Volume Manager (CLVM) coordinate changes between nodes through the Group Services component of the Reliable Scalable Cluster Technology (RSCT) facility in AIX. Group Services protocols flow over the communications links between the cluster nodes. Support for enhanced concurrent mode and prior concurrent mode capabilities has the following differences:

- Any disk supported by HACMP for attachment to multiple nodes can be included in an enhanced concurrent mode volume group; the special facilities of SSA disks are *not* required.
- The same capabilities for online changes of volume group and logical volume structure that have always been supported by AIX for SSA concurrent mode volume groups are available for enhanced concurrent mode volume groups.

Keep in mind the following when planning an HACMP environment:

- When concurrent volume groups are created on AIX, they are created as enhanced concurrent mode volume groups by default.
- SSA concurrent mode is *not* supported by AIX v.5.3. You must convert them to enhanced concurrent mode.
- If one node in a concurrent resource group runs a 64-bit kernel, then it must use an enhanced concurrent mode volume group.
- SSA concurrent mode is *not* supported on 64-bit kernels.
- SSA disks with the 32-bit kernel can use SSA concurrent mode.
- The C-SPOC utility does *not* work with RAID concurrent volume groups. You need to convert them to enhanced concurrent mode (otherwise, AIX sees them as non-concurrent).
- The C-SPOC utility does *not* allow you to create *new* SSA concurrent mode volume groups on either AIX v.5.2 or 5.3. (If you upgraded from previous releases of HACMP, you can use existing volume groups in SSA concurrent mode, but C-SPOC does *not* allow you to create new groups of this type.) You can convert these volume groups to enhanced concurrent mode. If you are running AIX v.5.3, you *must* convert all volume groups to enhanced concurrent mode.
- You can include enhanced concurrent mode volume groups into shared resource groups. HACMP lists them in volume group picklists in resource group configuration SMIT panels. When enhanced concurrent volume groups are used in a non-concurrent environment, the volume groups are *not* concurrently accessed, they are still accessed by only one node at any given time.
- You can turn volume groups that are enhanced concurrent into geographically mirrored volume groups, if you have installed HACMP/XD for GLVM. For information, see Sites in Chapter 2: HACMP Cluster Nodes, Sites, Networks, and Heartbeating.

Fast Disk Takeover

Failed volume groups are taken over faster than in previous releases of HACMP due to the improved disk takeover mechanism. If you have installed AIX v. 5.2 or 5.3 and HACMP, and if you include in your non-concurrent resource groups enhanced concurrent mode volume groups, HACMP automatically detects these volume groups, and ensures that the faster option for volume group takeover is launched in the event of a node failure.

This functionality is especially useful for fallover of volume groups made up of a large number of disks.

Note: Fast disk takeover is *not* used in clusters with HACMP/XD for GLVM, if you use enhanced concurrent mode volume groups as geographically mirrored volume groups.

During fast disk takeover, HACMP skips the extra processing needed to break the disk reserves, or update and synchronize the LVM information by running lazy update. As a result, the disk takeover mechanism used for enhanced concurrent volume groups is faster than disk takeover used for standard volume groups.

In addition, enhanced concurrent volume groups are included as choices in picklists for shared resource groups in SMIT panels for adding/changing resource groups, and in C-SPOC SMIT panels.

Complementary Cluster Software

A broad range of additional tools aids you in efficiently building, managing and expanding high availability clusters in AIX environments. These include:

- General Parallel File System (GPFS) for AIX, a cluster-wide file system that allows users shared access to files that span multiple disk drives and multiple nodes.
- Workload Manager for AIX provides resource balancing between applications.
- Smart Assist software for configuring Oracle, DB2, and WebSphere in HACMP clusters.
- HACMP/XD features provide software solutions for disaster recovery. For more information, see the section on HACMP/XD in About This Guide.

Chapter 5: Ensuring Application Availability

This chapter describes how the HACMP software ensures application availability by ensuring the availability of cluster components. HACMP eliminates single points of failure for all key system components, and eliminates the need for scheduled downtime for most routine cluster maintenance tasks.

This chapter covers the following topics:

- Eliminating Single Points of Failure in an HACMP Cluster
- Minimizing Scheduled Downtime with HACMP
- Starting Cluster Services without Stopping Applications
- Minimizing Takeover Time: Fast Disk Takeover
- Maximizing Disaster Recovery
- Cluster Events.

Overview

The key facet of a highly available cluster is its ability to detect and respond to changes that could interrupt the essential services it provides. The HACMP software allows a cluster to continue to provide application services critical to an installation even though a key system component—a network interface card, for example—is no longer available. When a component becomes unavailable, the HACMP software is able to detect the loss and shift the workload from that component to another component in the cluster. In planning a highly available cluster, you attempt to ensure that key components do *not* become *single points of failure*.

In addition, HACMP software allows a cluster to continue providing application services while routine maintenance tasks are performed using a process called *dynamic reconfiguration*. In dynamic reconfiguration, you can change components in a running cluster, such as adding or removing a node or network interface, without having to stop and restart cluster services. The changed configuration becomes the active configuration dynamically. You can also dynamically replace a failed disk.

The following sections describe conceptually how to use the HACMP software to:

- Eliminate single points of failure in a cluster.
- Minimize scheduled downtime in an HACMP cluster with the dynamic reconfiguration, resource group management, and cluster management (C-SPOC) utilities.
- Minimize unscheduled downtime with the fast recovery feature, and by specifying a delayed fallback timer policy for resource groups.
- Minimize the time it takes to perform disk takeover.
- Interpret and emulate cluster events.

Note: You may need to monitor the cluster activity while a key component fails and the cluster continues providing availability of an application. For more information on which monitoring and diagnostic tools you can use, see Chapter 7: HACMP Configuration Process and Facilities.

Eliminating Single Points of Failure in an HACMP Cluster

The HACMP software enables you to build clusters that are both highly available and scalable by eliminating single points of failure (SPOF). A *single point of failure* exists when a critical cluster function is provided by a single component. If that component fails, the cluster has no other way to provide that function and essential services become unavailable.

For example, if all the data for a critical application resides on a single disk that is *not* mirrored, and that disk fails, the disk is a single point of failure for the entire system. Client nodes cannot access that application until the data on the disk is restored.

Potential Single Points of Failure in an HACMP Cluster

HACMP provides recovery options for the following cluster components:

- Nodes
- Applications
- Networks and network interfaces
- Disks and disk adapters.

To be highly available, a cluster must have no single point of failure. While the goal is to eliminate all single points of failure, compromises may have to be made. There is usually a cost associated with eliminating a single point of failure. For example, redundant hardware increases cost. The cost of eliminating a single point of failure should be compared to the cost of losing services should that component fail. The purpose of the HACMP software is to provide a cost-effective, highly available computing environment that can grow to meet future processing demands.

Eliminating Nodes as a Single Point of Failure

Nodes leave the cluster either through a planned transition (a node shutdown or stopping cluster services on a node), or because of a failure.

Node failure begins when a node monitoring a neighbor node ceases to receive heartbeat traffic for a defined period of time. If the other cluster nodes agree that the failure is a node failure, the failing node is removed from the cluster and its resources are taken over by the nodes configured to do so. An active node may, for example, take control of the shared disks configured on the failed node. Or, an active node may masquerade as the failed node (by acquiring its service IP address) and run the processes of the failed node while still maintaining its own processes. Thus, client applications can switch over to a surviving node for shared-disk and processor services.

The HACMP software provides the following facilities for processing node failure:

- Disk takeover
- IP Address Takeover via IP Aliases
- IP Address Takeover via IP Replacement (with or without Hardware Address Takeover).

Disk Takeover

In an HACMP environment, shared disks are physically connected to multiple nodes.

Disk Takeover in Concurrent Environments

In concurrent access configurations, the shared disks are actively connected to multiple nodes at the same time. Therefore, disk takeover is *not* required when a node leaves the cluster. The following figures illustrate disk takeover in concurrent environments.

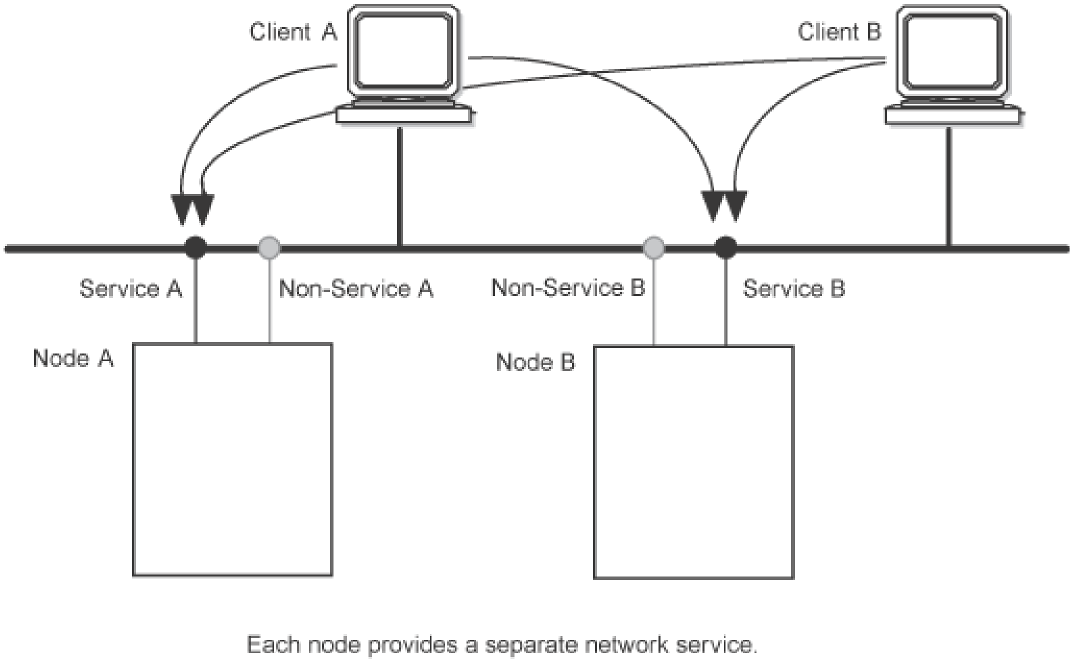

Figure 7. Concurrent Access Configuration before Disk Takeover

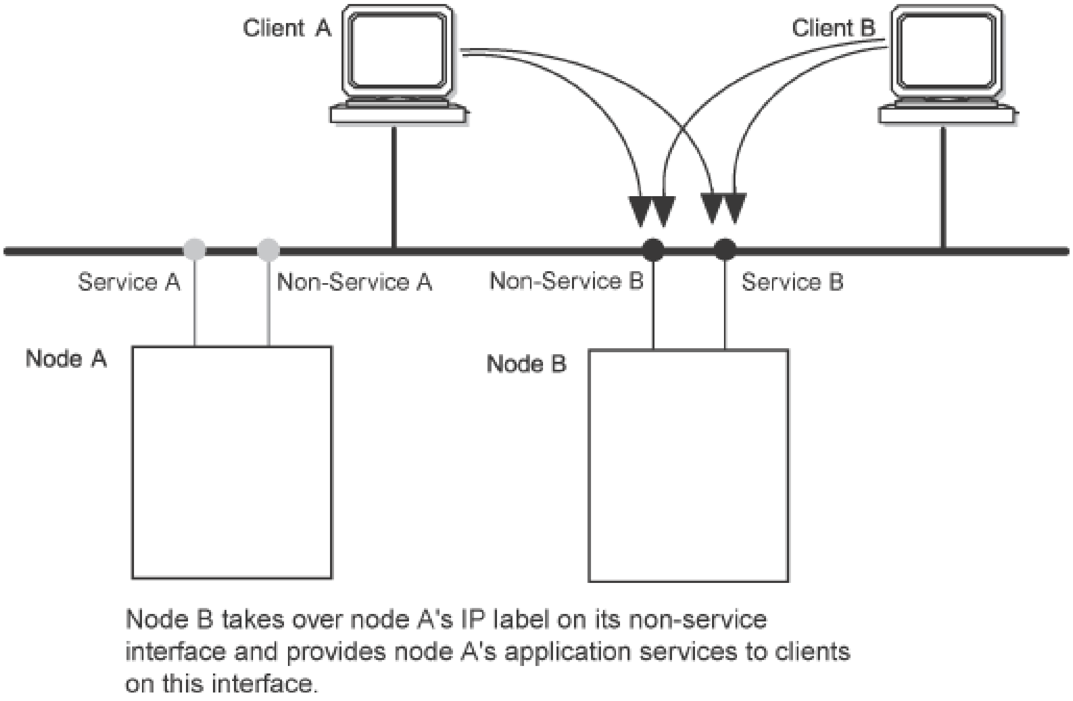
Client A
Client B
Service A
Non-Service A
Non-Service B
Service B
Node A
Node B
Node B takes over node A's IP label on its non-service interface and provides node A's application services to clients on this interface.

Figure 8. Concurrent Access Configuration after Disk Takeover

Fast Disk Takeover

In the case of a cluster failure, enhanced concurrent volume groups are taken over faster than in previous releases of HACMP due to the improved disk takeover mechanism. HACMP automatically detects enhanced concurrent volume groups and ensures that the faster option for volume group takeover is launched in the event of a node failure. For more information, see Minimizing Takeover Time: Fast Disk Takeover in this chapter.

Disk Takeover in Non-Concurrent Environments

In non-concurrent environments, only one connection is active at any given time, and the node with the active connection owns the disk. *Disk takeover* occurs when the node that currently owns the disk leaves the cluster and an active node assumes control of the shared disk so that it remains available. Note, however, that shared file systems can be exported and NFS cross-mounted by other cluster nodes that are under the control of HACMP.

The **cl_export_fs** utility can use the optional **/usr/es/sbin/cluster/etc/exports** file instead of the standard **/etc/exports** file for determining export options.

IP Address Takeover

IP address takeover (IPAT) is a networking capability that allows a node to acquire the network address of a node that has left the cluster. IP address takeover is necessary in an HACMP cluster when a service being provided to clients is bound to a specific IP address, that is, when a service IP label through which services are provided to the clients is included as a resource in a cluster resource group. If, instead of performing an IPAT, a surviving node simply did a disk and application takeover, clients would *not* be able to continue using the application at the specified server IP address.

HACMP uses two types of IPAT:

- IPAT via IP Aliases (the default)
- IPAT via IP Replacement.

For more information on each type, see IP Address Takeover via IP Aliases and IP Address Takeover via IP Replacement in Chapter 2: HACMP Cluster Nodes, Sites, Networks, and Heartbeating.

The following figures illustrate IP address takeover via IP Replacement.

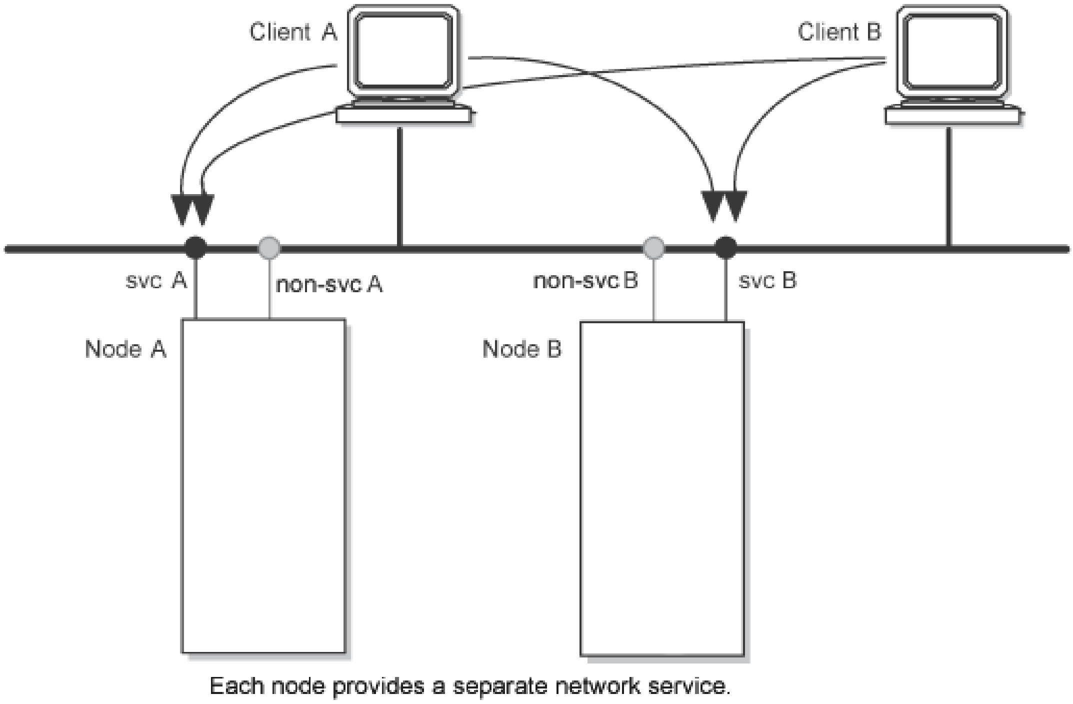

Figure 9. Configuration before IP Address Takeover via IP Replacement

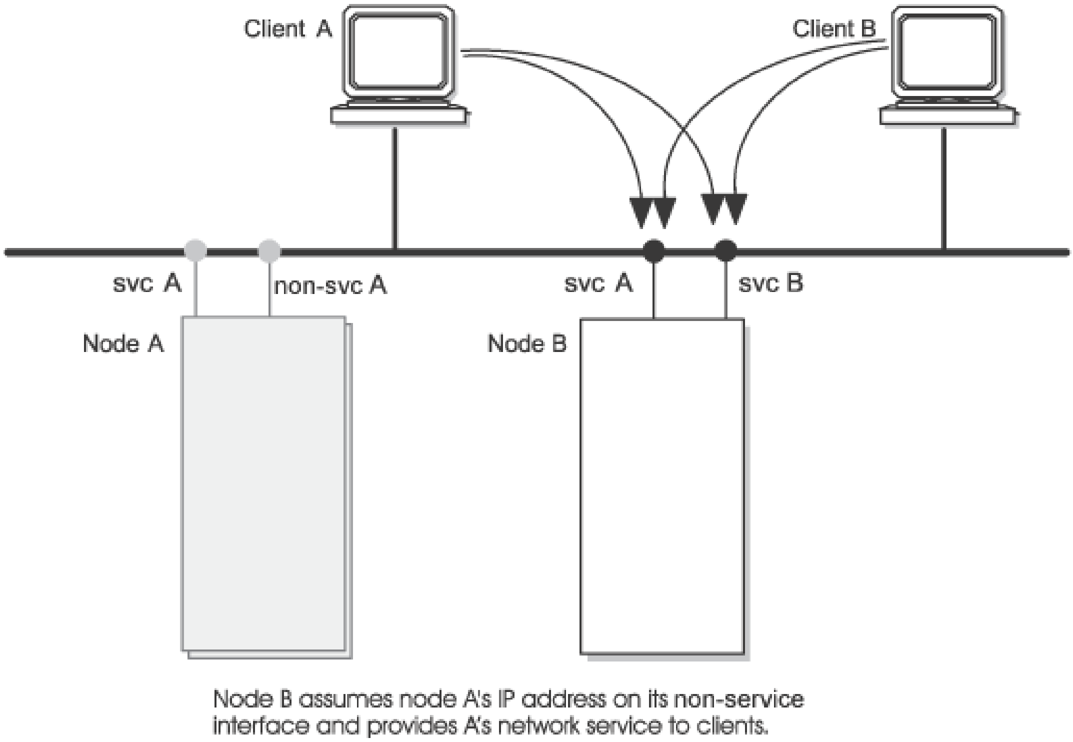
Client A
Client B
svc A
non-svc A
svc A
svc B
Node A
Node B
Node B assumes node A's IP address on its non-service interface and provides A's network service to clients.

Figure 10. Configuration after IP Address Takeover via IP Replacement

Note: In HACMP on the RS/6000 SP, special considerations apply to IP address takeover on the SP Switch network. For more information, see the section Planning for IPAT with the SP Switch Networking Chapter 3: Planning Cluster Network Connectivity in the *Planning Guide*.

Hardware Address Swapping and IP Address Takeover via IP Replacement

Hardware address swapping works in conjunction with IP address takeover via IP Replacement. With hardware address swapping enabled, a node also assumes the hardware network address (in addition to the IP address) of a node that has failed so that it can provide the service that the failed node was providing to the client nodes in the cluster. Hardware address swapping is also referred to as *hardware address takeover* (HWAT).

Without hardware address swapping, TCP/IP clients and routers that reside on the same subnet as the cluster nodes must have their Address Resolution Protocol (ARP) cache updated. The ARP cache contains a mapping of IP addresses to hardware addresses. The use of hardware address swapping is highly recommended for clients that cannot run the Clinfo daemon (machines *not* running AIX) or that cannot easily update their ARP cache.

Note: SP Switch networks do *not* support hardware address swapping. However, note that the SP switch networks can be configured so that their IP network interface cards update their ARP caches automatically when IP address takeover occurs. IP aliases are used in such cases.

Keep in mind that when an IP address takeover occurs, the netmask of the physical network interface card on which a service IP label is configured is obtained by the network interface card on another node; thus, the netmask follows the service IP address.

This means that with IPAT via IP Replacement, the netmask for all network interfaces in an HACMP network must be the same to avoid communication problems between network interfaces after an IP address takeover via IP Replacement, and during the subsequent release of the IP address acquired during takeover. The reasoning behind this requirement is as follows:

Communication problems occur when the network interface card (NIC) on another node releases the service IP address. This NIC assumes its original address, but retains the netmask of the service IP address. This address reassignment causes the NIC on another node to function on a different subnet from other backup NICs in the network. This netmask change can cause changes in the broadcast address and the routing information such that other backup NICs may now be unable to communicate on the same logical network.

Eliminating Applications as a Single Point of Failure

The primary reason to create HACMP clusters is to provide a highly available environment for mission-critical applications. For example, an HACMP cluster could run a database server program that services client applications. The clients send queries to the server program that responds to their requests by accessing a database, stored on a shared external disk.

In an HACMP cluster, these critical applications can be a single point of failure. To ensure the availability of these applications, the node configured to take over the resources of the node leaving the cluster should also restart these applications so that they remain available to client processes.

You can make an application highly available by using:

- An application server
- Cluster control
- Application monitors
- Application Availability Analysis Tool.

To put the application under HACMP control, you create an *application server* cluster resource that associates a user-defined name of the server with the names of user-provided written scripts to start and stop the application. By defining an application server, HACMP can start another instance of the application on the takeover node when a fallover occurs.

Certain applications can be made highly available without application servers. You can place such applications under cluster control by configuring an aspect of the application as part of a resource group. For example, Fast Connect services can all be added as resources to a cluster resource group, making them highly available in the event of node or network interface failure.

Note: Application takeover is usually associated with IP address takeover. If the node restarting the application also acquires the IP service address on the failed node, the clients only need to reconnect to the same server IP address. If the IP address was *not* taken over, the client needs to connect to the new server to continue accessing the application.

Additionally, you can use the AIX System Resource Controller (SRC) to monitor for the presence or absence of an application daemon and to respond accordingly.

Application Monitors

You can also configure an *application monitor* to check for process failure or other application failures and automatically take action to restart the application.

In HACMP 5.2 and up, you can configure multiple application monitors and associate them with one or more application servers. By supporting multiple monitors per application, HACMP can support more complex configurations. For example, you can configure one monitor for each instance of an Oracle parallel server in use. Or, you can configure a custom monitor to check the health of the database, and a process termination monitor to instantly detect termination of the database process.

Application Availability Analysis Tool

The Application Availability Analysis tool measures the exact amount of time that any of your applications have been available. The HACMP software collects, time-stamps, and logs extensive information about the applications you choose to monitor with this tool. Using SMIT, you can select a time period and the tool displays uptime and downtime statistics for a specific application during that period.

Eliminating Communication Interfaces as a Single Point of Failure

The HACMP software handles failures of network interfaces on which a service IP label is configured. Two types of such failures are:

- Out of two network interfaces configured on a node, the network interface with a service IP label fails, but an additional “backup” network interface card remains available on the *same* node. In this case, the Cluster Manager swaps the roles of these two interface cards on that node. Such a network interface failure is transparent to you except for a small delay while the system reconfigures the network interface on a node.
- Out of two network interfaces configured on a node, an additional, or a “backup” network interface fails, but the network interface with a service IP label configured on it remains available. In this case, the Cluster Manager detects a “backup” network interface failure, logs the event, and sends a message to the system console. If you want additional processing, you can customize the processing for this event.

The following figures illustrate network interface swapping that occurs on the *same* node:

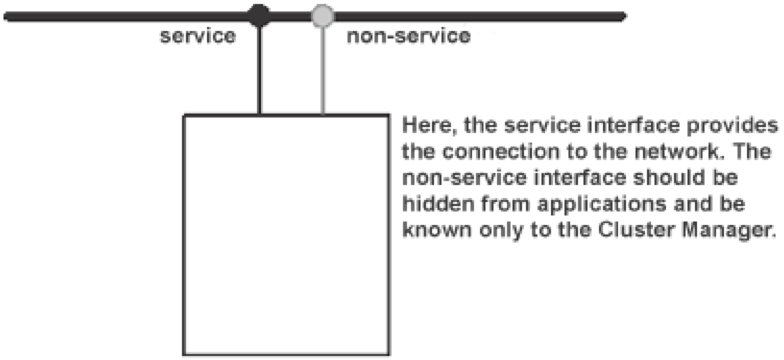
service
non-service
Here, the service interface provides the connection to the network. The non-service interface should be hidden from applications and be known only to the Cluster Manager.

Figure 11. Configuration before Network Adapter Swap

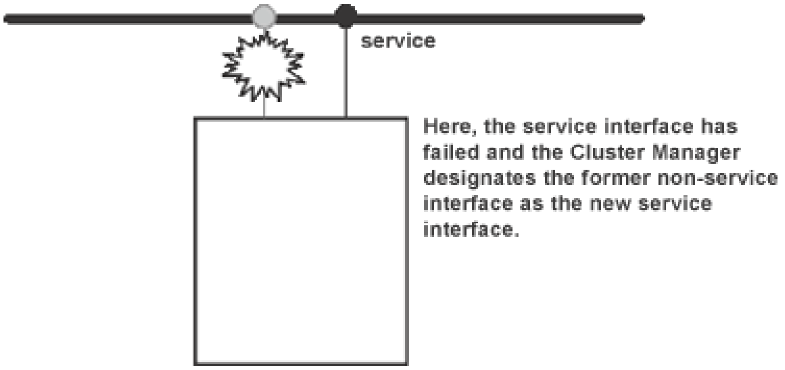
service
Here, the service interface has failed and the Cluster Manager designates the former non-service interface as the new service interface.

Figure 12. Configuration after Network Adapter Swap

Hardware Address Swapping and Adapter Swapping

Hardware address swapping works in conjunction with adapter swapping (as well as IP address takeover via IP Replacement). With hardware address swapping enabled, the “backup” network interface assumes the hardware network address (in addition to the IP address) of the failed network interface that had the service IP label configured on it so that it can provide the service that the failed network interface was providing to the cluster clients.

Without hardware address swapping, TCP/IP clients and routers that reside on the same subnet as the cluster nodes must have their Address Resolution Protocol (ARP) cache updated. The ARP cache contains a mapping of IP addresses to hardware addresses. The use of hardware address swapping is highly recommended for clients that cannot run the Clinfo daemon (machines *not* running AIX), or that cannot easily update their ARP cache.

Note: SP Switch networks do *not* support hardware address swapping. However, note that the SP switch networks can be configured such that their IP network interfaces update their ARP caches automatically when IP Address Takeover via IP Aliases occurs. For more information, see the *Administration Guide.*

Eliminating Networks as a Single Point of Failure

Network failure occurs when an HACMP network fails for all the nodes in a cluster. This type of failure occurs when none of the cluster nodes can access each other using any of the network interface cards configured for a given HACMP network.

The following figure illustrates a network failure:

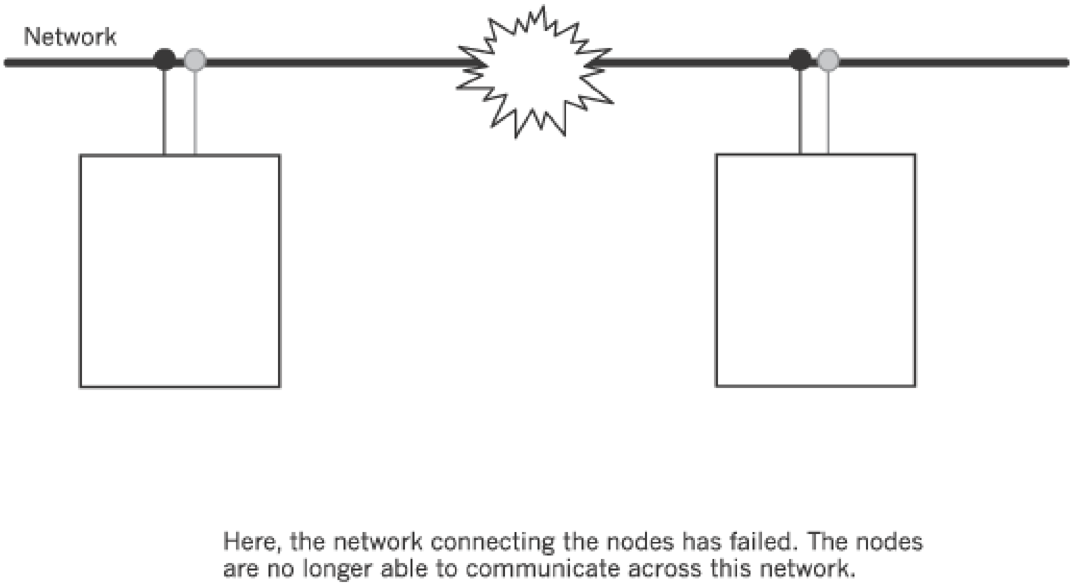
Network
Here, the network connecting the nodes has failed. The nodes are no longer able to communicate across this network.

Figure 13. Network Failure

The HACMP software's first line of defense against a network failure is to have the nodes in the cluster connected by multiple networks. If one network fails, the HACMP software uses a network that is still available for cluster traffic and for monitoring the status of the nodes.

You can specify additional actions to process a network failure—for example, re-routing through an alternate network. Having at least two networks to guard against network failure is highly recommended.

When a local network failure event occurs, the Cluster Manager takes selective recovery actions for resource groups containing a service IP label connected to that network. The Cluster Manager attempts to move only the resource groups affected by the local network failure event, rather than all resource groups on a particular node.

Node Isolation and Partitioned Clusters

Node isolation occurs when all networks connecting two or more parts of the cluster fail. Each group (one or more) of nodes is completely isolated from the other groups. A cluster in which certain groups of nodes are unable to communicate with other groups of nodes is a *partitioned cluster.*

In the following illustration of a partitioned cluster, Node A and Node C are on one side of the partition and Node B and Node D are on the other side of the partition.

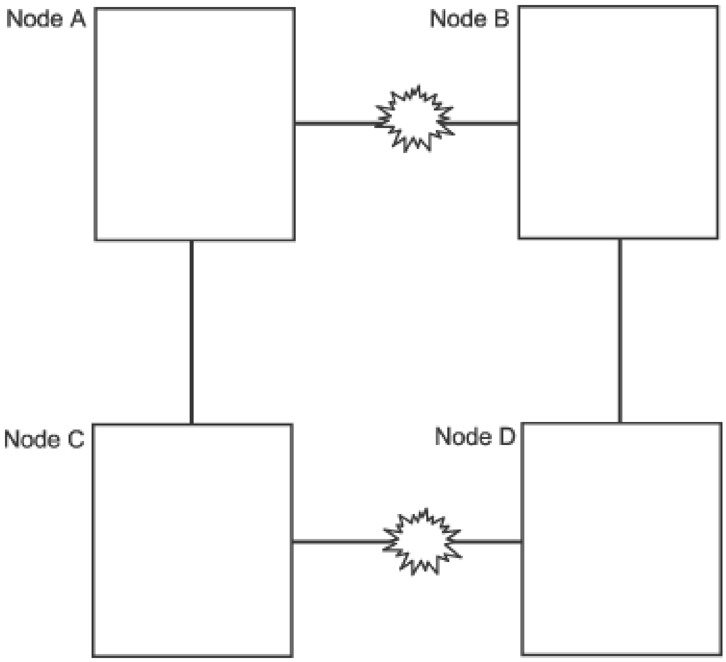

Figure 14. Partitioned Cluster

The problem with a partitioned cluster is that the nodes on one side of the partition interpret the absence of heartbeats from the nodes on the other side of the partition to mean that those nodes have failed and then generate node failure events for those nodes. Once this occurs, nodes on each side of the cluster (if so configured) attempt to take over resources from a node that is still active and, therefore, still legitimately owns those resources. These attempted takeovers can cause unpredictable results in the cluster—for example, data corruption due to a disk being reset.

Using Device-Based Networks to Prevent Partitioning

To guard against the TCP/IP subsystem failure causing node isolation, each node in the cluster should be connected by a point-to-point non-IP-based network to its neighboring nodes, forming a logical “ring.” This logical ring of point-to-point networks reduces the chance of node isolation by allowing neighboring Cluster Managers to communicate even when all TCP/IP-based networks fail.

You can configure two kinds of point-to-point, non-IP-based networks in HACMP:

- Point-to-point networks, which use serial network interface cards and RS232 connections. *Not* all serial ports can be used for this function. For more information, see the *Planning Guide*.
- Disk networks, which use a shared disk and a disk bus as a point-to-point network. Any disk that is included in an HACMP enhanced concurrent volume group can be used. (You can also use TM SSA or SCSI disks that are *not* included in an enhanced concurrent volume group).

Point-to-point, device-based networks are especially important in concurrent access configurations so that data does *not* become corrupted when TCP/IP traffic among nodes is lost. Device-based networks do *not* carry TCP/IP communication between nodes; they only allow nodes to exchange heartbeats and control messages so that Cluster Managers have accurate information about the status of peer nodes.

Using Global Networks to Prevent Partitioning

You can also configure a "logical" global network that groups multiple networks of the same type. Global networks help to avoid node isolation when an HACMP cluster network fails.

Eliminating Disks and Disk Adapters as a Single Point of Failure

The HACMP software does *not* itself directly handle disk and disk adapter failures. Rather, AIX handles these failures through LVM mirroring on disks and by internal data redundancy on the IBM 2105 ESS and SSA disks.

For example, by configuring the system with multiple SCSI-3 chains, serial adapters, and then mirroring the disks across these chains, any single component in the disk subsystem (adapter, cabling, disks) can fail without causing unavailability of data on the disk.

If you are using the IBM 2105 ESS and SSA disk arrays, the disk array itself is responsible for providing data redundancy.

AIX Error Notification Facility

The AIX Error Notification facility allows you to detect an event *not* specifically monitored by the HACMP software—a disk adapter failure, for example—and to program a response to the event.

Permanent hardware errors on disk drives, controllers, or adapters can affect the fault resiliency of data. By monitoring these errors through error notification methods, you can assess the impact of a failure on the cluster's ability to provide high availability. A simple implementation of error notification would be to send a mail message to the system administrator to investigate the problem further. A more complex implementation could include logic to analyze the failure and decide whether to continue processing, stop processing, or escalate the failure to a node failure and have the takeover node make the volume group resources available to clients.

It is strongly recommended that you implement an error notification method for all errors that affect the disk subsystem. Doing so ensures that degraded fault resiliency does *not* remain undetected.

AIX error notification methods are automatically used in HACMP to monitor certain recoverable LVM errors, such as volume group loss errors.

Automatic Error Notification

You can automatically configure error notification for certain cluster resources using a specific option in SMIT. If you select this option, error notification is turned on automatically on all nodes in the cluster for particular devices.

Certain non-recoverable error types are supported by automatic error notification: disk, disk adapter, and SP switch adapter errors. This feature does *not* support media errors, recovered errors, or temporary errors. One of two error notification methods is assigned for all error types supported by automatic error notification.

In addition, if you add a volume group to a resource group, HACMP creates an AIX Error Notification method for it. In the case where a volume group loses quorum, HACMP uses this method to selectively move the affected resource group to another node. Do *not* edit or alter the error notification methods that are generated by HACMP.

Error Emulation

The Error Emulation utility allows you to test your error notification methods by simulating an error. When the emulation is complete, you can check whether your customized notification method was exercised as intended.

Minimizing Scheduled Downtime with HACMP

The HACMP software enables you to perform most routine maintenance tasks on an active cluster dynamically—without having to stop and then restart cluster services to make the changed configuration the active configuration. Several features contribute to this:

- Starting Cluster Services without Stopping Applications
- Dynamic Automatic Reconfiguration (DARE)
- Resource Group Management
- Cluster Single Point of Control (C-SPOC)
- Dynamic Adapter Swap
- Automatic Verification and Synchronization.

Starting Cluster Services without Stopping Applications

In HACMP 5.4.1, you can start the HACMP cluster services on the node(s) without stopping your applications. For more information on configuring application monitoring and steps needed to start cluster services without stopping the applications, see the chapter on Starting and Stopping Cluster Services in the *Administration Guide*.

Dynamic Automatic Reconfiguration (DARE)

This process, called *dynamic automatic reconfiguration* or *dynamic reconfiguration (DARE)*, is triggered when you synchronize the cluster configuration after making changes on an active cluster. Applying a cluster snapshot using SMIT also triggers a dynamic reconfiguration event.

For example, to add a node to a running cluster, you simply connect the node to the cluster, add the node to the cluster topology on any of the existing cluster nodes, and synchronize the cluster. The new node is added to the cluster topology definition on all cluster nodes and the changed configuration becomes the currently active configuration. After the dynamic reconfiguration event completes, you can start cluster services on the new node.

HACMP verifies the modified configuration before making it the currently active configuration to ensure that the changes you make result in a valid configuration.

How Dynamic Reconfiguration Works

With Dynamic Reconfiguration of a running cluster, whenever HACMP starts, it creates a private copy of the HACMP-specific object classes stored in the system default Object Data Model (ODM). From now on, the ODM is referred to as the HACMP Configuration Database.

Two directories store configuration database data:

- The Active Configuration Directory (ACD), a private directory, stores the HACMP Configuration Database data for reference by all the HACMP daemons, scripts, and utilities on a running node.
- The Default Configuration Directory (DCD), the system default directory, stores HACMP configuration database and data.

Note: The operation of DARE is described here for completeness. No manual intervention is required to ensure that HACMP carries out these operations. HACMP correctly manages all dynamic reconfiguration operations in the cluster.

The DCD is the directory named **/etc/objrepos**. This directory contains the default system object classes, such as the customized device database (CuDv) and the predefined device database (PdDv), as well as the HACMP-specific object classes. The ACD is **/usr/es/sbin/cluster/etc/objrepos/active**.

Note: When you configure a cluster, you modify the HACMP configuration database data stored in the DCD—*not* data in the ACD. SMIT and other HACMP configuration utilities all modify the HACMP configuration database data in the DCD. In addition, all user commands that display HACMP configuration database data, such as the **cllsif** command, read data from the DCD.

The following figure illustrates how the HACMP daemons, scripts, and utilities all reference the ACD when accessing configuration information.

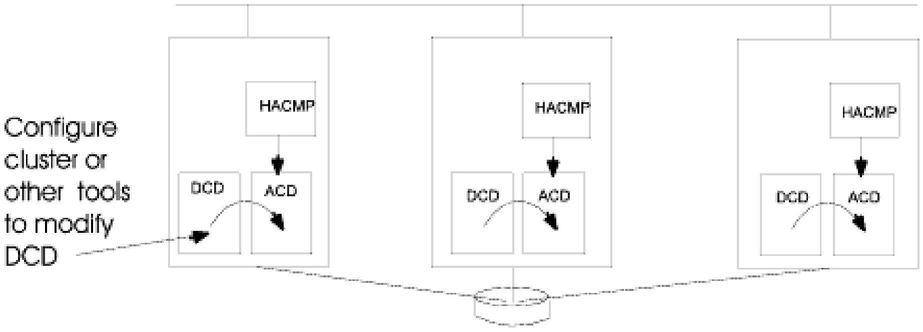

Figure 15. Relationship of HACMP to ACD at Cluster Start-Up

Reconfiguring a Cluster Dynamically

The HACMP software depends on the location of certain HACMP configuration database repositories to store configuration data. The presence or absence of these repositories is sometimes used to determine steps taken during cluster configuration and operation. The ODMPATH environment variable allows HACMP configuration database commands and subroutines to query locations other than the default location (held in the ODMDIR environment variable) if the queried object does *not* exist in the default location. You can set this variable, but it must *not* be set to include the **/etc/objrepos** directory or you will lose the integrity of the HACMP configuration information.

To change the configuration of an active cluster, you modify the cluster definition stored in the HACMP-specific HACMP configuration database classes stored in the DCD using SMIT. When you change the cluster configuration in an active cluster, you use the same SMIT paths to make the changes, but the changes do *not* take effect immediately. Therefore, you can make several changes in one operation. When you synchronize your configuration across all cluster nodes, a cluster-wide dynamic reconfiguration event occurs. When HACMP processes a dynamic reconfiguration event, it updates the HACMP configuration database object classes stored in the DCD on each cluster and replaces the HACMP configuration database data stored in the ACD with the new HACMP configuration database data in the DCD, in a coordinated, cluster-wide transition. It also refreshes the cluster daemons so that they reference the new configuration data.

After this processing, the cluster heartbeat is suspended briefly and the cluster is in an unstable state. The changed configuration becomes the active configuration. After cluster services are started on the newly added node, it is automatically integrated into the cluster.

The following figure illustrates the processing involved with adding a node to an active cluster using dynamic reconfiguration.

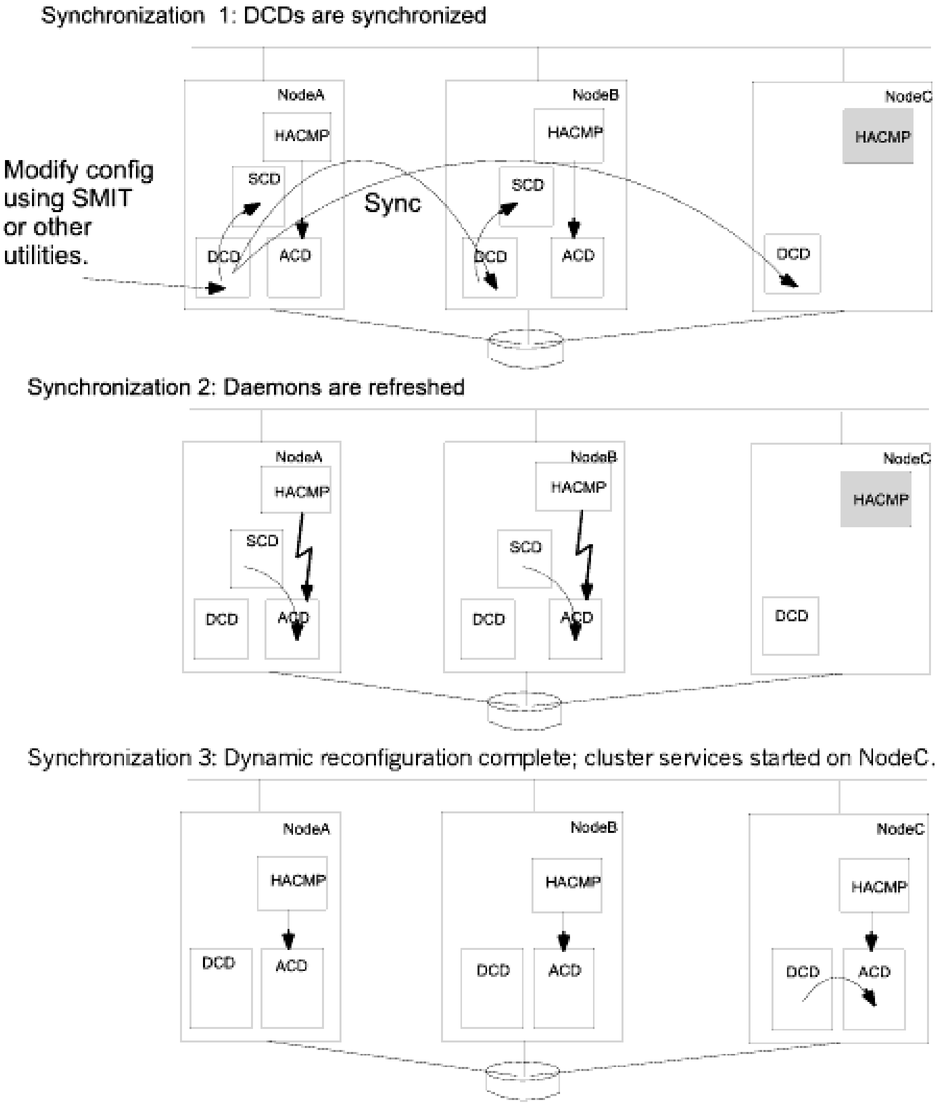
Synchronization 1: DCDs are synchronized
NodeA
NodeB
NodeC
HACMP
SCD
DCD
ACD
Sync
Modify config using SMIT or other utilities.
Synchronization 2: Daemons are refreshed
Synchronization 3: Dynamic reconfiguration complete; cluster services started on NodeC.

Figure 16. Dynamic Reconfiguration Processing

The node to be added is connected to a running cluster, but cluster services are inactive on this node. The configuration is redefined on NodeA. When the changes to the configuration are synchronized, the HACMP configuration database data stored in the DCD on NodeA is copied to the DCDs on other cluster nodes and a dynamic reconfiguration event is triggered. HACMP copies the new HACMP configuration database data in the DCD into a temporary location on each node, called the Staging Configuration Directory (SCD). The location of the SCD is **/usr/es/sbin/cluster/etc/objrepos/stage**. By using this temporary location, HACMP allows you to start making additional configuration changes while a dynamic reconfiguration is in progress. Before copying the new HACMP configuration database data in the SCD over the current HACMP configuration database data in the ACD, HACMP verifies the new configuration.

Note: You can initiate a second reconfiguration while a dynamic reconfiguration is in progress, but you cannot synchronize it. The presence of an SCD on any cluster node acts as a lock, preventing the initiation of a new dynamic reconfiguration.

Resource Group Management

You can use the *Resource Group Management (**clRGmove**)* utility to move resource groups to other cluster nodes (or sites) or take them online or offline without stopping cluster services. This gives you the flexibility for managing resource groups and their applications. You can also use this utility to free the node of any resource groups to perform system maintenance on a particular cluster node.

In HACMP 5.4.1, the HACMP Resource Group Management Utility, **clRGmove**, is significantly improved, making it easier for you to move the resource groups around for cluster management. In addition, you also have a clear and easy way to understand the consequences of manually moving resource groups. For example, you can clearly predict whether the groups will stay on the nodes to which they were moved.

HACMP follows this simple principle: In all cases, when you move the resource groups, they stay on the nodes until you move them again. (Note that HACMP moves them around when it needs to recover them).

If you want to check whether the resource group is currently hosted on the highest priority node that is now available, HACMP presents intelligent picklist choices for nodes and sites. For instance, if the group is currently hosted on one node, and HACMP *finds another node that has a higher priority*, then the SMIT picklist with destination nodes indicates which node *has* a higher priority. This way, you can always choose to move this group to this node.

When you move groups to other nodes, these rules apply:

- For resource groups with a fallback policy of **Never Fallback**, moving a group will have no effect on the behavior of that group during the future cluster events. The same is true for resource groups with the site fallback policy **Online on Either Site**.
- For resource groups with a fallback policy other than **Never Fallback** (and **Prefer Primary Site,** if sites are defined), moving a group will result in a destination node becoming an “acting” highest priority node until you move it again, in which case, again, the node becomes an “acting” highest priority node.

  One important consideration for this behavior has to do with resource groups that have **Fallback to Highest Priority Node** policy (or **Prefer Primary Site** policy). When you move such a resource group to a node other than its highest priority node (or Primary site), the node to which it was moved becomes its temporarily “preferred” node while *not* being its highest priority node (as configured). Such groups stay on the nodes to which they were moved until you move them again. The groups also fall back to these nodes (or sites).

For more information about resource group management, see the following:

- Overview in Chapter 7: HACMP Configuration Process and Facilities
- Chapter on planning resource groups in the *Planning Guide*
- Chapter on changing resources and resource groups in the *Administration Guide* for complete information and instructions on performing resource group management through SMIT.

User-Requested Resource Group Management vs. Automatic Resource Group Management

In general, to keep applications highly available, HACMP automatically manages (and sometimes moves) resource groups and applications included in them. For instance, when it is necessary to recover a resource group, HACMP may attempt to recover it automatically on another node during fallover or fallback operations. While moving a group, HACMP adheres to the resource group policies that you specified, and other settings (for instance, rather than automatically recovering a failed resource group on another node, you can tell HACMP to just notify you of the group's failure).

When you request HACMP to perform resource group management, it uses the **clRGmove** utility, which moves resource groups by calling an **rg_move** event.

Note: When troubleshooting log files, it is important to distinguish between an **rg_move** event that in some cases is triggered automatically by HACMP, and an **rg_move** event that occurs when you request HACMP to manage resource groups for you. To identify the causes of operations performed on the resource groups in the cluster, look for the command output in SMIT and for information in the **hacmp.out** file.

Resource Group Management Operations

Use resource group management to:

- Move a resource group from the node on one site to the node on another site.
- Move a resource group from one node to another.

  In a working cluster, temporarily move a non-concurrent resource group from a node it currently resides on to any destination node. Resource groups that you move continue to behave consistently with the way you configured them, that is, they follow the startup, fallover and fallback policies specified for them. The SMIT user interface lets you clearly specify and predict the resource group's behavior, if you decide to move it to another node.

  If you use SMIT to move a resource group to another node, it remains on its new destination node until you manually move it again. Note that HACMP may need to move it during a fallover.
- Move the resource group back to the node that was originally its highest priority.

  The resource group may or may *not* have a fallback policy. If a resource group has a fallback policy of **Fallback to Highest Priority Node**, after you move it, the group assumes that the "new" node is now its preferred temporary location, and falls back to this node. To change this behavior, you can always move the group back to the node that was originally its highest priority node.

  Similarly, if you have a resource group that has a fallback policy **Never Fallback**, once you move this resource group, it will *not* move back to the node from which it was moved but will remain on its new destination node, until you move it again to another node. This way, you can be assured that the group always follows the **Never Fallback** policy that you specified for it.
- Bring a resource group online or offline on one or all nodes in the cluster. See the *Administration Guide* for detailed information on what kinds of online and offline operations you can perform on concurrent and non-concurrent resource groups.

Cluster Single Point of Control (C-SPOC)

With the C-SPOC utility, you can make changes to the whole cluster from a single cluster node. Instead of performing administrative tasks on each cluster node, you can use the SMIT interface to issue a C-SPOC command once, on a single node, and the change is propagated across all cluster nodes.

For more information about C-SPOC, see the section HACMP System Management with C-SPOC in Chapter 7: HACMP Configuration Process and Facilities.

Dynamic Adapter Swap

The dynamic adapter swap functionality lets you swap the IP address of an active network interface card (NIC) with the IP address of a user-specified active, available “backup” network interface card on the *same* node and network. Cluster services do *not* have to be stopped to perform the swap.

This feature can be used to move an IP address off a network interface card that is behaving erratically, to another NIC without shutting down the node. It can also be used if a hot pluggable NIC is being replaced on the node. Hot pluggable NICs can be physically removed and replaced without powering off the node. When the (hot pluggable) NIC to be replaced is pulled from the node, HACMP makes the NIC unavailable as a backup.

You can configure adapter swap using SMIT. The service IP address is moved from its current NIC to a user-specified NIC. The service IP address then becomes an available “backup” address. When the new card is placed in the node, the NIC is incorporated into the cluster as an available “backup” again. You can then swap the IP address from the backup NIC to the original NIC.

Note: This type of dynamic adapter swap can only be performed within a single node. You cannot swap the IP address with the IP address on a different node with this functionality. To move a service IP address to another node, move its resource group using the Resource Group Management utility.

Note: The dynamic adapter swap feature is *not* supported on the SP switch network.

Automatic Verification and Synchronization

Automatic verification and synchronization minimizes downtime when you add a node to your cluster. This process runs prior to starting cluster services and checks to make sure that nodes joining a cluster are synchronized appropriately. This process checks nodes entering either active or inactive configurations.

Automatic verification and synchronization ensures that typical configuration inconsistencies are corrected as follows:

- RSCT numbers are consistent across the cluster
- IP addresses are configured on the network interfaces that RSCT expects
- Shared volume groups are *not* set to be automatically varied on
- File Systems are *not* set to be automatically mounted.

If any additional configuration errors are found, cluster services are *not* started on the node, and detailed error messages enable you to resolve the inconsistencies.

For more information about automatic verification and synchronization, see Chapter 7: Verifying and Synchronizing a Cluster Configuration in the *Administration Guide*.

Minimizing Unscheduled Downtime

Another important goal with HACMP is to minimize unscheduled downtime in response to unplanned cluster component failures. The HACMP software provides the following features to minimize unscheduled downtime:

- *Fast recovery* to speed up the fallover in large clusters
- A *delayed fallback timer* to allow a custom resource group to fall back at a specified time
- *IPAT via IP Aliases* to speed up the processing during recovery of service IP labels
- *Automatic recovery of resource groups* that are in the ERROR state, whenever a cluster node comes up. For more information, see the following section.

Recovering Resource Groups on Node Startup

Prior to HACMP 5.2, when a node joined the cluster, it did *not* acquire any resource groups that had previously gone into an ERROR state on any other node. Such resource groups remained in the ERROR state and required use of the Resource Group Migration utility, **clRGmove**, to manually bring them back online.

Starting with HACMP 5.2, the Cluster Manager tries to bring the resource groups that are currently in the ERROR state into the online (active) state on the joining node. This further increases the chances of bringing the applications back online. When a node starts up, if a resource group is in the ERROR state on any node in the cluster, this node attempts to acquire the resource group. Note that the node must be included in the nodelist for the resource group.

The resource group recovery on node startup is different for non-concurrent and concurrent resource groups:

- If the starting node fails to activate a *non-concurrent resource group* that is in the ERROR state, the resource group continues to fall over to another node in the nodelist, if a node is available. The fallover action continues until all available nodes in the nodelist have been tried.
- If the starting node fails to activate a *concurrent resource group* that is in the ERROR state on the node, the concurrent resource group is left in the ERROR state on that node. Note that the resource group might still remain online on other nodes.

Fast Recovery

The HACMP fast recovery feature speeds up fallover in large clusters.

Fast recovery lets you select a file systems consistency check and a file systems recovery method:

- If you configure a file system to use a consistency check and a recovery method, it saves time by running **logredo** rather than **fsck** on each file system. If the subsequent **mount** fails, then it runs a full **fsck**.

If a file system suffers damage in a failure but can still be mounted, **logredo** may *not* succeed in fixing the damage, producing an error during data access.

- In addition, it saves time by acquiring, releasing, and falling over all resource groups and file systems in parallel, rather than serially.

  Do *not* set the system to run these commands in parallel if you have shared, nested file systems. These must be recovered sequentially. (Note that the cluster verification utility does *not* report file system and fast recovery inconsistencies.)

  The **varyonvg** and **varyoffvg** commands always run on volume groups in parallel, regardless of the setting of the recovery method.

Delayed Fallback Timer for Resource Groups

The Delayed Fallback Timer lets a resource group fall back to the higher priority node at a time that you specify. The resource group that has a delayed fallback timer configured and that currently resides on a non-home node falls back to the higher priority node at the recurring time (daily, weekly, monthly or yearly), or on a specified date.

For more information on the delayed fallback timer, see the *Planning Guide.*

Minimizing Takeover Time: Fast Disk Takeover

In the case of a cluster failure, enhanced concurrent volume groups are taken over faster than in previous releases of HACMP due to the improved disk takeover mechanism.

HACMP automatically detects enhanced concurrent volume groups and ensures that the faster option for volume group takeover is launched in the event of a node failure, if:

- You have installed AIX 5.2 or 5.3 and HACMP.
- You include in your non-concurrent resource groups the enhanced concurrent mode volume groups (or convert the existing volume groups to enhanced concurrent volume groups).

This functionality is especially useful for fallover of volume groups made up of a large number of disks.

During fast disk takeover, HACMP skips the extra processing needed to break the disk reserves, or update and synchronize the LVM information by running lazy update. As a result, the disk takeover mechanism of HACMP used for enhanced concurrent volume groups is faster than disk takeover used for standard volume groups included in non-concurrent resource groups.

Maximizing Disaster Recovery

HACMP can be an integral part of a comprehensive disaster recovery plan for your enterprise. Three possible ways to distribute backup copies of data to different sites, for possible disaster recovery operations, include:

- HACMP/XD for Geographic LVM (GLVM)
- HACMP/XD for Metro Mirror (synchronous PPRC with ESS and DS systems)
- HACMP/XD for HAGEO (IP Mirroring)

- Cross-Site LVM Mirroring.

For more information on the disaster recovery solutions included in HACMP/XD, see About This Guide for the documentation and the location of Release Notes.

Cross-Site LVM Mirroring

Starting with HACMP 5.2, you can set up disks located at two different sites for remote LVM mirroring, using a Storage Area Network (SAN), for example. Cross-site LVM mirroring replicates data between the disk subsystem at each site for disaster recovery.

A SAN is a high-speed network that allows the establishment of direct connections between storage devices and processors (servers) within the distance supported by Fibre Channel. Thus, two or more servers (nodes) located at different sites can access the same physical disks, which can be separated by some distance as well, through the common SAN. The disks can be combined into a volume group via the AIX Logical Volume Manager, and this volume group can be imported to the nodes located at different sites. The logical volumes in this volume group can have up to three mirrors. Thus, you can set up at least one mirror at each site. The information stored on this logical volume is kept highly available, and in case of certain failures, the remote mirror at another site will still have the latest information, so the operations can be continued on the other site.

HACMP automatically synchronizes mirrors after a disk or node failure and subsequent reintegration. HACMP handles the automatic mirror synchronization even if one of the disks is in the PVREMOVED or PVMISSING state. Automatic synchronization is *not* possible for all cases, but you can use C-SPOC to synchronize the data manually from the surviving mirrors to stale mirrors after a disk or site failure and subsequent reintegration.

Cluster Events

This section describes how the HACMP software responds to changes in a cluster to maintain high availability.

The HACMP cluster software monitors all the components that make up the highly available application including disks, network interfaces, nodes and the applications themselves. The Cluster Manager uses different methods for monitoring different resources:

- RSCT subsystem is responsible for monitoring networks and nodes.
- The AIX LVM subsystem produces error notifications for volume group quorum loss.
- The Cluster Manager itself dispatches application monitors.

An HACMP cluster environment is event-driven. An event is a change of status within a cluster that the Cluster Manager recognizes and processes. A cluster event can be triggered by a change affecting a network interface card, network, or node, or by the cluster reconfiguration process exceeding its time limit. When the Cluster Manager detects a change in cluster status, it executes a script designated to handle the event and its subevents.

Note: The logic of cluster events is described here for completeness. No manual intervention is required to ensure that HACMP carries out cluster events correctly.

The following examples show some events the Cluster Manager recognizes:

- **node_up** and **node_up_complete** events (a node joining the cluster)
- **node_down** and **node_down_complete** events (a node leaving the cluster)
- Local or global **network_down** event (a network has failed)
- **network_up** event (a network has connected)
- **swap_adapter** event (a network adapter failed and a new one has taken its place)
- Dynamic reconfiguration events.

When a cluster event occurs, the Cluster Manager runs the corresponding event script for that event. As the event script is being processed, a series of subevent scripts may be executed. The HACMP software provides a script for each event and subevent. The default scripts are located in the **/usr/es/sbin/cluster/events** directory.

By default, the Cluster Manager calls the corresponding event script supplied with the HACMP software for a specific event. You can specify additional processing to customize event handling for your site if needed. For more information, see the section Customizing Event Processing.

Processing Cluster Events

The two primary cluster events that HACMP software handles are fallover and reintegration:

- *Fallover* refers to the actions taken by the HACMP software when a cluster component fails or a node leaves the cluster.
- *Reintegration* refers to the actions that occur within the cluster when a component that had previously left the cluster returns to the cluster.

Event scripts control both types of actions. During event script processing, cluster-aware application programs see the state of the cluster as unstable.

Fallover

A fallover occurs when a resource group moves from its home node to another node because its home node leaves the cluster.

Nodes leave the cluster either by a planned transition (a node shutdown or stopping cluster services on a node), or by failure. In the former case, the Cluster Manager controls the release of resources held by the exiting node and the acquisition of these resources by nodes still active in the cluster. When necessary, you can override the release and acquisition of resources (for example, to perform system maintenance). You can also postpone the acquisition of the resources by integrating nodes (by setting the delayed fallback timer for custom resource groups).

Node failure begins when a node monitoring a neighboring node ceases to receive keepalive traffic for a defined period of time. If the other cluster nodes agree that the failure is a node failure, the failing node is removed from the cluster and its resources are taken over by the active nodes configured to do so.

If other components, such as a network interface card, fail, the Cluster Manager runs an event script to switch network traffic to a backup network interface card (if present).

Reintegration

A reintegration, or a fallback occurs when a resource group moves to a node that has just joined the cluster.

When a node joins a running cluster, the cluster becomes temporarily unstable. The member nodes coordinate the beginning of the join process and then run event scripts to release any resources the joining node is configured to take over. The joining node then runs an event script to take over these resources. Finally, the joining node becomes a member of the cluster. At this point, the cluster is stable again.

Emulating Cluster Events

HACMP provides an emulation utility to test the effects of running a particular event without modifying the cluster state. The emulation runs on every active cluster node, and the output is stored in an output file on the node from which the emulation was launched.

For more information on the Event Emulator utility, see Chapter 7: HACMP Configuration Process and Facilities.

Customizing Event Processing

The HACMP software has an event customization facility you can use to tailor event processing. The Cluster Manager's ability to recognize a specific series of events and subevents permits a very flexible customization scheme. Customizing event processing allows you to provide the most efficient path to critical resources should a failure occur.

You can define multiple pre- and post-events for a list of events that appears in the picklist in the **Change/Show Pre-Defined HACMP Events** SMIT panel.

Customization for an event could include notification to the system administrator before and after the event is processed, as well as user-defined commands or scripts before and after the event processing, as shown in the list:

- Notification to system administrator of event to be processed
- Pre-event script or command
- HACMP for AIX event script
- Post-event script or command
- Notification to system administrator event processing is complete.

Use this facility for the following types of customization:

- Pre- and post-event processing
- Event notification
- Event recovery and retry.

Note: In HACMP, the event customization information stored in the HACMP configuration database is synchronized across all cluster nodes when the cluster resources are synchronized. Thus, pre- and post-notification, and recovery event script names must be the same on all nodes, although the actual processing done by these scripts can be different.

Cluster verification includes a function to monitor cluster configuration automatically by means of a new event called **cluster_notify**. You can use this event to configure an HACMP remote notification method (numeric or alphanumeric page, or text messaging) to send out a message if errors in cluster configuration are found. The output of this event is also logged in **hacmp.out** on each cluster node that is running cluster services.

You may also send email notification to cell phones through the event notification scripts; however, using remote notification has advantages. If you are the person responsible for responding to event notifications changes, you must manually change the address in each event notification script. Define for each person remote notification methods that contain all the events and nodes so you can switch the notification methods as a unit when responders change.

Defining New Events

In HACMP, it is possible to define new events as well as to tailor the existing ones.

Pre- and Post-Event Processing

To tailor event processing to your environment, specify commands or user-defined scripts that execute before and after a specific event is generated by the Cluster Manager. For pre-processing, for example, you may want to send a message to specific users, informing them to stand by while a certain event occurs. For post-processing, you may want to disable login for a specific group of users if a particular network fails.

Event Notification

You can specify a command or user-defined script that provides notification (for example, mail) that an event is about to happen and that an event has just occurred, along with the success or failure of the event. You can also define a notification method through the SMIT interface to issue a customized remote notification method in response to a cluster event.

Event Recovery and Retry

You can specify a command that attempts to recover from an event command failure. If the retry count is greater than zero and the recovery command succeeds, the event script command is run again. You can also specify the number of times to attempt to execute the recovery command.

Customizing Event Duration

HACMP software issues a system warning each time a cluster event takes more time to complete than a specified timeout period.

Using the SMIT interface, you can customize the time period allowed for a cluster event to complete before HACMP issues a system warning for it.

Chapter 6: HACMP Cluster Configurations

This chapter provides examples of the types of cluster configurations supported by the HACMP software.

Sample Cluster Configurations

The following sample cluster configurations are discussed in this chapter

- Standby Configurations—These are the traditional redundant hardware configurations where one or more standby nodes stand idle, waiting for a server node to leave the cluster.
- Takeover Configurations—In these configuration, *all cluster nodes do useful work,* processing part of the cluster's workload. There are no standby nodes. Takeover configurations use hardware resources more efficiently than standby configurations since there is no idle processor. Performance can degrade after node detachment, however, since the load on remaining nodes increases.

  Takeover configurations that use *concurrent access* use hardware efficiently and also minimize service interruption during fallover because there is no need for the takeover node to acquire the resources released by the failed node—the takeover node already shares ownership of the resources.
- Cluster Configurations with Multi-Tiered Applications—In these configurations, one application depends on another application. These cluster configurations use dependent resource groups.
- Cluster Configurations with Resource Group Location Dependencies—In these configurations, related applications are configured to always stay on the same node or to always stay on a different node from other applications. These cluster configurations use location dependent resource groups.
- Cross-Site LVM Mirror Configurations for Disaster Recovery—In these geographically dispersed configurations, LVM mirroring replicates data between the disk subsystems at each of two sites for disaster recovery.
- Cluster Configurations with Dynamic LPARs—In these configurations, HACMP clusters use LPARs as cluster nodes. This lets you perform routine system upgrades through the dynamic allocation of system resources and redistribute CPU and memory resources to manage the application workload.

This list is by no means an exhaustive catalog of the possible configurations you can define using the HACMP software. Rather, use them as a starting point for thinking about the cluster configuration best suited to your environment.

Standby Configurations

The standby configuration is a traditional redundant hardware configuration, where one or more standby nodes stand idle, waiting for a server node to leave the cluster.

The sample standby configurations discussed in this chapter show how the configuration is defined for these types of resource groups:

- Standby Configurations: Example 1 shows resource groups with the Online on Home Node Only startup policy, Fallover to Next Priority Node in the List fallover policy and Fallback to Higher Priority Node in the List fallback policy.
- Standby Configurations: Example 2 shows resource groups with the startup Online Using Distribution Policy (network or node), fallover policy Next Priority Node in the List, and fallback policy Never Fallback.

Concurrent resource groups require all nodes to have simultaneous access to the resource group and cannot be used in a standby configuration.

Standby Configurations: Example 1

The following figure shows a two-node standby configuration that uses resource groups with these policies:

- Startup policy: Online on Home Node Only
- Fallover policy: Fallover to Next Priority Node in the List
- Fallback policy: Fallback to Higher Priority Node in the List.

In the figure, a lower number indicates a higher priority:

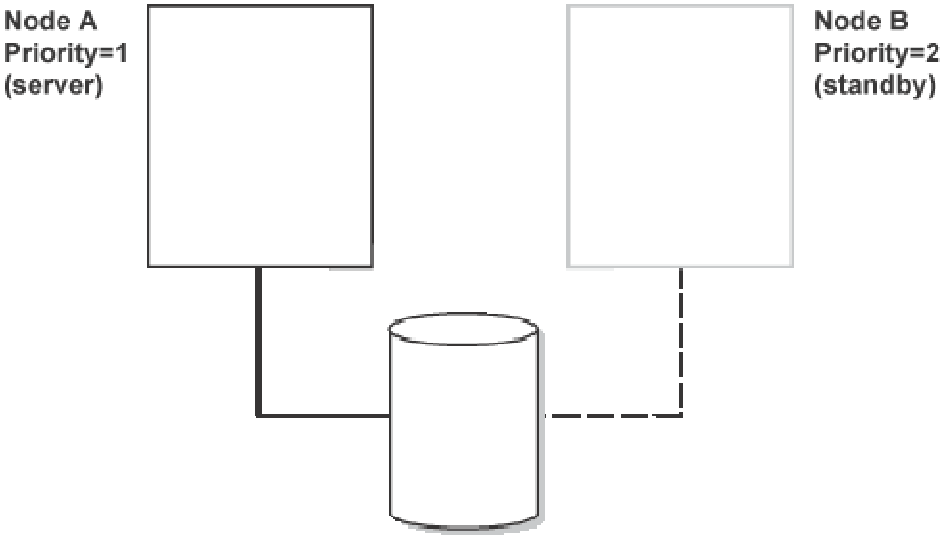

Figure 17. One-for-One Standby Configuration where IP Label Returns to the Home Node

In this setup, the cluster resources are defined as part of a single resource group. A nodelist is then defined as consisting of two nodes. The first node, Node A, is assigned a takeover (ownership) priority of 1. The second node, Node B, is assigned a takeover priority of 2.

At cluster startup, Node A (which has a priority of 1) assumes ownership of the resource group. Node A is the "server" node. Node B (which has a priority of 2) stands idle, ready should Node A fail or leave the cluster. Node B is, in effect, the "standby".

If the server node leaves the cluster, the standby node assumes control of the resource groups owned by the server, starts the highly available applications, and services clients. The standby node remains active until the node with the higher takeover priority rejoins the cluster. At that point, the standby node releases the resource groups it has taken over, and the server node reclaims them. The standby node then returns to an idle state.

Extending Standby Configurations from Example 1

The standby configuration from the previously described example can be easily extended to larger clusters. The advantage of this configuration is that it makes better use of the hardware. The disadvantage is that the cluster can suffer severe performance degradation if more than one server node leaves the cluster.

The following figure illustrates a three-node standby configuration using the resource groups with these policies:

- Startup policy: Online on Home Node Only
- Fallover policy: Fallover to Next Priority Node in the List
- Fallback policy: Fallback to Higher Priority Node in the List.

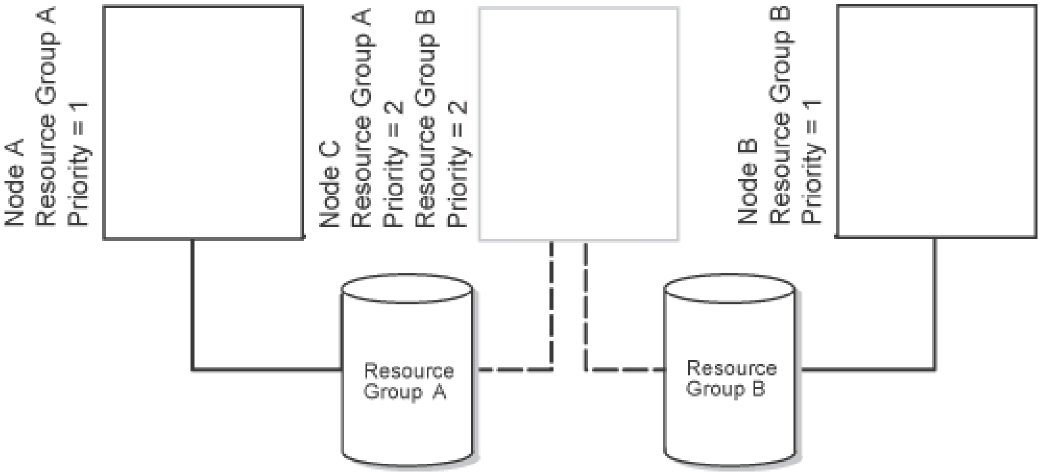

Figure 18. One-for-Two Standby Configuration with Three Resource Groups

In this configuration, two separate resource groups (A and B) and a separate nodelist for each resource group exist. The nodelist for Resource Group A consists of Node A and Node C. Node A has a takeover priority of 1, while Node C has a takeover priority of 2. The nodelist for Resource Group B consists of Node B and Node C. Node B has a takeover priority of 1; Node C again has a takeover priority of 2. (A resource group can be owned by only a single node in a non-concurrent configuration.)

Since each resource group has a different node at the head of its nodelist, the cluster's workload is divided, or partitioned, between these two resource groups. Both resource groups, however, have the same node as the standby in their nodelists. If either server node leaves the cluster, the standby node assumes control of that server node's resource group and functions as the departed node.

In this example, the standby node has three network interfaces (not shown) and separate physical connections to each server node's external disk. Therefore, the standby node can, if necessary, take over for both server nodes concurrently. The cluster's performance, however, would most likely degrade while the standby node was functioning as both server nodes.

Standby Configurations: Example 2

In the following standby configuration, the resource groups have these policies:

- Startup policy: Online Using Distribution Policy (network-based or node-based)
- Fallover policy: Next Priority Node in the List
- Fallback policy: Never Fallback.

This configuration differs from a standby configuration in which the ownership priority of resource groups is not fixed. Rather, the resource group is associated with an IP address that can rotate among nodes. This makes the roles of server and standby fluid, changing over time.

The following figure illustrates the one-for-one standby configuration that is described in this section:

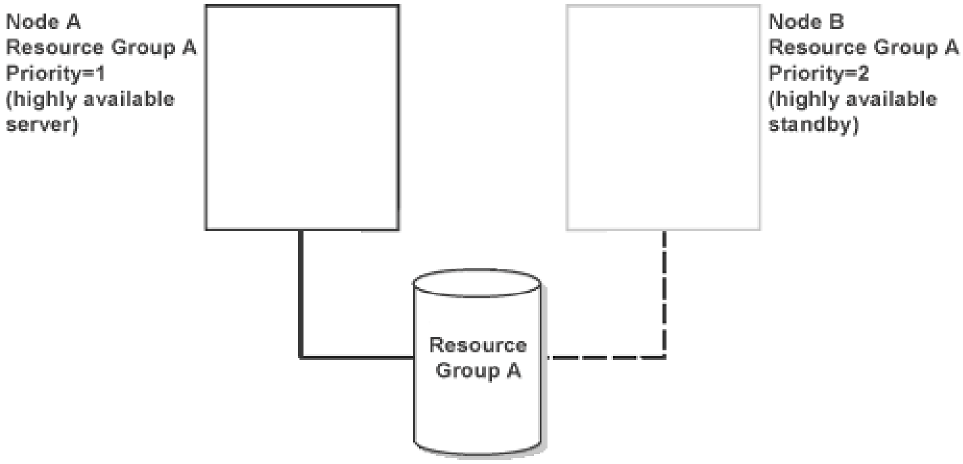

Figure 19. One-for-One Standby Configuration with Resource Groups where IP Label Rotates

At system startup, the resource group attaches to the node that claims the shared IP address. This node "owns" the resource group for as long as it remains in the cluster. If this node leaves the cluster, the peer node assumes the shared IP address and claims ownership of that resource group. Now, the peer node "owns" the resource group for as long as it remains in the cluster.

When the node that initially claimed the resource group rejoins the cluster, it does not take the resource group back. Rather, it remains idle for as long as the node currently bound to the shared IP address is active in the cluster. Only if the peer node leaves the cluster does the node that initially "owned" the resource group claim it once again. Thus, ownership of resources rotates between nodes.

Extending Standby Configurations From Example 2

As with the first example of the standby configuration, configurations from Example 2 can be easily extended to larger clusters. For example, in a one-for-two standby configuration from Example 2, the cluster could have two separate resource groups, each of which includes a distinct shared IP address.

At cluster startup, the first two nodes each claim a shared IP address and assume ownership of the resource group associated with that shared IP address. The third node remains idle. If an active node leaves the cluster, the idle node claims that shared IP address and takes control of that resource group.

Takeover Configurations

All nodes in a takeover configuration process part of the cluster's workload. There are no standby nodes. Takeover configurations use hardware resources more efficiently than standby configurations since there is no idle processor. Performance degrades after node detachment, however, since the load on remaining nodes increases.

One-Sided Takeover

The following figure illustrates a two-node, one-sided takeover configuration. In the figure, a lower number indicates a higher priority.

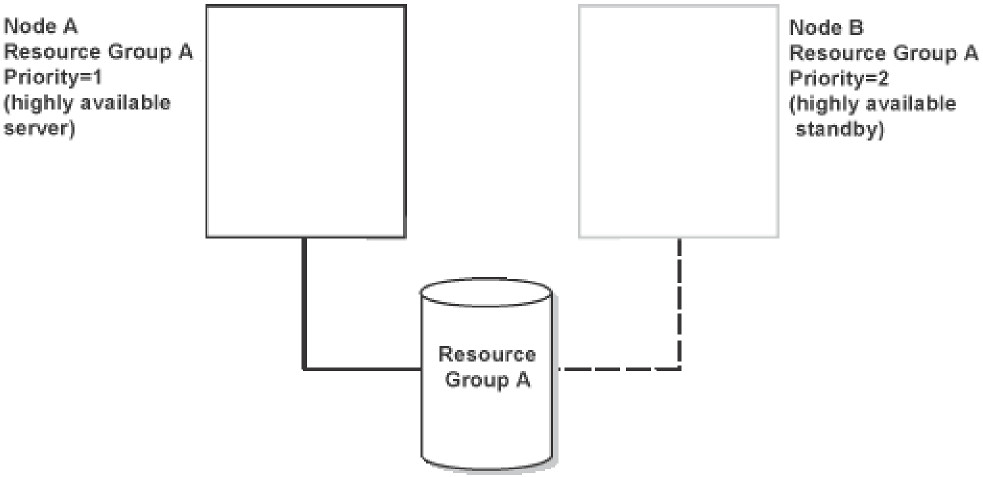

Figure 20. One-sided Takeover Configuration with Resource Groups in Which IP Label Returns to the Home Node

This configuration has two nodes actively processing work, but only one node providing highly available services to cluster clients. That is, although there are two sets of resources within the cluster (for example, two server applications that handle client requests), only one set of resources needs to be highly available. This set of resources is defined as an HACMP resource group and has a nodelist that includes both nodes. The second set of resources is not defined as a resource group and, therefore, is *not* highly available.

At cluster startup, Node A (which has a priority of 1) assumes ownership of Resource Group A. Node A, in effect, "owns" Resource Group A. Node B (which has a priority of 2 for Resource Group A) processes its own workload independently of this resource group.

If Node A leaves the cluster, Node B takes control of the shared resources. When Node A rejoins the cluster, Node B releases the shared resources.

If Node B leaves the cluster, however, Node A does *not* take over any of its resources, since Node B's resources are *not* defined as part of a highly available resource group in whose chain this node participates.

This configuration is appropriate when a single node is able to run all the critical applications that need to be highly available to cluster clients.

Mutual Takeover

The mutual takeover for non-concurrent access configuration has multiple nodes, each of which provides distinct highly available services to cluster clients. For example, each node might run its own instance of a database and access its own disk.

Furthermore, each node has takeover capacity. If a node leaves the cluster, a surviving node takes over the resource groups owned by the departed node.

The mutual takeover for non-concurrent access configuration is appropriate when each node in the cluster is running critical applications that need to be highly available and when each processor is able to handle the load of more than one node.

The following figure illustrates a two-node mutual takeover configuration for non-concurrent access. In the figure, a lower number indicates a higher priority.

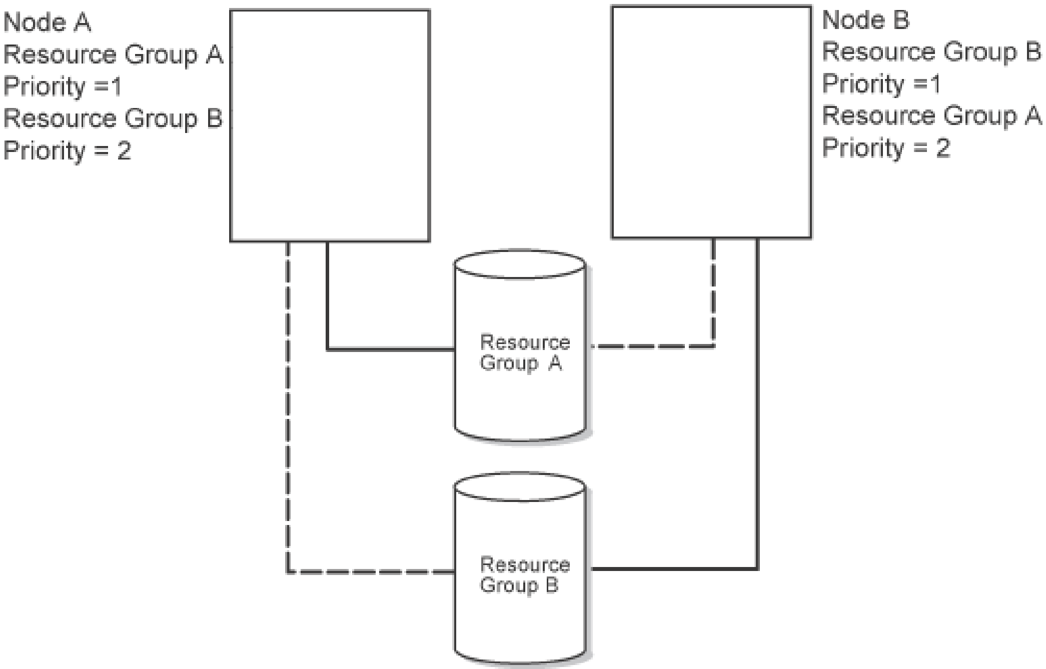

Figure 21. Mutual Takeover Configuration for Non-Concurrent Access

The key feature of this configuration is that the cluster's workload is divided, or partitioned, between the nodes. Two resource groups exist, in addition to a separate resource chain for each resource group. The nodes that participate in the resource chains are the same. It is the differing priorities within the chains that designate this configuration as mutual takeover.

The chains for both resource groups consist of Node A and Node B. For Resource Group A, Node A has a takeover priority of 1 and Node B has a takeover priority of 2. For Resource Group B, the takeover priorities are reversed. Here, Node B has a takeover priority of 1 and Node A has a takeover priority of 2.

At cluster startup, Node A assumes ownership of the Resource Group A, while Node B assumes ownership of Resource Group B.

If either node leaves the cluster, its peer node takes control of the departed node's resource group. When the "owner" node for that resource group rejoins the cluster, the takeover node relinquishes the associated resources; they are reacquired by the higher-priority, reintegrating node.

Two-Node Mutual Takeover Configuration for Concurrent Access

The following figure illustrates a two-node mutual takeover configuration for concurrent access:

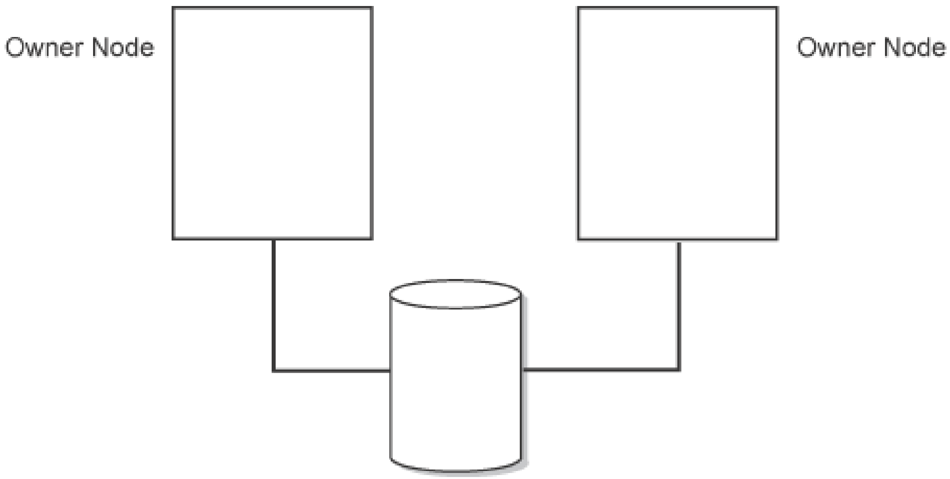

Figure 22. Two-Node Mutual Takeover Configuration for Concurrent Access

In this configuration, both nodes have simultaneous access to the shared disks and own the same disk resources. There is no “takeover” of shared disks if a node leaves the cluster, since the peer node already has the shared volume group varied on.

In this example, both nodes are running an instance of a server application that accesses the database on the shared disk. The application’s proprietary locking model is used to arbitrate application requests for disk resources.

Running multiple instances of the same server application allows the cluster to distribute the processing load. As the load increases, additional nodes can be added to further distribute the load.

Eight-Node Mutual Takeover Configuration for Concurrent Access

The following figure illustrates an eight-node mutual takeover configuration for concurrent access:

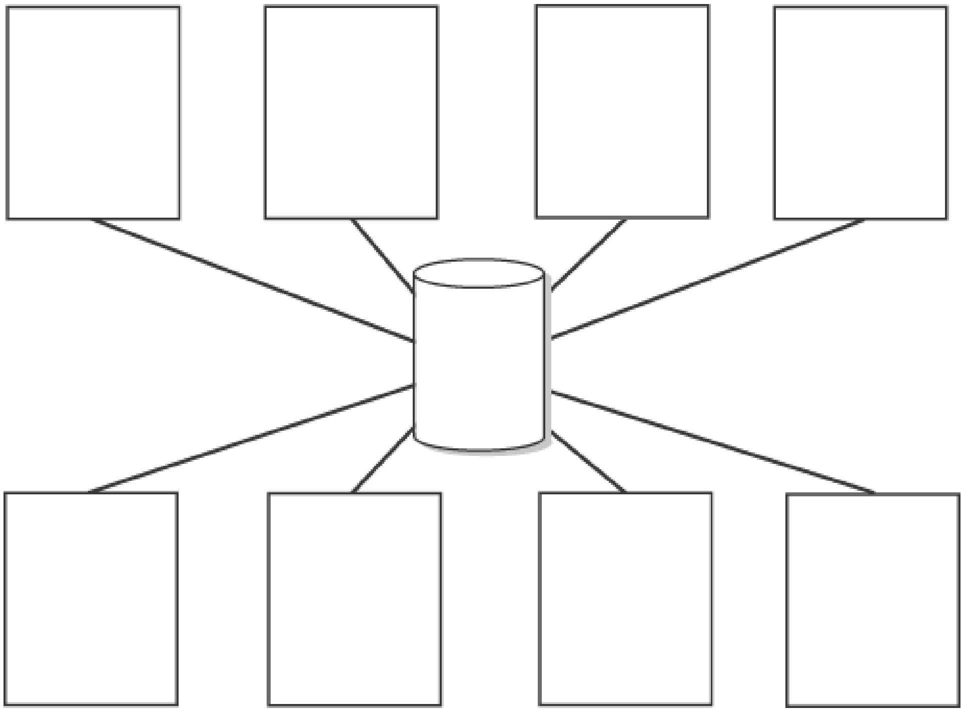

Figure 23. Eight-Node Mutual Takeover for Concurrent Access

In this configuration, as in the previous configuration, all nodes have simultaneous—but *not* concurrent—access to the shared disks and own the same disk resources. Here, however, each node is running a different server application. Clients query a specific application at a specific IP address. Therefore, each application server and its associated IP address must be defined as part of a non-concurrent resource group, and all nodes that are potential owners of that resource group must be included in a corresponding nodelist.

Concurrent access resource groups are supported in clusters with up to 32 nodes in HACMP.

Cluster Configurations with Multi-Tiered Applications

A typical cluster configuration that could utilize parent/child dependent resource groups is the environment in which an application such as WebSphere depends on another application such as DB2.

Note: It is important to distinguish the application server, such as WebSphere, from the HACMP application server that you configure in HACMP by specifying the application server start and stop scripts.

In order to satisfy business requirements, a cluster-wide parent/child dependency must be defined between two or more resource groups. The following figure illustrates the business scenario that utilizes dependencies between applications:

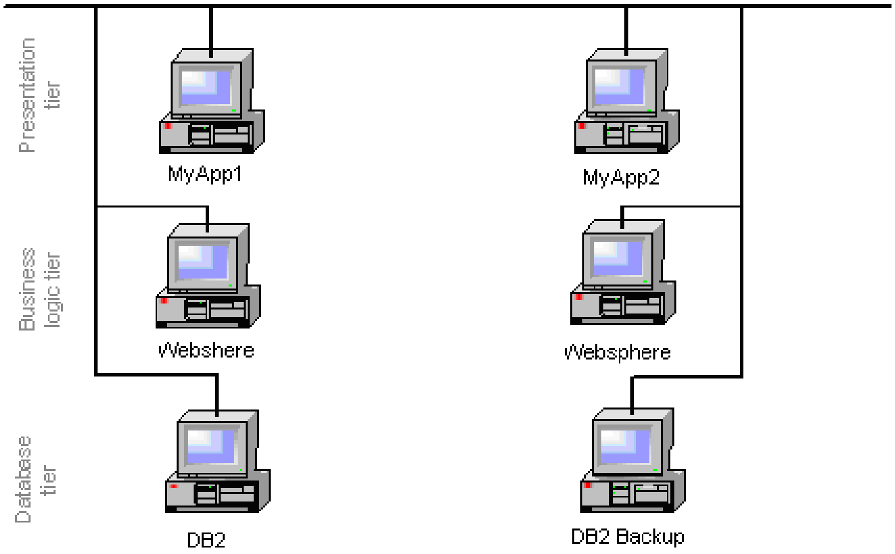

Figure 24. Typical Multi-Tier Cluster Environment with Dependencies between Applications

Multi-Tiered Applications

Business configurations that use layered, or multi-tiered applications can also utilize dependent resource groups. For example, the back end database must be online before the application server. In this case, if the database goes down and is moved to a different node, the resource group containing the application server would have to be brought down and back up on any node in the cluster.

Environments such as SAP require applications to be cycled (stopped and restarted) anytime a database fails. An environment like SAP provides many application services, and the individual application components often need to be controlled in a specific order.

Another area where establishing interdependencies between resource groups proves useful is when system services are required to support application environments. Services such as **cron** jobs for pruning log files or initiating backups need to move from node to node along with an application, but are typically *not* initiated until the application is established. These services can be built into application server start and stop scripts. When greater granularity is needed, they can be controlled through pre- and post- event processing. Parent/child dependent resource groups allow an easier way to configure system services to be dependent upon applications they serve.

For an overview of dependent resource groups, see the section Resource Group Dependencies in Chapter 3: HACMP Resources and Resource Groups.

Cluster Configurations with Resource Group Location Dependencies

With HACMP 5.4.1, you can configure the cluster so that certain applications stay on the same node, on the same site, or on different nodes *not* only at startup, but during fallover and fallback events. To do this, you configure the selected resource groups as part of a location dependency set.

Publishing Model with Same Node and Different Nodes Dependencies

Consider this example: The XYZ Publishing company follows a business continuity model that involves separating the different platforms used to develop the web content. XYZ uses location dependency policies to keep some resource groups strictly on separate nodes and others together on the same node.

The Production database (PDB) and Production application (PApp) are hosted on the same node to facilitate maintenance (and perhaps the highest priority node for these resource groups has the most memory or faster processor). It also makes sense to set up a parent/child relation between them, since the application depends on the database. The database must be online for the application to function.The same conditions are true for the System Database (SDB) and the System application (Sapp) and for the QA Database (QADB) and the QA application (QAapp).

Since keeping the production database and application running is the highest priority, it makes sense to configure the cluster so that the three database resource groups stay on different nodes (make them an Online On Different Nodes dependency set), and assign the PDB resource group with the **high** priority. The SDB is the **Intermediate** priority and the QADB is the **low** priority.

The databases and their related applications are each configured to belong to an Online On Same Node dependency set.

HACMP handles these groups somewhat differently depending on how you configure startup, fallover, and fallback policies. It makes sense to have the participating nodelists differ for each database and application set to facilitate keeping these resource groups on the preferred nodes.

The figure below shows the basic configuration of the three nodes and six resource groups.

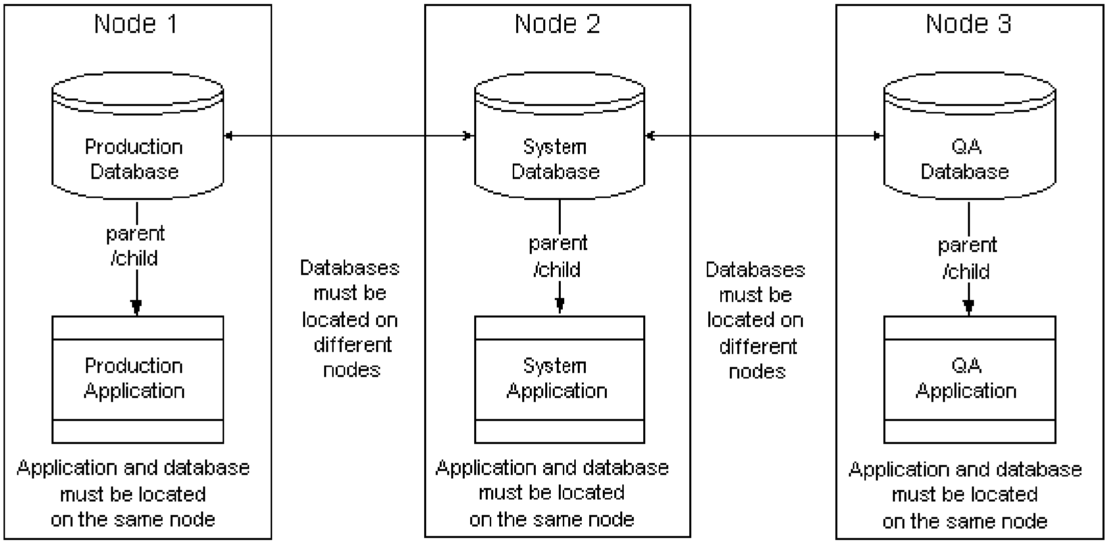
Node 1
Production Database
parent /child
Production Application
Application and database must be located on the same node
Databases must be located on different nodes
Node 2
System Database
parent /child
System Application
Application and database must be located on the same node
Databases must be located on different nodes
Node 3
QA Database
parent /child
QA Application
Application and database must be located on the same node

Figure 25. Publishing Model with Parent/Child and Location Dependencies

Resource Group Policies

For the sake of illustration of this case, all six resource groups might have the following behavioral policies:

- Startup Policy: Online On First Available Node
- Fallover Policy: Fallover to Next Priority Node
- Fallback Policy: Never Fallback

Participating Nodes	Location Dependency	Parent/Child Dependency
• PApp: 1, 2, 3 • PDB: 1, 2, 3 • SApp: 2, 3 • SDB: 2, 3 • QAApp: 3 • QADB: 3	Online On The Same Node Dependent Groups: • PApp with PDB • SApp with SDB • QAApp with QADB Online On Different Nodes Dependent set: [PDB SDB QADB] Priority: PDB > SDB > QADB	• PApp (child) depends on PDB (parent) • SApp (child) depends on SDB (parent) • QAApp (child) depends on QADB (parent)

See Appendix B in the *Administration Guide* for more examples of location dependencies and use cases.

Cross-Site LVM Mirror Configurations for Disaster Recovery

In HACMP 5.2 and up, you can set up disks located at two different sites for remote LVM mirroring, using a Storage Area Network (SAN). A SAN is a high-speed network that allows the establishment of direct connections between storage devices and processors (servers) within the distance supported by Fibre Channel. Thus, two or more distantly separated servers (nodes) located at different sites can access the same physical disks, which may be distantly separated as well, via the common SAN. These remote disks can be combined into volume groups, using C-SPOC.

The logical volumes in a volume group can have up to three mirrors or copies, for example, one mirror at each site. Thus the information stored on this logical volume may be kept highly available, and in case of a certain failures—for example, all nodes at one site, including the disk subsystem at that site—the remote mirror at another site will still have the latest information and the operations can be continued on that site.

The primary intent of this feature is to support two-site clusters where LVM mirroring through a SAN replicates data between the disk subsystem at each site for disaster recovery.

Another advantage of cross-site LVM mirroring is that after a site/disk failure and subsequent site reintegration, HACMP attempts to synchronize the data from the surviving disks to the joining disks automatically. The synchronization occurs in the background and does *not* significantly impact the reintegration time.

The following figure illustrates a cross-site LVM mirroring configuration using a SAN:

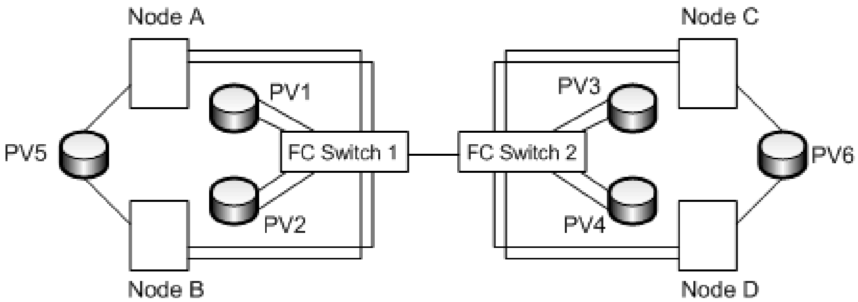

Figure 26. Cross-Site LVM Mirroring Configuration for Disaster Recovery

The disks that are connected to at least one node at each of the two sites can be mirrored. In this example, PV4 is seen by nodes A and B on Site 1 via the Fibre Channel Switch 1-Fibre Channel Switch 2 connection, and is also seen on node C via Fibre Channel Switch 2. You could have a mirror of PV4 on Site 1. The disks that are connected to the nodes on one site only (PV5 and PV6) cannot be mirrored across sites.

The disk information is replicated from a local site to a remote site. The speed of this data transfer depends on the physical characteristics of the channel, the distance, and LVM mirroring performance.

Cluster Configurations with Dynamic LPARs

The advanced partitioning features of AIX v. 5.2 and up provide the ability to dynamically allocate system CPU, memory, and I/O slot resources (*dynamic LPAR*).

Using HACMP in combination with LPARs lets you:

- Perform routine system upgrades through the dynamic allocation of system resources. When used with dynamic LPARs, HACMP can reduce the amount of downtime for well-planned systems upgrades by automating the transition of your application workload from one logical partition to another, so that the first logical partition may be upgraded without risk to the application.
- Effectively redistribute CPU and memory resources to manage the workload. Combining HACMP with dynamic LPAR lets you use customized application start and stop scripts to dynamically redistribute CPU and memory resources to logical partitions that are currently executing application workload, to further support application transition within a single frame. This way you maintain the processing power and resources necessary to support your applications, while minimal resources are devoted to upgrading, a less resource intensive task.

Note: Do *not* have all your cluster nodes configured as LPARs within the *same* physical server. This configuration could potentially be a significant single point of failure.

The following example illustrates a cluster configuration that uses three LPARs:

- LPAR #1 is running a back end database (DB2 UDB)
- LPAR #2 is running WebSphere Application Server (WAS)
- LPAR #3 is running as a backup (standby) for both the DB2 and WAS LPARs. This LPAR contains only minimal CPU and memory resources.

When it is time to move either the DB2 or WAS application to the third LPAR (due to a planned upgrade or a resource failure in these LPARs, for instance), you can use customized application start and stop scripts in HACMP to automate the dynamic reallocation of CPU and memory from the primary LPAR to the standby LPAR. This operation allows the third LPAR to acquire the CPU and memory resources necessary to meet business performance requirements. When HACMP moves the resource group containing the application back to its home LPAR, the CPU and memory resources automatically move with it.

Note: In general, dynamic LPARs allow dynamic allocation of CPU, memory and I/O slot resources. HACMP and dynamic LPAR I/O slot resources are *not* compatible (although you can dynamically allocate I/O slot resources outside of HACMP cluster).

The following figure illustrates this cluster environment:

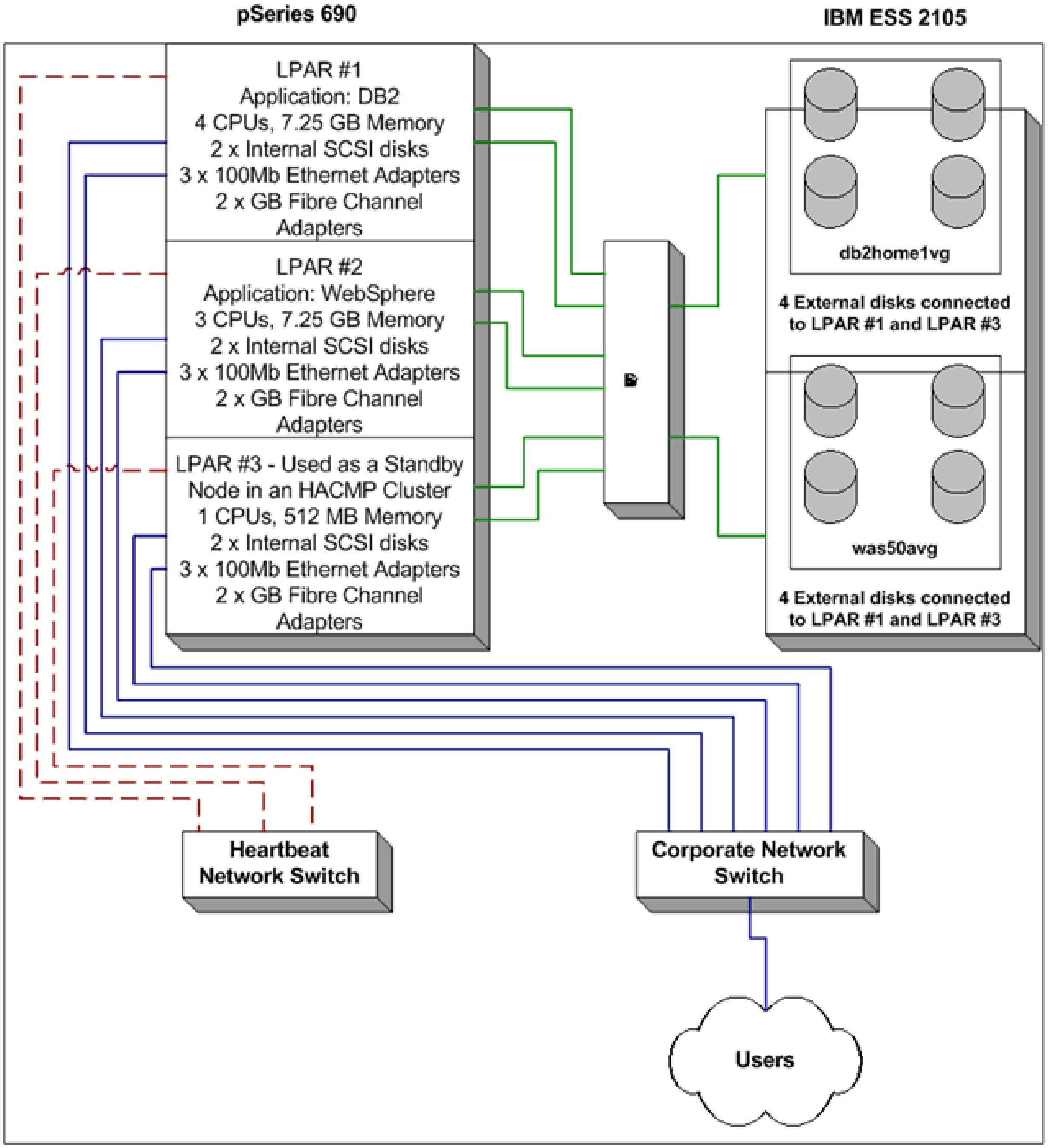
pSeries 690
IBM ESS 2105
LPAR #1
Application: DB2
4 CPUs, 7.25 GB Memory
2 x Internal SCSI disks
3 x 100Mb Ethernet Adapters
2 x GB Fibre Channel Adapters
LPAR #2
Application: WebSphere
3 CPUs, 7.25 GB Memory
2 x Internal SCSI disks
3 x 100Mb Ethernet Adapters
2 x GB Fibre Channel Adapters
LPAR #3 - Used as a Standby Node in an HACMP Cluster
1 CPUs, 512 MB Memory
2 x Internal SCSI disks
3 x 100Mb Ethernet Adapters
2 x GB Fibre Channel Adapters
db2home1vg
4 External disks connected to LPAR #1 and LPAR #3
was50avg
4 External disks connected to LPAR #1 and LPAR #3
Heartbeat Network Switch
Corporate Network Switch
Users

Figure 27. Cluster with Three LPARs

DLPARs and Capacity Upgrade on Demand

Capacity Upgrade on Demand (CUoD) is one of the features of Dynamic Logical Partitioning (DLPAR) on some of the System p™ IBM servers that lets you activate preinstalled but yet inactive processors as resource requirements change. The additional CPUs and memory, while physically present, are *not* used until you decide that the additional capacity you need is worth the cost. This provides you with a fast and easy upgrade in capacity to meet peak or unexpected loads.

HACMP 5.2 and up integrates with the Dynamic Logical Partitioning and CUoD functions. You can configure cluster resources in a way where the logical partition with minimally allocated resources serves as a standby node, and the application resides on another LPAR node that has more resources than the standby node.

Chapter 7: HACMP Configuration Process and Facilities

This chapter provides an overview of the HACMP cluster configuration process. It covers the following topics:

- Information You Provide to HACMP
- Information Discovered by HACMP
- Cluster Configuration Options: Standard and Extended.

This chapter also provides an overview of the following administrative tools supplied with the HACMP software:

- Cluster Security
- Installation, Configuration and Management Tools
- Monitoring Tools
- Troubleshooting Tools
- Cluster Test Tool
- Emulation Tools.

Information You Provide to HACMP

Prior to configuring a cluster, make sure the building blocks are planned and configured, and the initial communication path exists for HACMP to reach each node. This section covers the basic tasks you need to perform to configure a cluster.

Information on Physical Configuration of a Cluster

Physical configuration of a cluster consists of the following planning and configuration tasks:

- Ensure the TCP/IP network support for the cluster.
- Ensure the point-to-point network support for the cluster.
- Ensure the heartbeating support for the cluster.
- Configure the shared disk devices for the cluster.
- Configure the shared volume groups for the cluster.
- Consider the mission-critical applications for which you are using HACMP. Also, consider application server and what type of resource group management is best for each application.
- Examine issues relating to HACMP clients.
- Ensure physical redundancy by using multiple circuits or uninterruptable power supplies, redundant physical network interface cards, multiple networks to connect nodes and disk mirroring.

These tasks are described in detail in the *Planning Guide*.

AIX Configuration Information

Cluster components must be properly configured on the AIX level. For this task, ensure that:

- Basic communication to cluster nodes exists.
- Volume groups, logical volumes, mirroring and file systems are configured and set up. To ensure logical redundancy, consider different types of resource groups, and plan how you will group your resources in resource groups.

For the specifics of configuring volume groups, logical volumes and file systems, refer to the AIX manuals and to the *Installation Guide.*

Establishing the Initial Communication Path

The initial communication path is a path to a node that you are adding to a cluster. To establish the initial communication path, you provide the name of the node, or other information that can serve as the name of the node.

In general, a node name and a hostname can be the same. When configuring a new node, you can enter any of the following denominations that will serve as an initial communication path to a node:

- An IP address of a physical network interface card (NIC) on that node, such as `1.2.3.4.` In this case, the address is used as a communication path for contacting a node.
- An IP label associated with an IP address of a NIC on that node, such as `servername`. In this case, the name is used to determine the communication path for contacting a node, based on the assumption that the local TCP/IP configuration (Domain Nameserver or Hosts Table) supplies domain qualifiers and resolves the IP label to an IP address.
- A *Fully Qualified Domain Name (FQDN)*, such as "`servername.thecompanyname.com`". In this case the communication path is "`servername.thecompanyname.com`", based on the assumption that the local TCP/IP configuration (Domain Nameserver or Hosts Table) supplies domain qualifiers and resolves the IP label to an IP address.

When you enter any of these names, HACMP ensures unique name resolution and uses the hostname as a node name, unless you explicitly specify otherwise.

Note: In HACMP, node names and hostnames have to be different in some cases where the application you are using requires that the AIX “hostname attribute” moves with the application in the case of a cluster component failure. This procedure is done through setting up special event scripts.

If the nodes and physical network interface cards have been properly configured to AIX, HACMP can use this information to assist you in the configuration process, by running the automatic discovery process discussed in the following section.

Information Discovered by HACMP

You can define the basic cluster components in just a few steps. To assist you in the cluster configuration, HACMP can automatically retrieve the information necessary for configuration from each node.

Note: For easier and faster cluster configuration, you can also use a cluster configuration assistant. For more information, see Two-Node Cluster Configuration Assistant.

For the automatic discovery process to work, the following conditions should be met in HACMP:

- You have previously configured the physical components and performed all the necessary AIX configurations.
- Working communications paths exist to each node. This information will be used to automatically configure the cluster TCP/IP topology when the *standard configuration path* is used.

Once these tasks are done, HACMP automatically discovers predefined physical components within the cluster, and selects default behaviors. In addition, HACMP performs discovery of cluster information if there are any changes made during the configuration process.

Running discovery retrieves current AIX configuration information from all cluster nodes. This information appears in picklists to help you make accurate selections of existing components.

The HACMP automatic discovery process is easy, fast, and does *not* place a "waiting" burden on you as the cluster administrator.

Cluster Configuration Options: Standard and Extended

In this section, the configuration process is significantly simplified. While the details of the configuration process are covered in the *Administration Guide*, this section provides a brief overview of two ways to configure an HACMP cluster.

Configuring an HACMP Cluster Using the Standard Configuration Path

You can add the basic components of a cluster to the HACMP configuration database *in a few steps*. The standard cluster configuration path simplifies and speeds up the configuration process, because HACMP automatically launches discovery to collect the information and to select default behaviors.

If you use this path:

- Automatic discovery of cluster information runs by default. Before starting the HACMP configuration process, you need to configure network interfaces/devices in AIX. In HACMP, you establish initial communication paths to other nodes. Once this is done, HACMP collects this information and automatically configures the cluster nodes and networks based on physical connectivity. All discovered networks are added to the cluster configuration.

- IP aliasing is used as the *default* mechanism for binding IP labels/addresses to network interfaces.
- You can configure the most common types of resources. However, customizing of resource group fallover and fallback behavior is limited.

Configuring an HACMP Cluster Using the Extended Configuration Path

In order to configure the less common cluster elements, or if connectivity to each of the cluster nodes is *not* established, you can manually enter the information in a way similar to previous releases of the HACMP software.

When using the HACMP extended configuration SMIT paths, if any components are on remote nodes, you must manually initiate the discovery of cluster information. That is, discovery is optional (rather than automatic, as it is when using the standard HACMP configuration SMIT path).

Using the options under the extended configuration menu, you can add the basic components of a cluster to the HACMP configuration database, as well as many additional types of resources. Use the extended configuration path to customize the cluster for all the components, policies, and options that are *not* included in the standard configuration menus.

Overview: HACMP Administrative Facilities

The HACMP software provides you with the following administrative facilities:

- Cluster Security
- Installation, Configuration and Management Tools
- Monitoring Tools
- Troubleshooting Tools
- Emulation Tools.

Cluster Security

All communication between nodes is sent through the Cluster Communications daemon, **clcomd**, which runs on each node. The **clcomd** daemon manages the connection authentication between nodes and any message authentication or encryption configured. HACMP’s Cluster Communications daemon uses the trusted **/usr/es/sbin/cluster/etc/rhosts** file, and removes reliance on an **/.rhosts** file. In HACMP 5.2 and up, the daemon provides support for message authentication and encryption.

Installation, Configuration and Management Tools

HACMP includes the tools described in the following sections for installing, configuring, and managing clusters.

Two-Node Cluster Configuration Assistant

HACMP provides the Two-Node Cluster Configuration Assistant to simplify the process for configuring a basic two-node cluster. The wizard-like application requires the minimum information to define an HACMP cluster and uses discovery to complete the cluster configuration. The application is designed for users with little knowledge of HACMP who want to quickly set up a basic HACMP configuration. The underlying AIX configuration must be in place before you run the Assistant.

Smart Assists for Integrating Specific Applications with HACMP

The Smart Assist for a given application examines the configuration on the system to determine the resources HACMP needs to monitor (Service IP label, volume groups). The Smart Assist then configures one or more resource groups to make applications and their resources highly available.

The Smart Assist takes the following actions:

- Discovers the installation of the application and if necessary the currently configured resources such as service IP address, file systems and volume groups
- Provides a SMIT interface for getting or changing configuration information from the user including a new service IP address
- Defines the application to HACMP and supplies custom start and stop scripts for it
- Supplies an application monitor for the application
- Configures a resource group to contain:
  - Primary and takeover nodes
  - The application
  - The service IP address
  - Shared volume groups.
- Configures resource group temporal and location dependencies, should the application solution require this
- Specifies files that need to be synchronized using the HACMP File Collections feature
- Modifies previously configured applications as necessary
- Verifies the configuration
- Tests the application's cluster configuration.

Supported Applications

HACMP 5.4.1 supplies Smart Assists for the following applications and configuration models:

- DB2
  - DB2 - Hot Standby

  - DB2 - Mutual Takeover
- WebSphere 6.0
  - WebSphere Application Server 6.0
  - WebSphere Cluster Transaction Log recovery
  - Deployment Manager
  - Tivoli Directory Server
  - IBM HTTP Server
- Oracle 10G

General Application Smart Assist

The General Application Smart Assist helps users to configure installed applications that do *not* have their own Smart Assist. The user supplies some basic information such as:

- Primary node - by default, the local node
- Takeover node(s) - by default, all configured nodes except the local node
- Application Name
- Application Start Script
- Application Stop Script
- Service IP label.

The General Smart Assist then completes the cluster configuration in much the same way as the Two-Node Cluster Configuration Assistant (but the configuration can have more than two nodes). The user can modify, test, or remove the application when using the General Application Smart Assist.

Smart Assist API

HACMP 5.4.1 includes a *Smart Assist Developers Guide* so that OEMs can develop Smart Assists to integrate their own applications with HACMP.

Planning Worksheets

Along with your HACMP software and documentation set, you have two types of worksheets to aid in planning your cluster topology and resource configuration: online or paper.

Online Planning Worksheets

HACMP provides the Online Planning Worksheets application, which enables you to:

- Plan a cluster.
- Create a cluster definition file.
- Examine the configuration for an HACMP cluster. You can review information about a cluster configuration in an easy-to-view format for use in testing and troubleshooting situations.

After you save an HACMP cluster definition file, you can open that file in an XML editor or in Online Planning Worksheets running on a node, a laptop, or other computer running the application. This enables you to examine the cluster definition on a non-cluster node or share the file with a colleague.

Besides providing an easy-to-view format, the XML structure enables your configuration information to be quickly converted from one format to another, which eases data exchange between applications. For example, you can save a cluster snapshot and then import it into your OLPW configuration.

For more information on the requirements and instructions for using the Online Planning Worksheets application, see the *Planning Guide*.

Paper Worksheets

The HACMP documentation includes a set of planning worksheets to guide your entire cluster planning process, from cluster topology to resource groups and application servers. You can use these worksheets as guidelines when installing and configuring your cluster. You may find these paper worksheets useful in the beginning stages of planning. The planning worksheets are found in the *Planning Guide*.

Starting, Stopping and Restarting Cluster Services

Once you install HACMP and configure your cluster, you can start cluster services. In HACMP 5.4.1, your options for starting, stopping and restarting cluster services have been streamlined and improved. HACMP handles your requests to start and stop cluster services without disrupting your applications, allowing you to have full control.

In HACMP 5.4.1, you can:

- *Start and restart cluster services*. When you start cluster services, or restart them after a shutdown, HACMP by default automatically activates the resources according to how you defined them, taking into consideration application dependencies, application start and stop scripts, dynamic attributes and other parameters. That is, HACMP automatically manages (and activates, if needed) resource groups and applications in them.

  You can also start HACMP cluster services and tell it *not to start up any resource groups* (and applications) automatically for you. If an application is already running, you no longer need to stop it before starting the cluster services. HACMP relies on the application monitor and application startup script to verify whether it needs to start the application for you or the application is already running (HACMP attempts *not* to start a second instance of the application).

  Note: HACMP relies on the configured application monitors to detect application failures. Application monitors must be configured for HACMP to detect a running cluster during startup so that it does not start duplicate instances of the application. The alternative approach is to run scripts that ensure duplicate instances of the application server are *not* started.

- *Shut down the cluster services*. During an HACMP shutdown, you may select one of the following three actions for the resource groups:
  - Bring Offline.

- Move to other node(s).
- Place resource groups in an UNMANAGED state.

The Cluster Manager "remembers" the state of all the nodes and responds appropriately when users attempt to restart the nodes.

For information on how to configure application monitors as well as HACMP cluster startup and shutdown options, see the *Administration Guide*

SMIT Interface

You can use the SMIT panels supplied with the HACMP software to perform the following tasks:

- Configure clusters, nodes, networks, resources, and events.
- Capture and restore snapshots of cluster configurations.
- Read log files.
- Diagnose cluster problems.
- Manage a cluster using the C-SPOC utility.
- Perform resource group management tasks.
- Configure Automatic Error Notification.
- Perform dynamic adapter swap.
- Configure cluster performance tuning.
- Configure custom disk methods.

Web-Based SMIT Interface

WebSMIT is a Web-based user interface that provides consolidated access to the SMIT functions of configuration and management, display of interactive cluster status, and the HACMP documentation. Starting with HACMP 5.4, you can use WebSMIT to navigate and view the status of the running cluster, configure and manage the cluster, and view graphical displays of sites, networks, nodes and resource group dependencies.

The WebSMIT interface is similar to the ASCII SMIT interface. Because WebSMIT runs in a Web browser, it can be accessed from any platform.

To use the WebSMIT interface, you must configure and run a Web server process on at least one of the cluster node(s) to be administered. The **/usr/es/sbin/cluster/wsm/README** file contains information on basic Web server configuration, the default security mechanisms in place when HACMP is installed, and the configuration files available for customization.

For more information on installing and configuring WebSMIT, see the *Installation Guide*.

For more information on using WebSMIT, see Using WebSMIT for Configuring, Managing, and Monitoring a Cluster in Chapter 2: Administering an HACMP Cluster using WebSMIT in the *Administration Guide*.

Cluster Status Display Linked to Management Functions

When using the WebSMIT interface to see the cluster status display, you have links to the related WebSMIT management functions. Therefore, HACMP provides a consolidated user interface for cluster status with management capabilities.

For example, the node status display has a link to (among other options) the SMIT panels for starting and stopping Cluster Services. Now you can manipulate entities in the status display interactively rather than having to go to an ASCII SMIT interface on the node.

Specifying Read-Only User Access

In HACMP 5.4.1, you can specify a group of users that have read-only access to WebSMIT. Users with read-only access may view the configuration and cluster status, and may navigate through the WebSMIT screens, but cannot execute commands or make any changes to the configuration. For more information about configuring read-only access to WebSMIT, see the section on WebSMIT Security Considerations and WebSMIT Prerequisites in the *Installation Guide*.

HACMP System Management with C-SPOC

To facilitate management of a cluster, HACMP provides a way to run commands from one node and then verify and synchronize the changes to all the other nodes. You can use the HACMP System Management tool, the Cluster Single Point of Control (C-SPOC) to add users, files, and hardware automatically without stopping mission-critical jobs.

C-SPOC lets you perform the following tasks:

- Start/Stop HACMP Services
- HACMP Communication Interface Management
- HACMP Resource Group and Application Management
- HACMP File Collection Management
- HACMP Log Viewing and Management
- HACMP Security and Users Management
- HACMP Logical Volume Management
- HACMP Concurrent Logical Volume Management
- HACMP Physical Volume Management
- GPFS File System Support
- Open a SMIT Session on a Node.

The C-SPOC utility simplifies maintenance of shared LVM components in clusters of up to 32 nodes. C-SPOC commands provide comparable functions in a cluster environment to the standard AIX commands that work on a single node. By automating repetitive tasks, C-SPOC eliminates a potential source of errors, and speeds up the process.

Without C-SPOC functionality, the system administrator must execute administrative tasks individually on each cluster node. For example, to add a user you usually must perform this task on each cluster node. Using the C-SPOC utility, a command executed on one node is also executed on other cluster nodes. Thus C-SPOC minimizes administrative overhead and reduces the possibility of inconsistent node states. Using C-SPOC, you issue a C-SPOC command once on a single node, and the user is added to all specified cluster nodes.

C-SPOC also makes managing logical volume components and controlling cluster services more efficient. You can use the C-SPOC utility to start or stop cluster services on nodes from a single node. The following figure illustrates a two-node configuration and the interaction of commands, scripts, and nodes when starting cluster services from a single cluster node. Note the prefix **cl_** begins all C-SPOC commands.

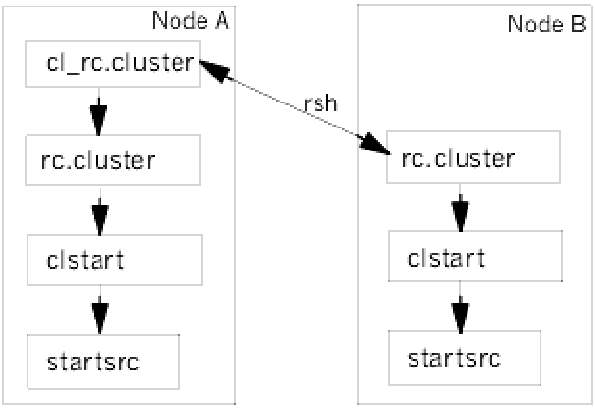

Figure 28. Flow of Commands Used at Cluster Startup by C-SPOC Utility

C-SPOC provides this functionality through its own set of cluster administration commands, accessible through SMIT menus and panels. To use C-SPOC, select the **Cluster System Management** option from the HACMP SMIT menu.

Cluster Snapshot Utility

The Cluster Snapshot utility allows you to save cluster configurations you would like to restore later. You also can save additional system and cluster information that can be useful for diagnosing system or cluster configuration problems. You can create your own custom snapshot methods to store additional information about your cluster.

A cluster snapshot lets you skip saving log files in the snapshot. Cluster snapshots are used for recording the cluster configuration information, whereas cluster logs only record the operation of the cluster and *not* the configuration information. By default, HACMP no longer collects cluster log files when you create the cluster snapshot, although you can still specify collecting the logs in SMIT. Skipping the logs collection speeds up the running time of the snapshot utility and reduces the size of the snapshot.

Customized Event Processing

You can define multiple pre- and post-events to tailor your event processing for your site's unique needs. For more information about writing your own scripts for pre- and post-events, see the *Administration Guide*.

Resource Group Management Utility

The resource group management utility, **clRGmove**, provides a means for managing resource groups in the cluster, and enhances failure recovery capabilities of HACMP. It allows you to move any type of resource group (along with its resources—IP addresses, applications, and disks) online, offline or to another node, without stopping cluster services.

Resource group management helps you to manage your cluster more effectively, giving you better use of your cluster hardware resources. Resource group management also lets you perform selective maintenance without rebooting the cluster or disturbing operational nodes. For instance, you can use this utility to free the node of any resource groups to perform system maintenance on a particular cluster node.

Using the resource group management utility does *not* affect other resource groups currently owned by a node. The current node releases it, and the destination node acquires it just as it would during a node fallover. (If you have location dependencies configured between resource groups, HACMP verifies and ensures that they are honored).

Use resource group management to:

- Temporarily move a non-concurrent resource group from one node to another (and from one site to another) in a working cluster.
- Bring a resource group online or offline on one or all nodes in the cluster.

When you move a group, it stays on the node to which it was moved, until you move it again.

If you move a group that has Fallback to Highest Priority Node fallback policy, the group falls back or returns to its "new" temporary highest priority node (in cases when HACMP has to recover it on other nodes during subsequent cluster events).

If you want to move the group again, HACMP intelligently informs you (in the picklists with destination nodes) if it finds that a node with a higher priority exists that can host a group. You can always choose to move the group to that node.

HACMP File Collection Management

Like volume groups, certain files located on each cluster node need to be kept in sync in order for HACMP (and other applications) to behave correctly. Such files include event scripts, application scripts, and some AIX and HACMP configuration files.

HACMP File Collection management provides an easy way to request that a list of files be kept in sync across the cluster. Using HACMP file collection, you do *not* have to manually copy an updated file to every cluster node, verify that the file is properly copied, and confirm that each node has the same version of it.

Also, if one or more of these files is inadvertently deleted or damaged on one or more cluster nodes, it can take time and effort to determine the problem. Using HACMP file collection, this scenario is mitigated. HACMP detects when a file in a file collection is deleted or if the file size is changed to zero, and logs a message to inform the administrator.

Two predefined HACMP file collections are installed by default:

- **Configuration_Files**. A container for essential system files, such as **/etc/hosts** and **/etc/services**.
- **HACMP_Files**. A container for all the user-configurable files in the HACMP configuration. This is a special file collection that the underlying file collection propagation utility uses to reference all the user-configurable files in the HACMP configuration database (ODM) classes.

For a complete list of configuration files and user-configurable HACMP files, see the *Installation Guide*.

For information on configuring file collections in SMIT, see the *Administration Guide*.

Monitoring Tools

HACMP supplies the monitoring tools described in the following sections:

- Cluster Manager
- Cluster Information Program
- Application Monitoring
- Show Cluster Applications SMIT Option
- Cluster Status Utility (clstat)
- HAView Cluster Monitoring Utility
- Cluster Monitoring and Administration with Tivoli Framework
- Application Availability Analysis Tool
- Persistent Node IP Labels
- HACMP Verification and Synchronization.

Many of the utilities described in this section use the **clhosts** file to enable communication among HACMP cluster nodes. For information about the **clhosts** file, see Understanding the clhosts File. For detailed information about using each of these monitoring utilities, see the *Administration Guide.*

Cluster Manager

The Cluster Manager provides SNMP information and traps for SNMP clients. It gathers cluster information relative to cluster state changes of nodes and interfaces. Cluster information can be retrieved using SNMP commands or by SNMP-based client programs such as HATivoli. For more information, see the section Cluster Manager and SNMP Monitoring Programs in Chapter 4: HACMP Cluster Hardware and Software.

Cluster Information Program

The Cluster Information Program (Clinfo) gathers cluster information from SNMP and enables clients communicating with this program to be aware of changes in a cluster state. For information about Clinfo, see the section Cluster Information Program in Chapter 4: HACMP Cluster Hardware and Software.

Application Monitoring

Application Monitoring enables you to configure multiple monitors for an application server to monitor specific applications and processes; and define action to take upon detection of an unexpected termination of a process or other application failures. See the section Application Monitors in Chapter 5: Ensuring Application Availability.

Show Cluster Applications SMIT Option

The Show Cluster Applications SMIT option provides an application-centric view of the cluster configuration. This utility displays existing interfaces and information in an “application down” type of view. You can access it from both ASCII SMIT and WebSMIT.

Cluster Status Utility (clstat)

The Cluster Status utility, **/usr/es/sbin/cluster/clstat**, monitors cluster status. The utility reports the status of key cluster components: the cluster itself, the nodes in the cluster, the network interfaces connected to the nodes, and the resource groups on each node. It reports whether the cluster is up, down, or unstable. It also reports whether a node is up, down, joining, leaving, or reconfiguring, and the number of nodes in the cluster. The **clstat** utility provides ASCII, Motif, X Windows, and HTML interfaces. You can run **clstat** from either ASCII SMIT or WebSMIT.

For the cluster as a whole, **clstat** indicates the cluster state and the number of cluster nodes. For each node, **clstat** displays the IP label and address of each service network interface attached to the node, and whether that interface is up or down. **clstat** also displays resource group state.

You can view cluster status information in ASCII or X Window display mode or through a web browser.

Note: The **clstat** utility uses the Clinfo API to retrieve information about the cluster. Therefore, ensure Clinfo is running on the client system to view the **clstat** display.

HAView Cluster Monitoring Utility

The **HAView** utility extends Tivoli NetView services so you can monitor HACMP clusters and cluster components across a network from a single node. Using HAView, you can also view the full cluster event history in the **/var/hacmp/adm/history/cluster.mmddyyyy** file.

The HAView cluster monitoring utility makes use of the Tivoli TME 10 NetView for AIX graphical interface to provide a set of visual maps and submaps of HACMP clusters. HAView extends NetView services to allow you to monitor HACMP clusters and cluster components across a network from a single node. HAView creates symbols that reflect the state of all nodes, networks, and network interface objects associated in a cluster. You can also monitor resource groups and their resources through HAView.

HAView monitors cluster status using the Simple Network Management Protocol (SNMP). It combines periodic polling and event notification through traps to retrieve cluster topology and state changes from the HACMP Management Information Base (MIB). The MIB is maintained by the Cluster Manager, the HACMP management agent. HAView allows you to:

- View maps and submaps of cluster symbols showing the location and status of nodes, networks, and addresses, and monitor resource groups and resources.
- View detailed information in NetView dialog boxes about a cluster, network, IP address, and cluster events.
- View cluster event history using the HACMP Event Browser.
- View node event history using the Cluster Event Log.
- Open a SMIT HACMP session for an active node and perform cluster administration functions from within HAView, using the HAView Cluster Administration facility.

Cluster Monitoring and Administration with Tivoli Framework

The Tivoli Framework enterprise management system enables you to monitor the state of an HACMP cluster and its components and perform cluster administration tasks. Using various windows of the Tivoli Desktop, you can monitor the following aspects of your cluster:

- Cluster state and substate
- Configured networks and network state
- Participating nodes and node state
- Configured resource groups and resource group state
- Resource group location.

In addition, you can perform the following cluster administration tasks through Tivoli:

- Start cluster services on specified nodes.
- Stop cluster services on specified nodes.
- Bring a resource group online.
- Bring a resource group offline.
- Move a resource group to another node.

For complete information about installing, configuring, and using the cluster monitoring through Tivoli functionality, see the *Administration Guide.*

Application Availability Analysis Tool

The Application Availability Analysis tool measures uptime statistics for applications with application servers defined to HACMP. The HACMP software collects, time-stamps, and logs extensive information about the applications you choose to monitor with this tool. Using SMIT, you can select a time period and the tool displays uptime and downtime statistics for a given application during that period.

Persistent Node IP Labels

A *persistent node IP label* is a useful administrative “tool” that lets you contact a node even if the HACMP cluster services are down on that node. (In this case, HACMP attempts to put an IP address on the node). Assigning a persistent node IP label to a network on a node allows you to have a node-bound IP address on a cluster network that you can use for administrative purposes to access a specific node in the cluster.

A persistent node IP label is an IP alias that can be assigned to a specific node on a cluster network and that:

- Always stays on the same node (is *node-bound*)
- Co-exists on a network interface card that already has a service IP label defined
- Does *not* require installing an additional physical network interface card on that node
- Is *not* part of any *resource group*.

There can be one persistent node IP label per network per node.

HACMP Verification and Synchronization

The HACMP verification and synchronization process verifies that HACMP-specific modifications to AIX system files are correct, that the cluster and its resources are configured correctly, that security (if set up) is configured correctly, that all nodes agree on the cluster topology, network configuration, and the ownership and takeover of HACMP resources, among other things. Verification also indicates whether custom cluster snapshot methods exist and whether they are executable on each cluster node.

Whenever you have configured, reconfigured, or updated a cluster, you should then run the cluster verification procedure. If the verification succeeds, the configuration is automatically synchronized. Synchronization takes effect immediately on an active cluster.

The verification utility keeps a detailed record of the information in the HACMP configuration database on each of the nodes after it runs. Subdirectories for each node contain information for the last successful verification (pass), the next-to-last successful verification (pass.prev), and the last unsuccessful verification (fail).

Messages output by the utility indicate where the error occurred (for example, the node, device, command, and so forth).

Verification with Automatic Cluster Configuration Monitoring

HACMP 5.4.1 provides automatic cluster configuration monitoring. By default, HACMP automatically runs **verification** on the node that is first in alphabetical order once every 24 hours at midnight. The cluster administrator is notified if the cluster configuration has become invalid.

When cluster verification completes on the selected cluster node, this node notifies the other cluster nodes. Every node stores the information about the date, time, which node performed the verification, and the results of the verification in the **/var/hacmp/log/clutils.log** file. If the selected node becomes unavailable or cannot complete cluster verification, you can detect this by the lack of a report in the **/var/hacmp/log/clutils.log** file.

If cluster verification completes and detects some configuration errors, you are notified about the potential problems:

- The exit status of verification is published across the cluster along with the information about cluster verification process completion.
- Broadcast messages are sent across the cluster and displayed on **stdout**. These messages inform you about detected configuration errors.
- A general_notification event runs on the cluster and is logged in **hacmp.out** (if cluster services is running).

Verification with Corrective Actions

Cluster verification consists of a series of checks performed against various user-configured HACMP server components. Each check attempts to detect either a cluster consistency issue or a configuration error. Some error conditions result when information important to the operation of HACMP, but *not* part of the HACMP software itself, is *not* propagated properly to all cluster nodes.

By default, verification runs with the automatic corrective actions mode enabled for both the Standard and Extended configuration. This is the recommended mode for running verification. If necessary, the automatic corrective actions mode can be disabled for the Extended configuration. However, note that running verification in automatic corrective action mode enables you to automate many configuration tasks, such as creating a client-based **clhosts** file, which is used by many of the monitors described in this chapter. For more information about both the client-based and server-based **clhosts** file, see the section Understanding the clhosts File.

When **verification** detects any of the following conditions, you can authorize a corrective action before error checking continues:

- HACMP shared volume group time stamps do *not* match on all nodes.
- The **/etc/hosts** file on a node does *not* contain all HACMP-managed labels/IP addresses.
- SSA concurrent volume groups need SSA node numbers.
- A file system is *not* created on a node that is part of the resource group, although disks are available.
- Disks are available, but a volume group has *not* been imported to a node.
- Required **/etc/services** entries are missing on a node.
- Required HACMP **snmpd** entries are missing on a node.

If an error found during verification triggers any corrective actions, then the utility runs all checks again after it finishes the first pass. If the same check fails again and the original problem is an error, the error is logged and verification fails. If the original condition is a warning, verification succeeds.

Custom Verification Methods

Through SMIT you also can add, change, or remove custom-defined verification methods that perform specific checks on your cluster configuration.

You can perform verification from the command line or through the SMIT interface to issue a customized remote notification method in response to a cluster event.

Understanding the clhosts File

Many of the monitors described in this section, including Clinfo, HAView, and **clstat** rely on the use of a **clhosts** file. The **clhosts** file contains IP address information that helps enable communications among HACMP cluster nodes. The **clhosts** file resides on all HACMP cluster servers and clients. There are differences, depending on where the file resides, as summarized in the following table.

clhosts File	Description
server-based file	This file resides in the **/usr/es/sbin/cluster/etc/** directory on all HACMP server nodes. During the installation of HACMP, the IP address `127.0.0.1` is automatically added to the file. (The name `loopback` and the alias `localhost` that this IP address usually defines are *not* required.)
client-based file	This file resides in the **/usr/es/sbin/cluster/etc/** directory on all HACMP client nodes. When you run verification with automatic corrections enabled, this file is automatically generated on the server and named **clhosts.client**. You must copy the file to the client nodes manually and rename it as **clhosts**. This file contains all known IP addresses. It should never contain `127.0.0.1`, `loopback`, or `localhost`.

When a monitor daemon starts up, it reads in the local **/usr/es/sbin/cluster/etc/clhosts** file to determine which nodes are available for communication as follows:

- For daemons running on an HACMP server node, the local server-based **clhosts** file only requires the loopback address (`127.0.0.1`), that is automatically added to the server-based **clhosts** file when the server portion of HACMP is installed.
- For daemons running on an HACMP client node, the local client-based **clhosts** file should contain a list of the IP addresses for the HACMP server nodes. In this way, if a particular HACMP server node is unavailable (for example, powered down), then the daemon on the client node still can communicate with other HACMP server nodes.

The HACMP verification utility assists in populating the client-based **clhosts** file in the following manner:

When you run cluster verification with automatic corrective actions enabled, HACMP finds all available HACMP server nodes, creates a **/usr/es/sbin/cluster/etc/clhosts.client** file on the server nodes, and populates the file with the IP addresses of those HACMP server nodes.

After you finish verifying and synchronizing HACMP on your cluster, you must manually copy this **clhosts.client** file to each client node as **/usr/es/sbin/cluster/etc/clhosts** (rename it by removing the **.client** extension).

For more information about verification, see HACMP Verification and Synchronization in this chapter.

Troubleshooting Tools

Typically, a functioning HACMP cluster requires minimal intervention. If a problem occurs, however, diagnostic and recovery skills are essential. Thus, troubleshooting requires that you identify the problem quickly and apply your understanding of the HACMP software to restore the cluster to full operation.

HACMP supplies the following tools:

- Log Files
- Resetting HACMP Tunable Values
- Cluster Status Information File
- Automatic Error Notification
- Custom Remote Notification
- User-Defined Events
- Event Preambles and Summaries
- Trace Facility.

These utilities are described in the following sections. For more detailed information on each of these utilities, see the *Administration Guide*.

For general testing or emulation tools see:

- Cluster Test Tool
- Emulation Tools.

Log Files

The HACMP software writes the messages it generates to the system console and to several log files. Because each log file contains a different level of detail, system administrators can focus on different aspects of HACMP processing by viewing different log files. The main log files include:

- The **/var/hacmp/adm/cluster.log** file tracks cluster events.
- The **/var/hacmp/log/hacmp.out** file records the output generated by configuration scripts as they execute. Event summaries appear after the verbose output for events initiated by the Cluster Manager, making it easier to scan the **hacmp.out** file for important information. In addition, event summaries provide HTML links to the corresponding events within the **hacmp.out** file.
- The **/var/hacmp/adm/history/cluster.mmddyyyy** log file logs the daily cluster history.
- The /**var/hacmp/clverify/clverify.log** file contains the verbose messages output during verification. Cluster verification consists of a series of checks performed against various HACMP configurations. Each check attempts to detect either a cluster consistency issue or an error. The messages output by the verification utility indicate where the error occurred (for example, the node, device, command, and so forth).

HACMP lets you view, redirect, save and change parameters of the log files, so you can tailor them to your particular needs.

You can also collect log files for problem reporting. For more information, see the *Administration Guide*.

Resetting HACMP Tunable Values

While configuring and testing a cluster, you may change a value for one of the HACMP tunable values that affects the cluster performance. Or, you may want to reset tunable values to their default settings without changing any other aspects of the configuration. A third-party cluster administrator or a consultant may be asked to take over the administration of a cluster that they did *not* configure and may need to reset the tunable values to their defaults.

You can reset cluster tunable values using the SMIT interface. HACMP takes a cluster snapshot, prior to resetting. After the values have been reset to defaults, if you want to return to customized cluster settings, you can apply the cluster snapshot.

Resetting the cluster tunable values resets information in the cluster configuration database. The information that is reset or removed comprises two categories:

- Information supplied by the users (for example, pre- and post-event scripts and network parameters, such as netmasks). Note that resetting cluster tunable values *does not* remove the pre- and post-event scripts that you already have configured. However, if you reset the tunable values, HACMP's knowledge of pre- and post-event scripts is removed from the configuration, and these scripts are no longer used by HACMP to manage resources in your cluster. You can reconfigure HACMP to use these scripts again, if needed.
- Information automatically generated by HACMP during configuration and synchronization. This includes node and network IDs, and information discovered from the operating system, such as netmasks. Typically, users cannot see generated information.

For a complete list of tunable values that you can restore to their default settings, see the *Installation Guide.*

For instructions on how to reset the tunable values using SMIT, see the *Administration Guide.*

Cluster Status Information File

When you use the HACMP Cluster Snapshot utility to save a record of a cluster configuration (as seen from each cluster node), you optionally cause the utility to run many standard AIX commands and HACMP commands to obtain status information about the cluster. This information is stored in a file, identified by the **.info** extension, in the snapshots directory. The snapshots directory is defined by the value of the SNAPSHOTPATH environment variable. By default, the cluster snapshot utility includes the output from the commands, such as **cllssif**, **cllsnw**, **df**, **ls**, and **netstat**. You can create custom snapshot methods to specify additional information you would like stored in the **.info** file.

A cluster snapshot lets you skip saving log files in the snapshot. Cluster snapshots are used for recording the cluster configuration information, whereas cluster logs only record the operation of the cluster and *not* the configuration information. By default, HACMP no longer collects cluster log files when you create the cluster snapshot, although you can still specify collecting the logs in SMIT. Skipping the logs collection reduces the size of the snapshot and speeds up running the snapshot utility. The size of the cluster snapshot depends on the configuration. For instance, a basic two-node configuration requires roughly 40KB.

Automatic Error Notification

You can use the AIX Error Notification facility to detect events *not* specifically monitored by the HACMP software—a disk adapter failure, for example—and specify a response to take place if the event occurs.

Normally, you define error notification methods manually, one by one. HACMP provides a set of pre-specified notification methods for important errors that you can automatically “turn on” in one step through the SMIT interface, saving considerable time and effort by *not* having to define each notification method manually.

Custom Remote Notification

You can define a notification method through the SMIT interface to issue a customized notification method in response to a cluster event. In HACMP 5.4.1, you can also send text messaging notification to any address including a cell phone, or mail to an email address.

After configuring a remote notification method, you can send a test message to confirm that the configuration is correct.

You can configure any number of notification methods, for different events and with different text messages and telephone numbers to dial. The same notification method can be used for several different events, as long as the associated text message conveys enough information to respond to all of the possible events that trigger the notification.

User-Defined Events

You can define your own events for which HACMP can run your specified recovery programs. This adds a new dimension to the predefined HACMP pre- and post-event script customization facility.

Note: *HACMP 5.2 and up interact with the RSCT Resource Monitoring and Control (RMC) subsystem instead of with the RSCT Event Management subsystem. (The Event Management subsystem continues to be used for interaction with Oracle 9i). Only a subset of Event Management user-defined event definitions is automatically converted to the corresponding RMC event definitions, upon migration to HACMP 5.2 and up. After migration is complete, all user-defined event definitions must be manually reconfigured with the exception of seven UDE definitions defined by DB2. For more information, see the Administration Guide.*

You specify the mapping between events that you define and recovery programs defining the event recovery actions through the SMIT interface. This lets you control both the scope of each recovery action and the number of event steps synchronized across all nodes. For details about registering events, see the *RSCT documentation.*

You must put all the specified recovery programs on all nodes in the cluster, and make sure they are executable, before starting the Cluster Manager on any node.

- *AIX resource monitor*. This monitor generates events for OS-related events such as the percentage of CPU that is idle or percentage of disk space in use. The attribute names start with:
  - `IBM.Host.`
  - `IBM.Processor.`
  - `IBM.PhysicalVolume.`
- *Program resource monitor*. This monitor generates events for process-related occurrences such as unexpected termination of a process. It uses the resource attribute `IBM.Program.ProgramName`.

Note: You cannot use the Event Emulator to emulate a user-defined event.

Event Preambles and Summaries

Details of cluster events are recorded in the **hacmp.out** file. The verbose output of this file contains many lines of event information; you see a concise summary at the end of each event's details. For a quick and efficient check of what has happened in the cluster lately, you can view a compilation of only the event summary portions of current and previous **hacmp.out** files, by using the **Display Event Summaries** panel in SMIT. You can also select to save the compiled event summaries to a file of your choice. Optionally, event summaries provide HTML links to the corresponding events in the **hacmp.out** file.

The Cluster Manager also prints out a preamble that tells you which resource groups are enqueued for processing for each event; you can see the processing order that will be followed.

For details on viewing event preambles and summaries, see the *Troubleshooting Guide*.

Trace Facility

If the log files have no relevant information and the component-by-component investigation does *not* yield concrete results, you may need to use the HACMP trace facility to attempt to diagnose the problem. The trace facility provides a detailed look at selected system events.

Note that both the HACMP and AIX software must be running in order to use HACMP tracing.

For details on using the trace facility, see the *Troubleshooting Guide.*

Cluster Test Tool

The Cluster Test Tool is a utility that lets you test an HACMP cluster configuration to evaluate how a cluster behaves under a set of specified circumstances, such as when a node becomes inaccessible, a network becomes inaccessible, a resource group moves from one node to another, and so forth. You can start the test, let it run unattended, and return later to evaluate the results of your testing.

If you want to run an automated suite of basic cluster tests for topology and resource group management, you can run the automated test suite from SMIT. If you are an experienced HACMP administrator and want to tailor cluster testing to your environment, you can also create custom tests that can be run from SMIT.

It is recommended to run the tool after you initially configure HACMP and before you put your cluster into a production environment; after you make cluster configuration changes while the cluster is out of service; or at regular intervals even though the cluster appears to be functioning well.

Emulation Tools

HACMP includes the Event Emulator for running cluster event emulations and the Error Emulation functionality for testing notification methods.

HACMP Event Emulator

The HACMP Event Emulator is a utility that emulates cluster events and dynamic reconfiguration events by running event scripts that produce output but do *not* affect the cluster configuration or status. Emulation allows you to predict a cluster's reaction to a particular event just as though the event actually occurred.

The Event Emulator follows the same procedure used by the Cluster Manager given a particular event, but does *not* execute any commands that would change the status of the Cluster Manager. For descriptions of cluster events and how the Cluster Manager processes these events, see the *Administration Guide* for more information.

You can run the Event Emulator through SMIT or from the command line. The Event Emulator runs the events scripts on every active node of a stable cluster, regardless of the cluster's size. The output from each node is stored in an output file on the node from which the event emulator is invoked. You can specify the name and location of the output file using the environment variable **EMUL_OUTPUT**, or use the default output file, **/var/hacmp/log/emuhacmp.out**.

Note: The Event Emulator requires that both the Cluster Manager and the Cluster Information Program (**clinfo**) be running on your cluster.

The events emulated are categorized in two groups:

- Cluster events
- Dynamic reconfiguration events.

Emulating Cluster Events

The cluster events that can be emulated are:

node_up	fail_standby
node_down	join_standby
network_up	swap_adapter
network_down	

Emulating Dynamic Reconfiguration Events

The dynamic reconfiguration event that can be emulated is Synchronize the HACMP Cluster.

Restrictions on Event Emulation

Note: If your current cluster does *not* meet any of the following restrictions, you can use the cluster test tool as an alternative to executing cluster events in emulation mode. The cluster test tool performs real cluster events and pre- and post-event customizations.

The Event Emulator has the following restrictions:

- You can run only one instance of the event emulator at a time. If you attempt to start a new emulation in a cluster while an emulation is already running, the integrity of the results cannot be guaranteed.
- **clinfo** must be running.
- You cannot run successive emulations. Each emulation is a standalone process; one emulation cannot be based on the results of a previous emulation.
- When you run an event emulation, the Emulator’s outcome may be different from the Cluster Manager’s reaction to the same event under certain conditions:
  - The Event Emulator will *not* change the configuration of a cluster device. Therefore, if your configuration contains a process that makes changes to the Cluster Manager (disk fencing, for example), the Event Emulator will *not* show these changes. This could lead to a different output, especially if the hardware devices cause a fallover.
  - The Event Emulator runs customized scripts (pre- and post-event scripts) associated with an event, but does *not* execute commands within these scripts. Therefore, if these customized scripts change the cluster configuration when actually run, the outcome may differ from the outcome of an emulation.
- When emulating an event that contains a customized script, the Event Emulator uses the **ksh** flags **-n** and **-v**. The **-n** flag reads commands and checks them for syntax errors, but does *not* execute them. The **-v** flag indicates verbose mode. When writing customized scripts that may be accessed during an emulation, be aware that the other **ksh** flags may *not* be compatible with the **-n** flag and may cause unpredictable results during the emulation. See the **ksh** man page for flag descriptions.

Emulation of Error Log Driven Events

Although the HACMP software does *not* monitor the status of disk resources, it does provide a SMIT interface to the AIX Error Notification facility.

HACMP uses the following utilities for monitoring purposes:

- RSCT
- AIX Error Notification
- RMC
- User-defined events
- Application monitoring.

The AIX Error Notification facility allows you to detect an event *not* specifically monitored by the HACMP software—a disk adapter failure, for example—and to program a response (notification method) to the event. In addition, if you add a volume group to a resource group, HACMP automatically creates an AIX Error Notification method for it. In the case where the loss of quorum error occurs for a mirrored volume group, HACMP uses this method to selectively move the affected resource group to another node. Do *not* edit or alter the error notification methods that are generated by HACMP.

HACMP provides a utility for testing your error notification methods. After you add one or more error notification methods with the AIX Error Notification facility, you can test your methods by emulating an error. By inserting an error into the AIX error device file (**/dev/error**), you cause the AIX error daemon to run the appropriate pre-specified notification method. This allows you to determine whether your pre-defined action is carried through, without having to actually cause the error to occur.

When the emulation is complete, you can view the error log by typing the **errpt** command to be sure the emulation took place. The error log entry has either the resource name EMULATOR, or a name as specified by the user in the **Resource Name** field during the process of creating an error notification object.

You will then be able to determine whether the specified notification method was carried out.

Chapter 8: HACMP 5.4.1: Summary of Changes

This chapter lists all new or enhanced features in HACMP 5.4.1 and also notes discontinued features.

- List of New Features
- Discontinued Features

List of New Features

HACMP Version 5.4.1 helps protect critical business applications from outages. For over a decade, HACMP has been providing reliable monitoring, failure detection, and automated failover for 24 x 7 business application environments. The optional HACMP Extended Distance (HACMP/XD) feature adds unlimited distance data mirroring and recovery solutions for critical business needs; the optional HACMP Smart Assist feature helps you easily deploy high availability into your critical applications.

HACMP v5.4.1 offers you:

- AIX WPAR support so you can attain high availability for your applications by configuring them as a resource group and assigning the resource group to an AIX WPAR. By using HACMP in combination with AIX WPAR, you can leverage the advantages of application environment isolation and resource control assignment (provided by AIX WPAR) and the high availability feature provided by HACMP v5.4.1.
- HACMP/XD support of PPRC Consistency Groups and Freeze. HACMP/XD allows you to maintain data consistency for application-dependent writes on the same logical subsystems (LSS) pair or across multiple LSS pairs. Failure of a PPRC resource triggers a freeze of all PPRC pairs in all consistency groups managed by a HACMP resource group.
- New Geographical Logical Volume Manager (GLVM) Status Monitor that is comprised of two new line mode commands that provide the ability to monitor GLVM status and state. These monitors better enable you to keep track of the state and ongoing status of RPVs and GMVGs.
- NFSv4 improvements that bring greater convenience in configuring NFSv4 exports, as well as improved failover of resource groups with NFSv4 exports.
- HACMP usability updates to WebSMIT and First Failure Data Capture that help improve your experience with managing HACMP.

New Features That Enhance Ease of Use

These features make the product easier to use:

- AIX Workload Partitions
- HACMP/XD Support of PPRC Consistency Groups and Freeze
- NFSv4 Support
- HACMP Usability Improvements

- AIX Workload Partitions
- Better Handling of Stopping and Starting HACMP Cluster Services
- Resource Group Management (clRGmove) Enhancements
- Verification Enhancements
- Improved WebSMIT Application

AIX Workload Partitions

WPAR, a part of AIX 6.1, is software-created virtualized operating system environments within a single instance of the AIX operating system. To most applications, the WPAR appears to be a separate instance of AIX. This is because applications and WPARs have a private execution environment. Applications are isolated in terms of process, signal, and file system space. Workload partitions have their own unique users and groups. Workload partitions have dedicated network addresses, and interprocess communication is restricted to processes executing in the same WPAR. You can attain high availability for your applications by configuring them as a HACMP resource group and assigning the resource group to an AIX WPAR. Using this approach, you can take advantage of the AIX WPAR features combined with the high availability features provided by HACMP.

In HACMP, a resource group is a logical collection of inter-related resources (such as applications, volume groups, and IP addresses) treated as a single unit in the context of availability. In HACMP V5.4.1, a resource group can be assigned to an AIX WPAR. When the resource group is brought online, HACMP associates all the resources of the resource group with the corresponding WPAR. All the applications, volume groups, filesystems, and IP addresses would get assigned to the WPAR on the node that has been selected to be the host of the resource group. You can control the selection of the node using HACMP configurations. The application is then started within the corresponding WPAR. By using HACMP in combination with AIX WPAR, you can leverage the advantages of application environment isolation and resource control assignment (provided by AIX WPAR) and the high availability feature provided by HACMP V5.4.1. By choosing to run your application within a WPAR, you can control the amount of resources that a certain application should use. This can be achieved by assigning a certain percentage of resources (like CPU, memory, and number of processes) to the WPAR that hosts the application.

HACMP/XD Support of PPRC Consistency Groups and Freeze

HACMP/XD supports the IBM TotalStorage disk subsystem Peer-to-Peer Remote (PPRC) function. The support involves defining a set of local and remote ESS disk volumes, whose contents are required to be kept in sync in real-time using the PPRC feature, as HACMP replicated resources. When HACMP is configured to be aware of PPRC-based replicated resources, PPRC commands and tasks are invoked from the PPRC interface to manage the PPRC relationships of these resources.

The PPRC metro mirror, or synchronous protocol, guarantees that the secondary copy is up-to-date and consistent by ensuring that the primary copy is written only if the primary receives acknowledgement that the secondary copy has been written. HACMP manages the volume relationships by either executing saved tasks or dynamically invoking command line API. Such operations are currently performed on each individual PPRC pair basis. For applications that have dependent writes, customers would like to be able to maintain data consistency of their application data, which is not possible if PPRC volumes are only managed on an individual pair basis. PPRC provides the consistency group feature that enables PPRC relationships to be combined and managed together in a group to maintain data consistency, which is essential for logical data integrity. PPRC consistency grouping also facilitates normal restart of databases after a disaster. This feature provides HACMP support for PPRC consistency groups and freeze actions.

By using PPRC Consistency Groups, you can maintain data consistency for application dependent writes on the same LSS pair or across multiple LSS pairs. Failure of a PPRC resource triggers a freeze of all PPRC pairs in all consistency groups managed by an HACMP resource group.

In several applications, one write is dependent on the completion of another. Such applications are said to have *dependent writes*. Using dependent writes, such applications are able to manage the consistency of their data so that a consistent state of the application data on disk is maintained if a failure occurs in the host processor, software, or storage subsystem. A common example of application dependent writes are databases and their associated log files. Database data sets are related, with values and pointers from indexes to data. Databases have pointers inside the data sets, in the database catalog and directory data sets, and in the logs. Therefore, data integrity must always be kept across these components of the database.

In disaster situations, it is unlikely that the entire complex will fail at the same moment. Failures tend to be intermittent and gradual, and disaster can occur over many seconds, even minutes. Because some data may have been processed and other data lost in this transition, data integrity on the secondary volumes is exposed. This situation is called a *rolling disaster*. The mirrored data at the recovery site must be managed so that cross-volume or LSS data consistency is preserved during the intermittent or gradual failure. Using the database scenario in which the database and its log files are on different volumes, as the database mirror fails, dependent logfile writes can still be mirrored and data consistency lost. By grouping these volumes in a consistency group, data consistency can be maintained across the recovery site.

This new functionality of HACMP/XD requires the **cluster.es.spprc** fileset. PPRC Consistency Groups are dependent upon the traps received and processed by HACMP/XD support of ESS SNMP Traps feature. Failures in the PPRC pair, whether hardware or network, cause the storage unit to broadcast an SNMP trap. The traps are received and processed by this new functionality. SNMP traps are now part of the base HACMP.

GLVM Status Monitor

GLVM provides replication of your data to a remote site over IP networks. It is available in stand-alone form with the base AIX. It is also available with HACMP/XD to provide an integrated remote replication and high availability disaster recovery solution. As more customers implement GLVM, and as customers put greater workloads upon GLVM, there is increasing need for real-time monitoring of GLVM state. The GLVM Status monitor meets this requirement and helps build confidence in the GLVM function by keeping you aware of ongoing GLVM status information.

HACMP/XD V5.4.1 includes the new GLVM Status Monitor. This new status monitor has two new line mode commands (along with associated SMIT panel interfaces) that enable you to monitor GLVM status and state. The **rpvstat** command provides real-time monitoring of GLVM remote physical volumes (RPVs). The **gmvgstat** command provides monitoring of GLVM geographically mirrored volume groups (GMVGs). Together, these new monitors better enable you to keep track of the state and ongoing status of RPVs and GMVGs.

By using the new **rpvstat** and **gmvgstat** commands, you are able to monitor the behavior and status of GLVM RPVs and GMVGs. For instance, the **rpvstat** command shows accumulated counts of completed and pending reads, writes, kilobytes read, and kilobytes written, and device errors for one or more RPVs. It can also be used to display the maximum recorded numbers of pending reads, writes, and kilobytes to be read and pending kilobytes to be written to an RPV device ("high water mark" values). For instance, the **gmvgstat** command can be used to display the total number of PVs (physical volumes), RPVs, stale volumes, total PPs (physical partitions), and stale PPs, as well as the synchronization percentage for one or more GMVGs. Both commands provide monitor modes, which allow the commands to run continuously and display updated information on a user-supplied interval basis.

You can use the GMVG Status Monitor tools, for example, to help detect failing or marginal mirroring network links, as well as determine the resynchronization status of the remote copy of a GMVG that has recently been restored and reintegrated.

The GLVM Status Monitor is a new function of GLVM. GLVM requires AIX 5.3, or later.

NFSv4 Support

NFSv4 support improvements include:

- Better failover of client state using stable storage.
- Support for configuring NFSv4 exports from SMIT screens. Previously, administrators had to edit the exports file to specify the protocol version.
- Support for configuring a file system to be exported with both NFSv2/3 and NFSv4.
- A smart assist to help create and modify resource groups with NFS exports.

NFSv4 support improvements bring greater convenience for configuring NFSv4 exports, as well as improved failover of resource groups with NFSv4 exports.

You now have greater flexibility in configuring NFS with HACMP. With a mix of NFSv3 and NFSv4, you are now able to export a file system to all clients. If you currently have NFSv3 and plan to migrate to NFSv4, you can export file systems with both protocols to allow for gradual adoption of NFSv4.

Prerequisites required:

- AIX 5.3 with Technology Level 5300-07, or later
- AIX 6.1, or later
- The **bos.net.nfs.client** fileset with v5.3.7.0, or later
- The **bos.net.nfs.server** fileset with v5.3.7.0, or later

HACMP Usability Improvements

A number of improvements have been made to the ease-of-use, performance, reliability, availability, and serviceability of the HACMP product. These improvements include changes for WebSMIT customization, performance, ease-of-use and reliability, and improved reliability for cluster event handling, First Failure Data Capture, improved logging and metric reporting. These changes are focused on improving your experience with managing HACMP.

The WebSMIT user interface now provides a more finished and customizable look-and-feel, improving both usability and accessibility. Performance, reliability, and the overall user interface experience have been much improved.

First Failure Data Capture provides critical data to the service team when a problem is encountered and reported by a customer.

The progress indicator for cluster verification now reports a finer level of detail so the administrator knows which phase, and at what percentage done, the utility has achieved.

HACMP Smart Assist Programs Enhancements

In HACMP 5.4.1, the three HACMP Smart Assists are enhanced to include more automatic discovery to help you easily integrate these applications into an HACMP cluster:

- **Smart Assist for WebSphere**. Extends an existing HACMP configuration to include monitoring and recovery support for various WebSphere components.
- **Smart Assist for DB2**. Extends an existing HACMP configuration to include monitoring and recovery support for DB2 Universal Database (UDB) Enterprise Server Edition.
- **Smart Assist for Oracle**. Provides assistance to those involved with the installation of Oracle® Application Server 10g (9.0.4) (AS10g) Cold Failover Cluster (CFC) solution on IBM AIX™ (5200) operating system.

HACMP 5.4.1 also includes a **General Configuration Smart Assist** that helps you quickly configure other applications.

HACMP 5.4.1 provides a documented Smart Assist Framework and API that allows users to write a Smart Assist program to integrate their own applications with HACMP.

HACMP Smart Assist for Oracle

HACMP Smart Assist for Oracle has been greatly improved. It helps you to configure Oracle Application Server and/or an associated Oracle database, in a highly available cluster environment. HACMP 5.4.1 Smart Assist for Oracle extends and improves upon the high availability solutions available in HACMP 5.3.

HACMP 5.4.1 Smart Assist for Oracle helps you to configure Oracle components in one of the multiple high availability configurations. You can choose the best high availability configuration for your environment based upon Oracle's recommendations, and then use the SMIT screens to implement the configuration in HACMP under the AIX environment.

HACMP Smart Assist for Oracle also monitors Oracle Application Server and Oracle database instances and processes. In order to bring the Application Server and database under the control of HACMP, you must have pre-installed the respective Oracle components.

Better Handling of Stopping and Starting HACMP Cluster Services

In HACMP 5.4.1, your options for starting, stopping and restarting cluster services have been streamlined and improved to allow you full control over your applications, without disrupting them.

You can:

- *Start and restart HACMP cluster services*. When you start cluster services, or restart them after a shutdown, HACMP by default automatically activates the resources according to how you defined them, taking into consideration application dependencies, application start and stop scripts, dynamic attributes and other parameters. That is, HACMP automatically manages (and activates, if needed) resource groups and applications in them.

You can also start HACMP cluster services and tell it *not to start up any resource groups* (and applications) automatically for you. If an application is already running, you no longer need to stop it before starting the cluster services. HACMP relies on the application monitor and application startup scripts to verify whether it needs to start the application for you or if the application is already running (HACMP tries *not* to start a second instance of the application).

- *Shut down HACMP cluster services*. You can tell HACMP to stop cluster services on the node(s) and along with this action to either bring the resources and applications offline, move them to other nodes, or keep them running on the same nodes (but stop managing them for high availability).

  The Cluster Manager "remembers" the state of the nodes and responds appropriately when users attempt to restart the nodes.

For detailed information on how to configure application monitors as well as HACMP cluster startup and shutdown options, see the chapter on Starting and Stopping Cluster Services in the *Administration Guide.*

Resource Group Management (clRGmove) Enhancements

In HACMP 5.4.1, the Resource Group Management utility, **clRGmove**, has been improved:

- **Improved SMIT interface**. Using a more straightforward user interface in SMIT, you can select a resource group and then a node (at either site) to which you want to move it. SMIT also informs you (in the picklists with destination nodes) if it finds that a node with a higher priority exists that can host a group. You can always choose to move the group to that node. (This is useful for groups with **Fallback to Highest Priority Node** fallback policy. Such groups will fall back to their "new" nodes, once you move them).
- **Proper handling of non-concurrent resource groups with No Fallback resource group policies**. In HACMP 5.4.1, once you move such a resource group, it remains on the destination node until you tell HACMP where to move it again, and does *not* fall back immediately to the node from which it was moved. Note, it may move to other nodes later, if HACMP reacts to some potential cluster events and has to take action to recover or redistribute resource groups.
- **Proper handling of non-concurrent resource groups with No Fallback site policy**. Similarly, HACMP 5.4.1 now honors the **No Fallback** policy *for sites,* when you move resource groups between sites. For instance, a resource group contains nodes that belong to different sites and the site policy is **No Fallback**. If you move the group to a node at another site, it does *not* fall back to its primary site immediately after you move it. Instead, it remains on the node at another site either until you tell HACMP to move it again, or, until, upon subsequent cluster events, HACMP may decide to move it.
- **Clear method to maintain the previously configured behavior for a resource group**. Once you move a resource group, this does *not* change the nodelist that was specified for this resource group before you moved it, or its startup, fallover or fallback policies. It may temporarily change the home node (for groups that have **Fallback to Highest Priority Node** fallback policy). The node to which you move such a group becomes its temporary home node and it falls back to this node.

In HACMP 5.4.1, when you move a resource group, it either stays on the destination node until you move it again, or HACMP moves it upon subsequent cluster events. This consistent behavior is especially important to notice for those resource groups that have the **Fallback to Highest Priority Node** fallback policy: Such resource groups fall back to their “new” nodes. Note that this new node may *not* be the highest priority node available. If HACMP sees this, SMIT clearly indicates to you that in the list of destination nodes there *is* a higher priority node available to which you can always move the group.

- **Improved status and troubleshooting utilities**. You can now use **clRGinfo -p** to obtain the history of the last cluster event that caused the resource group to move. (You can use this command only if HACMP is running on the nodes).
- **No need to set the Priority Override Location (POL)** for the node to which a resource group is moved. POL is a setting you had to specify for manually-moved resource groups in releases prior to HACMP 5.4.1. In HACMP 5.4.1, you no longer have set it when moving a resource group to another node.

Verification Enhancements

The following enhancements have been added to the verification process:

- *Non-IP Network Enhancement.* Verification checks that each node can reach each other node in the cluster through non-IP connections. If this is *not* true, a message is displayed. In addition, the SMIT **Configure HACMP Communication Interfaces/Devices** panel has been enhanced so that all 16 characters of the PVID field are viewable.
- *Failed component warnings.* The final verification report lists any nodes, networks and/or network interfaces that are in the 'failed' state at the time that cluster verification is run. The final verification report also lists other 'failed' components, if accessible from the Cluster Manager, such as applications, resource groups, sites, and application monitors that are in the suspended state. (Resource groups may be listed as in the ‘unmanaged’ state.)
- *Volume group verification checks.* Volume group verification checks have been restructured for faster processing.
- *Message format*. Messages have been reformatted for consistence and to remove repetitious entries.
- *Invalid netmasks*. Verification now checks for valid netmasks.
- *Broadcast addresses*. Verification now checks for valid broadcast addresses.
- *Mixed Volume Group state.* All Volume Groups and PVIDs must be on the vpath devices, *not* the hdisks when an SDD VPATH Device Volume Group is installed. When PVs are on hdisks on one node, but vpaths on another*,* an error occurs. The exception occurs when the hdisks are being managed by SDDPCM, which does not present vpath devices like SDD.
- *Persistent labels.* Verification now checks collocation or anti-collocation used with persistent label distribution preferences when persistent labels have *not* been defined. When a “with persistent” distribution preference is selected and no persistent label has been defined, a warning message is displayed during cluster verification,

Improved WebSMIT Application

With HACMP 5.4.1, WebSMIT expands its cluster management function with these enhancements:

- New WebSMIT framework for the user interface
- Graphical representation of:
  - Resource groups and their dependencies
  - Cluster site, network, and node information
- Ability to simultaneously view the cluster configuration and the cluster status
- Ability to navigate the running cluster
- Assisted WebSMIT set up

  To configure WebSMIT, you modify a sample post-install script to:
  - Copy and/or link the WebSMIT HTML and CGI files to the appropriate location
  - Change file permissions as needed
  - Update the Web server's **httpd.conf** file
  - Update the **wsm_smit.conf** file with the proper settings, if needed.
- User Authentication utility (optional). Administrators can specify a group of users that have read-only access. Those users can view the cluster configuration and status, and navigate through SMIT stanza screens, but *not* execute commands or make changes.
- Support for Mozilla-based browsers (Mozilla 1.7.3 for AIX and FireFox 1.0.6) in addition to Internet Explorer versions 6.0 and higher.

Cluster Test Tool

HACMP 5.4.1 enhances the Cluster Test Tool to support a more complete set of cluster events, including tests for managing resources, resource groups, and sites. The tool performs tests to manage moving resource groups, failing and joining various resources, and rudimentary support for stopping and starting entire sites.

In addition, HACMP 5.4.1 adds specific test plans for running site tests, non-IP network tests, IP network tests, and volume group tests, and extends the logic in the automated test tool to run these test plans as appropriate based on the cluster configuration. For more information, see the chapter about testing your cluster in the *Administration Guide*.

Fast Method for Node Failure Detection

HACMP uses the fast method for node failure detection and takes considerably less time to detect a node failure that occurred in the cluster, while reliably detecting node failures. As a mechanism for fast node failure detection, when a node fails, HACMP 5.4.1 uses disk heartbeating to place a *departing* message on the shared disk so neighboring nodes are aware of the node failure within one heartbeat period.

Remote nodes that share the disks receive this message and broadcast that the node has failed. Directly broadcasting the node failure event greatly reduces the time it takes for the entire cluster to become aware of the failure compared to waiting for the missed heartbeats, and therefore HACMP can take over critical resources faster.

Starting with HACMP 5.4, you can reduce the time it takes to detect a node failure by configuring disk heartbeating networks and specifying an FFD_ON parameter for the disk heartbeating network NIM. Once the cluster configuration contains disk heartbeating networks and this parameter is specified in SMIT, HACMP uses the fast method of node failure detection.

For more information, see the section Decreasing Node Fallover Time in Chapter 3: Planning Cluster Network Connectivity of the *Planning Guide*, and a chapter on configuring NIMs in the *Administration Guide*.

Features That Enhance Geographic Distance Capability

These features add to the capability for distributing the cluster over a geographic distance, for improved availability and disaster recovery.

The following functions are supported in clusters with HACMP/XD for GLVM 5.4.1:

- **Enhanced Concurrent Volume Groups**. In addition to non-concurrent volume groups, you can have enhanced concurrent mode volume groups configured with RPVs, so that they can serve as geographically mirrored volume groups. You can include such volume groups into both concurrent and non-concurrent resource groups in an HACMP cluster with GLVM.

  If you have enhanced concurrent volume groups, this means that you can also configure disk heartbeating over disks that belong to a geographically mirrored volume group, and that also belong to the nodes at the same site. (NOTE: disk heartbeating across sites is *not* supported. The heartbeating function is already performed in the HACMP cluster through an XD_ip network). Note, however, that another useful function that is allowed on enhanced concurrent volume groups in base HACMP—fast disk takeover, is *not* supported for geographically mirrored volume groups that are also enhanced concurrent.

- **Multiple Data Mirroring Networks**. In an HACMP cluster that has sites configured, you can now have up to four XD_data networks used for data mirroring (in previous releases, HACMP/XD for GLVM allowed only one XD_data network). Having more mirroring networks in the cluster increases data availability and mirroring performance. For instance, if one of the mirroring networks fails, the GLVM mirroring can continue over the redundant networks. Also, you have the flexibility to configure several low bandwidth XD_data networks and take advantage of the aggregate network bandwidth (you can also combine high bandwidth networks in the same manner).

  Plan the data mirroring networks in a way that they provide similar network latency and bandwidth, since each RPV client communicates with its corresponding RPV server over *more than one* IP-based network at the same time to ensure load balancing.

- **IP Address Takeover (IPAT) via IP Aliasing is Supported on XD-type Networks**. HACMP/XD for GLVM 5.4.1 supports adding highly available alias service IP labels to XD-type networks. IP Address Takeover via IP Aliases is now the default on XD networks.This allows users to create an alias service IP label on an XD network that can reside on multiple nodes.

  HACMP 5.4.1 lets you configure *site-specific service IP labels*, thus you can create a resource group that activates a given IP address on one site and a different IP address on another site.

Other Features

This section lists additional features that are related to HACMP.

Upgrading HACMP 5.4.1 with HACMP Cluster Services and Applications Running

With HACMP 5.4.1, you can upgrade the HACMP software on an individual node using *rolling migration,* while your critical applications and resources continue running on that node though they will *not* be highly available during the upgrade.

Starting Cluster Services While Applications Continue to Run

With HACMP 5.4.1, you can allow your applications that run outside of HACMP to continue running during installation of HACMP and when starting HACMP. There is no need to stop, restart or reboot the system or applications.

Stopping Cluster Services

With HACMP 5.4.1 you can stop cluster services using the **Unmanage Resource Groups** option (prior to HACMP 5.4, this was known as forced down) on a maximum of one node at a time. You then upgrade the node, start cluster services to begin monitoring resource groups and make the running applications highly available, and reintegrate the node into the cluster before upgrading the next node.

Other options for stopping cluster services include **Bring Resource Groups Offline** (formerly known as **Graceful stop**) and Move Resource Groups (formerly known as **Graceful with Takeover**).

Prior to HACMP 5.4, after installing HACMP you were required to restart your nodes in order to start all HACMP subsystems and to set some attributes in the scripts. In addition, the reboot process was required to keep the **clinfo** utility and the RSCT deadman switch (DMS) synchronized. HACMP 5.4.1 takes care of these issues without requiring you to stop and restart your applications.

For more information about installing HACMP while keeping critical applications running, see the *Installation Guide*.

Discontinued Features

Features discontinued in HACMP 5.2 and 5.3 are listed here.

Features Discontinued in HACMP 5.2 and Up

The following features, utilities, or functions are discontinued starting with HACMP 5.2:

- The **cllockd** or **cllockdES** (the Cluster Lock Manager) **is no longer supported**. During node-by-node migration, it is uninstalled. Installing HACMP 5.2 or 5.3 removes the Lock Manager binaries and definitions. Once a node is upgraded to HACMP 5.2 or 5.3, the Lock Manager state information in SNMP and **clinfo** shows the Lock Manager as being in the down state on all nodes, regardless of whether or *not* the Lock Manager is still running on a back-level node.

Before upgrading, make sure your applications use their proprietary locking mechanism. Check with your application's vendor about concurrent access support.

- **Cascading, rotating and predefined concurrent resource groups are *not* supported**. Also, Cascading without Fallback and Inactive Takeover settings are *not* used. In HACMP 5.2 and 5.3 you can continue using the groups migrated from previous releases. You can also configure these types of groups using the combinations of startup, fallover and fallback policies available for resource groups in HACMP 5.2 and 5.3. For information on how the predefined resource groups and their settings are mapped to the startup, fallover and fallback policies, see the chapter on upgrading to HACMP 5.3 in the *Installation Guide.*
- **Manual reconfiguration of user-defined events is required**. HACMP 5.2 and 5.3 interact with the RSCT Resource Monitoring and Control (RMC) subsystem instead of with the Event Management subsystem. This affects the following utilities:
  - Dynamic Node Priority (DNP)
  - Application Monitoring
  - User-Defined Events (UDE).

  You must manually reconfigure all user-defined event definitions with the exception of the several user-defined event definitions defined by DB2. The **clconvert** command only converts a subset of Event Management user-defined event definitions to the corresponding RMC event definitions. For complete information on the mapping of the Event Management resource variables to the RMC resource attributes, see Appendix D: RSCT: Resource Monitoring and Control Subsystem in the *Administration Guide.*

Features Discontinued in HACMP 5.3

The following features, utilities, or functions are discontinued starting with HACMP 5.3:

- **Changes due to the new architecture of Clinfo and Cluster Manager communication:**
  - The **clsmuxpd** daemon is eliminated in HACMP 5.3. Clinfo now only obtains data from SNMP when requested; it no longer obtains the entire cluster configuration at startup.
  - **Shared memory** is no longer used by Clinfo in HACMP 5.3. Client API requests and responses now flow through the message queue connection with the **clinfo** daemon.
  - HACMP 5.3 removes the **cl_registerwithclsmuxpd()** API routine; application monitoring effectively supersedes this function. If before upgrading to HACMP 5.3, you were using the pre- or post-event script for your application that referenced the **cl_registerwithclsmuxpd** API, the scripts will no longer work as expected. Instead, use the application monitoring function in HACMP, accessible through HACMP SMIT. See Chapter 5: Ensuring Application Availability in this guide for more information.

- **Changes to utilities available from the command line:**
  - The **cldiag** utility. The **cldiag** utility is no longer supported from the command line; however it is available from the HACMP SMIT Problem Determination Tools menu in an easier, more robust format. **The cldiag command line utility is deprecated for HACMP 5.3**.
  - The **clverify** utility.The **clverify** utility is no longer supported from the command line; however it is available from the HACMP SMIT Verification and Synchronization menu in an easier, more robust format. **The clverify command line utility is deprecated for HACMP 5.3**.

Where You Go from Here

For planning an HACMP cluster and installing HACMP, see the *Planning* and *Installation Guides.*

For configuring HACMP cluster components and troubleshooting HACMP clusters, see the *Administration* and *Troubleshooting Guides*.

Notices for HACMP Concepts and Facilities Guide

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions; therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Dept. LRAS / Bldg. 003
11400 Burnet Road
Austin, TX 78758-3493
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Index